

Algorithmische Analyse
makroskopischer Verbindungsstrukturen
im Primatengehirn

Inaugural-Dissertation

zur

Erlangung des Doktorgrades der
Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Antje Krumnack

aus Bonn

Mai 2008

Aus dem Institut für Informatik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit Genehmigung
der Mathematisch-Naturwissenschaftlichen Fakultät der Universität Düsseldorf

Referent: Prof. Dr. E. Wanke

1. Koreferent: Prof. Dr. R. Kötter

Tag der mündlichen Prüfung:

Zusammenfassung

In dieser Arbeit werden Fragen untersucht, die sich im Zusammenhang mit der Analyse von Verbindungsstrukturen im Primatengehirn stellen. Diese Probleme der Neurologie lassen sich mit Methoden der theoretischen Informatik bearbeiten. Das Ziel dieser Arbeit ist es, zunächst Formulierungen der Probleme zu finden, die neurologisch sinnvoll sind und mathematisch eine solide Grundlage bilden, um dann effiziente Algorithmen zur Lösung zu entwickeln. Zu allen vorgestellten Methoden werden auch konkrete Berechnungen mit neurologischen Daten durchgeführt.

Die Arbeit gliedert sich in folgende Bereiche:

Zunächst wird der Einfluß eines Knotens auf die Zusammenhangsstruktur eines gerichteten oder ungerichteten Netzwerkes mit Hilfe der Shapleywerttheorie gemessen. Im allgemeinen ist die Berechnung solcher Bewertungen sehr ineffizient. Es wird hier gezeigt, wie die Berechnung für viele Instanzen handhabbar gemacht werden kann, indem man große Netzwerke in kleinere Teile zerlegt.

Desweiteren werden die häufig zitierten Konzepte von Hierarchien und Streams als Organisationsprinzipien von Areal-Netzwerken untersucht. Dazu wird zuerst ein Hierarchiebegriff eingeführt, der eine kontinuierliche statt, wie bisher üblich, eine diskrete Skala verwendet. Mit einer solchen Skala lassen sich feinere Abstufungen in der Hierarchiestruktur wiedergeben. Es werden Algorithmen für die Berechnung solcher Hierarchien in gerichteten, gewichteten Graphen angegeben. Zusätzlich wird auch eine Optimierungsversion des Problems vorgestellt, mit der eine möglichst gute Hierarchie gefunden werden kann, falls der Graph keine exakte Hierarchie beschreibt.

Danach wird der Frage nachgegangen, ob es ein strukturelles Korrelat paralleler spezifischer Informationsverarbeitungsströme (Streams) innerhalb des visuellen Systems gibt. Es wird versucht, eine geeignete mathematische Definition für die getrennten Verarbeitungswege von räumlichen und Objektinformationen zu formulieren, um damit Streams nur anhand der Verbindungsinformation für das visuelle System zu identifizieren.

Abschließend wird das Problem inkongruenter Einteilungen von Hirnstrukturen (Parzellierungsproblem) betrachtet. Es wird eine Möglichkeit vorgestellt, Informationen aus dem koordinaten-basierten CARET-System mit den koordinatenunabhängigen Mappingaussagen aus der CoCoMac-Datenbank in Beziehung zu setzen. Die Ergebnisse dieses Verfahrens können auf Widersprüche im Datenbestand zwischen den beiden Systemen hinweisen.



Abstract

This thesis focuses on problems that arise in connection with the analysis of connectivity structures in the primate brain. These issues of neurology can be studied using methods of theoretical computer science. The main aim of this thesis is to find definitions for these problems that are neurologically meaningful and provide a mathematically sound basis for the development of efficient algorithm to solve them. Calculations for neurological data are done with every introduced method.

The thesis is structured in several larger sections.

First the influence of a vertex on the connectivity structure of a directed or undirected network is measured using Shapley value theory. In general calculations of such ratings are highly inefficient. It is shown how these calculations can be managed for many instances by the decomposition of large networks into smaller parts.

Secondly the often cited concepts of hierarchies and streams as organizational principles of connectivity networks are addressed. For that purpose a hierarchy notion is introduced which uses a continuous scale and not, as is customary, a discrete scale. Through this change finer nuances of the hierarchy structure can be expressed. Algorithms are specified that can compute such hierarchies for directed weighted graphs. In addition an optimization model is specified that approximates a hierarchy if the graph does not delineate an exact hierarchy.

Then the question is addressed if there is a structural correlate of parallel specific information processing streams within the visual system. An attempt is made to find a mathematical definition for the separate processing paths for spacial and object information so that streams can be identified using just the connectivity information of the visual system.

Finally the problem of incongruent partitions of brain structures (parcellation problem) is addressed. A method is introduced to relate coordinate based information from the CARET-system to the coordinate independent mapping information from the CoCoMac data base. This can point to contradictions in the data sets of both systems.



Inhaltsverzeichnis

1	Einleitung	1
2	Grundlegende Begriffe der Neurologie	7
2.1	Parzellierungsschemata	8
2.2	Funktionelle Systeme	8
2.3	Tracing-Studien	10
2.4	Die CoCoMac-Datenbank	13
3	Grundlegende Begriffe der Graphentheorie	15
3.1	Definitionen für ungerichtete Graphen	16
3.2	Definitionen für gerichtete Graphen	17
3.3	Darstellung von Gehirnetzwerken durch Graphen	18
4	Knotenbewertungen in Graphen	21
4.1	Einführung	21
4.2	Shapleywerte und ihre Verwendung in der Graphentheorie	22
4.2.1	Shapleys Axiome	22
4.2.2	Übertragung von Shapleys Axiomen auf Graphen	25
4.2.3	Arbeiten zu Shapleywerten und Graphen	26
4.3	Knotenbewertungen für gerichtete Graphen	28
4.4	Das Entscheidungsproblem „starke Separierbarkeit von zwei Knoten“ ist NP-vollständig	39
4.5	Knotenbewertungen für ungerichtete Graphen	41
4.5.1	Cycle-composed Graphen	44
4.5.2	Chordale Graphen	47
4.5.3	Module-composed Graphen	47
4.6	Berechnungen von Knotenbewertungen in neuronalen Netzwerken	54
4.6.1	Knotenbewertungen im visuellen System	54

4.6.2	Knotenbewertungen im präfrontalen Cortex	56
4.6.3	Diskussion	59
5	Hierarchien in funktionellen Systemen	63
5.1	Einführung	63
5.1.1	Die Arbeit von Felleman und Van Essen	64
5.1.2	Kritik an der Arbeit von Felleman und Van Essen	66
5.1.3	Ein mögliches Abstandsmaß für Verbindungen	68
5.1.4	Ein neues Modell zur Bestimmung von Hierarchien	69
5.2	Reellwertige Hierarchien	70
5.2.1	Das einfache Hierarchie-Problem	71
5.2.2	Das Intervall-Hierarchie-Problem	73
5.2.3	Das Δ -Hierarchie-Problem	80
5.2.4	Das Intervall- Δ -Hierarchie-Problem	86
5.3	Lösen der Hierarchie-Probleme mit Hilfe linearer Optimierung	90
5.3.1	Lineare Optimierung	90
5.3.2	Hierarchieberechnung mit linearer Optimierung	91
5.4	Berechnung von Hierarchien im visuellen System	96
5.4.1	Verwendung des SLN%-Wertes als Abstandsmaß	96
5.4.2	Berechnungen mit den Daten von Felleman und Van Essen	103
5.4.3	Diskussion	112
6	Streams im visuellem System	115
6.1	Einführung	115
6.2	Die Untersuchung von Young	116
6.3	Kriterien für Streams	118
6.4	Finden von Start- und Zielarealen	118
6.5	Verschiedene Ansätze zur Definition von Streams	120
6.5.1	Definition über die Verbindungen	120
6.5.2	Definition über die Hamming-Distanz der Adjazenzen	127
6.5.3	Definition über kürzeste Wege	131
6.5.4	Definition über kürzeste und um eins längere Wege	137
6.6	Diskussion	142
7	Das Parzellierungsproblem	145
7.1	Einführung	145

7.2	Verwendete Daten	146
7.3	Formalisierung der Begriffe	148
7.4	Bestimmung der Beziehungen zwischen den Arealen	149
7.4.1	Verfahren 1	150
7.4.2	Verfahren 2	156
7.4.3	Verfahren 3	162
7.5	Diskussion	164
 Literaturverzeichnis		 167
 A Verwendete Verbindungsdaten		 173
 B Berechnete Daten		 187

INHALTSVERZEICHNIS

Kapitel 1

Einleitung

Der zerebrale Cortex läßt sich gemäß unterschiedlicher Parzellierungsschemata in viele verschiedene Regionen einteilen [FE91, KW05, Zil04]. Diese Regionen, auch Areale genannt, sind untereinander verbunden und können so Informationen austauschen. Diese Verbindungsstrukturen bilden die Grundlage für eine komplexe Informationsverarbeitung. Deshalb kommt ihrer Untersuchung eine Schlüsselrolle für die Analyse von Gehirnfunktionen zu. Um die Funktionalität des Gehirns besser zu verstehen, ist es erforderlich, nachzuvollziehen, welche Möglichkeiten, miteinander in Verbindung zu treten, die einzelnen funktionellen Einheiten des Gehirns haben. Dabei ist die zentrale Frage: Wie sind diese Verbindungsnetzwerke organisiert?

Faßt man die Areale als Knoten und die Verbindungen als Kanten eines gerichteten Graphen auf, so entsteht ein Netzwerk von beachtlicher Komplexität. (Eine graphische Darstellung eines Parzellierungsschemas sowie der zugehörigen Verbindungsstruktur ist in Abbildung 1.1 zu sehen.) Da Informationsweiterleitung normalerweise mit der Möglichkeit einer Rückmeldung gepaart ist, sind die resultierenden Netzwerke in der Regel stark zusammenhängend.

In dieser Arbeit werden grundlegende Fragen der Organisation interregionaler Verbindungsstrukturen mit Methoden der theoretischen Informatik erforscht. Dafür werden Definitionen der neurologischen Fragestellungen formuliert, die mathematisch eine solide Grundlage bilden, um dann effiziente Algorithmen zur Lösung zu entwickeln.

Die Arbeit läßt sich anhand der untersuchten Fragen in vier Bereiche einteilen.

Nach einer kurzen Einführung in die Neurologie in Kapitel 2 und in die Graphentheorie in Kapitel 3 gehen wir in Kapitel 4 der Frage nach, was für einen Einfluß eine einzelne Hirnregion auf die Zusammenhangsstruktur des gesamten Netzwerkes hat. Diese Information könnte hilfreich sein, um die funktionalen Konsequenzen einer Läsion zu verstehen.

Wir messen den Einfluß eines Knotens auf die Zusammenhangsstruktur eines gerichteten Graphen $G = (V_G, E_G)$ durch eine Funktion ϕ , die aus der Shapleywerttheorie stammt, welche wiederum ursprünglich innerhalb der Spieltheorie entwickelt wurde [Sha53]. Unsere Funktion ϕ wird durch eine sogenannte *charakteristische Funktion* parametrisiert, die mit f_G bezeichnet wird. Sie zählt für eine Menge von Knoten $V' \subseteq V_G$ die Anzahl der starken Zusammenhangskomponenten in dem Teilgraphen von G , der durch die Knoten von V' induziert wird. Allgemein bezeichnet man für eine gegebene Menge von abstrakten Objekten N eine Abbildung f aus der Potenzmenge von N in die reellen Zahlen \mathbb{R} als charakteristische Funktion, wenn

wenn es keinen Kreis ohne Chord gibt, der durch u und v läuft. Die Erweiterung des Separationstheorems auf ungerichtete Graphen erlaubt es, die Bewertung $\phi_{f_G}(u)$ in Linearzeit zu berechnen, wenn G cycle-composed oder chordal ist, und in Polynomzeit, wenn G bipartit distanzerhaltend ist.

Zum Abschluß des Kapitels werden in Abschnitt 4.6 Bewertungen für die Areale des visuellen Systems sowie des präfrontalen Cortex berechnet und die Ergebnisse diskutiert.

Im allgemeinen wird davon ausgegangen, daß die Informationsverarbeitung im zerebralen Cortex, insbesondere im visuellen System, hierarchisch organisiert ist. In dieser Hierarchie sollen auf den unteren Ebenen einfache Merkmale einer Information erkannt werden, welche dann auf den höheren Stufen zu einem komplexeren Bild zusammengefügt werden. Allerdings ist der genaue Aufbau der Hierarchie, insbesondere für die höheren Hierarchieebenen, unklar [FE91].

In Kapitel 5 wird die hierarchische Struktur in funktionellen Systemen untersucht. Es wird ein Ansatz vorgestellt, wie anhand von Kantenbewertungen, die als Abstand zwischen den inzidenten Knoten in der Hierarchie aufgefaßt werden, in gerichteten Graphen mit Hilfe von Optimierung Hierarchien auf den Knoten berechnet werden können. Dabei wird die Abweichung der Hierarchie von der Kantenbewertung minimiert. Eine weitere wesentliche Neuerung ist, daß die berechneten Hierarchiestufen auf einer kontinuierlichen Skala liegen und nicht, wie bei den bisher in der Neurologie vorgestellten Hierarchien, auf einer Skala mit diskreten Stufen. Dies erlaubt eine größere Flexibilität der Abstände zwischen zwei Stufen, insbesondere dort, wo die Hierarchie flacher wird.

Es wird also für Knoten in einem gerichteten Graphen $G = (V_G, E_G)$ mit Kantenbewertung $f : E_G \rightarrow \mathbb{R}$ eine Abbildung $l : V_G \rightarrow \mathbb{R}$ aus der Knotenmenge in die reellen Zahlen gesucht, die die induzierte Hierarchiestruktur des Graphen möglichst gut repräsentiert. Für eine Kante $(u, v) \in E_G$ sollte demnach gelten $l(u) + f((u, v)) = l(v)$. Wenn diese Bedingung nicht für alle Kanten erfüllt werden kann, so sollte zumindest eine Lösung gefunden werden, bei der die Abweichung möglichst gering ist. Es soll dann für alle Kanten gelten $l(v) - \Delta \leq l(u) + f((u, v)) \leq l(v) + \Delta$ für ein minimales $\Delta \geq 0$. Ziel ist es dann nicht nur, eine Abbildung l , sondern auch das kleinste Δ zu bestimmen, das diese Bedingung erfüllt. Da in den neurologischen Daten in der Regel keine exakten Abstandsangaben vorhanden sind, ist es eine weitere interessante Variante des Problems als Kantengewichte auch Intervalle zuzulassen, in denen der Abstand liegen soll. Daraus ergibt sich eine Kantenbewertung $f : E_G \rightarrow I(\mathbb{R})$ mit $I(\mathbb{R}) = \{[a, b] | a, b \in \mathbb{R}\}$ und entsprechend die Bedingung $l(v) - l(u) \in f((u, v))$ für alle Kanten (u, v) . Sollte sich keine exakte Lösung ergeben, so kann man auch hier die Intervallgrenzen um ein möglichst kleines Δ verschieben.

In Abschnitt 5.2 werden für die verschiedenen Varianten des Problems algorithmische Lösungen untersucht. Diese arbeiten auf erweiterten Versionen des Graphen $G' = (V, E')$, $E' = \{(u, v) | (u, v) \in E \text{ oder } (v, u) \in E\}$, in denen zu jeder Kante auch die Kante in umgekehrter Richtung existiert und die Kantenbewertung entsprechend ergänzt wird. Damit die Kantenbewertung eine Hierarchie induziert, ist es notwendig, daß alle Wege zwischen zwei Knoten dieselbe Länge haben. Insbesondere beschreibt ein Graph keine Hierarchie, wenn in dem zugehörigen erweiterten Graphen Kreise positiver Länge vorhanden sind.

In Abschnitt 5.3 wird gezeigt, wie sich die vorgestellten Hierarchie-Probleme mit Hilfe von li-

naerer Optimierung lösen lassen. Dieses Verfahren wird in Abschnitt 5.4 genutzt, um anhand von verschiedenen Datensätzen konkrete Hierarchien für das visuelle System des Makaken zu berechnen. Lineare Optimierungsprobleme sind theoretisch in polynomieller Zeit lösbar [Kar84].

Ein weiteres Prinzip der Organisation des visuellen Systems ist die parallele Verarbeitung verschiedener Aspekte der Information. Allgemein wird davon ausgegangen, daß die visuelle Informationsverarbeitung parallel organisiert ist und visuelle Informationen gleichzeitig getrennte Verarbeitungswege („Streams“) für räumliche und Objektinformationen durchlaufen, bevor das Ergebnis zu einem Gesamteindruck integriert wird.

In Kapitel 6 wird der Frage nachgegangen, wie man diese getrennten Verarbeitungswege mit Hilfe der Verbindungsstruktur des visuellen Systems identifizieren kann. Streams stellen ein intuitiv einleuchtendes, aber graphentheoretisch schwer zu definierendes Konzept bei der Informationsverarbeitung im Gehirn dar. Was kann in der Topologie des Gehirns darauf hinweisen, daß verschiedene Aspekte einer Information parallel verarbeitet werden? Es gibt zu dem Thema Streams verschiedene Studien, aber bisher hat nur Young [You93] versucht, auch topologische Anzeichen für Streams zu finden. Es soll in diesem Kapitel versucht werden, eine auf der Verbindungsstruktur basierende Definition von Streams über die Graphenstruktur des visuellen Systems zu finden. Dazu werden verschiedene Definitionsmöglichkeiten diskutiert und es wird untersucht, welche Ergebnisse diese auf den auch von Young verwendeten Verbindungsdaten des visuellen Systems bringen.

Als Basis für alle Untersuchungen werden in Abschnitt 6.3 zuerst einmal Kriterien für Streams aufgestellt, an denen auch Ergebnisse gemessen werden können. Dann werden Zielareale mithilfe einer Breitensuche ausgehend vom Areal V1 bestimmt. Um die restlichen Areale, die weder Start- noch Zielareal sind, in zwei Streams einzuteilen, werden drei verschiedene Ansätze verfolgt. Bei dem ersten Ansatz werden die Areale so in zwei Gruppen eingeteilt, daß die Anzahl der Verbindungen zwischen den beiden Gruppen minimal wird (Abschnitt 6.5.1). Der zweite Ansatz in Abschnitt 6.5.2 verwendet die Hamming-Distanz der Adjazenzen, um die Ähnlichkeit der Verbindungsstruktur von Arealen zu messen. Die Areale werden dann wieder in Gruppen eingeteilt, so daß die Hamming-Distanzen zwischen den Gruppen maximal werden. Dies minimiert gleichzeitig die Hamming-Distanzen innerhalb der beiden Gruppen. Zum Schluß wird noch die Anzahl der Informationswege vom Startareal V1 zu den Zielarealen als Differenzierungskriterium verwendet. Um zu verhindern, daß dabei unnötig lange Wege oder gar solche mit Kreisen berücksichtigt werden, beschränken wir uns in Abschnitt 6.5.3 auf kürzeste Wege vom Startareal V1 zu einem der Zielareale und nehmen in Abschnitt 6.5.4 noch diejenigen Wege hinzu, die um eins länger sind als kürzeste Wege. Dazu wird ein Algorithmus vorgestellt, der für jede Kante die Anzahl von kürzesten Wegen zwischen Start- und Zielarealen ermittelt, die über diese Kante laufen.

Trotz dieser unterschiedlichen Methoden der Einteilung von Streams ergeben sich nicht die erwarteten Ergebnisse. Insgesamt scheinen alle Kriterien eher zwischen anterioren und posterioren Arealen zu unterscheiden, als zwischen einem ventralen und einem dorsalen Stream. Youngs These, daß sich die Streams auch in der Verbindungsstruktur des visuellen Systems widerspiegeln, kann mit diesen Methoden nicht bestätigt werden.

Die bisher vorgestellten Analysen basieren zu großen Teilen auf den wenigen Zusammenstel-

lungen, die es für Verbindungsdaten von funktionellen Systemen gibt, wie zum Beispiel die Arbeiten von Felleman und Van Essen [FE91] oder Young [You92]. Wegen der zahlreichen Parzellierungsverfahren für den Cortex, auch Hirnkarten genannt, sind die Ergebnisse neurologischer Studien nur mit viel Aufwand und Handarbeit von Experten vergleichbar. Für viele Hirnregionen ist es nämlich noch ungeklärt, in welchem Verhältnis die vorhandenen Einteilungen zueinander stehen. Dies macht es schwer, die reichhaltigen Forschungsergebnisse wirklich zu nutzen.

In Kapitel 7 wird ein Verfahren entwickelt, um zu entscheiden, in welchem Verhältnis zwei Areale aus verschiedenen Hirnkarten zueinander stehen: Sind sie identisch, ineinander enthalten, überlappend oder disjunkt? Grundlage dieser Entscheidung bildet eine Darstellung der Hirnkarten als Rasterpunkte auf der Cortexoberfläche. Es wird also keine vollständige Oberflächendarstellung verwendet, sondern es wird für ein Raster von Punkten auf der Cortexoberfläche angegeben, welche Punkte sich in welchem Areal einer bestimmten Karte befinden. Hierbei ist zu beachten, daß die Hirnkarten von Hand auf die Cortexoberfläche projiziert wurden. Dadurch kommt es zwangsläufig zu Ungenauigkeiten; so unterscheiden sich auch Punktemengen von identischen Arealen in verschiedenen Hirnkarten.

Der Ansatz, der hier gewählt wurde, um zwei Areale zu vergleichen, verwendet die relative Größe des gemeinsamen Schnittes in den beiden Arealen, also jeweils das Verhältnis der Anzahl der Punkte im Schnitt zu der Anzahl der Punkte im Areal. Ist diese für ein Areal nahe bei 1, so ist fast das ganze Areal in dem Schnitt enthalten. Ist die relative Größe des Schnitts für ein Areal hingegen nahe bei 0, so nimmt der Schnitt nur einen sehr kleinen Teil des Areals ein. Bei einem solchen Vergleich von zwei Arealen erhält man also mit der relativen Schnittgröße für jedes der beiden Areale einen Wert in dem reellen Intervall $[0, 1]$ und damit zu jedem Vergleich einen Wert in $[0, 1] \times [0, 1]$.

In Abschnitt 7.4.1 werden Überlegungen angestellt, welche Verteilungen dieser Werte für die verschiedenen Beziehungsarten in dem Quadrat $[0, 1] \times [0, 1]$ zu erwarten sind. Um diese zu überprüfen, wurden die relativen Größen für Arealpaare berechnet, deren Beziehung schon aus der CoCoMac-Datenbank bekannt war. Damit ergibt sich eine Verteilung von Punkten, die den verschiedenen Beziehungsarten zugeordnet sind. Basierend auf theoretischen Überlegungen werden mit Hilfe eines Optimierungsverfahrens Bereiche für die verschiedenen Arealrelationen eingeteilt, die die Punkteverteilung möglichst gut repräsentieren.

In einer Erweiterung dieses Ansatzes wird in Abschnitt 7.4.2 eine Gewichtung der Punkte eines Areals definiert, um die ungenauen Grenzen der Areale zu kompensieren. Dabei erhalten Punkte der Areale mehr Gewicht, je weiter sie im Inneren liegen. Dadurch erhält auch eine Schnittmenge, die im Zentrum eines Areals liegt, mehr Gewicht als eine gleichgroße Schnittmenge, die im Randbereich eines Areals liegt. Es wird ein Algorithmus vorgestellt, der diese Gewichtung effizient berechnet. Auch für diese Daten wird eine Einteilung in Bereiche für die Arealrelationen berechnet. Dabei wird dasselbe Verfahren wie zuvor verwendet.

Zum Abschluß wird in Abschnitt 7.4.3 noch ein alternatives Konzept zur Einteilung der Bereiche für die Verbindungsarten vorgestellt. Auch für dieses Schema werden die entsprechenden Bereiche, sowohl für die gewichteten als auch für die ungewichteten Daten, mit Hilfe einer Optimierung berechnet. Die Ergebnisse deuten nicht nur auf deutliche Widersprüche in den verwendeten Daten hin, sondern geben auch klare Anhaltspunkte dafür, wo diese liegen. Die verwendeten Verfahren können also helfen, diese Widersprüche aufzuspüren und sie zu charakterisieren.

Die in dieser Arbeit behandelten Themen bieten sowohl neue Fragestellungen für die Informatik als auch konkrete Anwendungen in der Neurologie. Der Fokus soll hier auf der theoretischen Aufarbeitung und Darstellung der informatischen Methoden liegen. In jedem Kapitel werden aber auch Berechnungen durchgeführt, die Relevanz für die Neurologie haben.

Kapitel 2

Grundlegende Begriffe der Neurologie

Dieses Kapitel soll einen groben Überblick über die in dieser Arbeit verwendeten neurologischen Begriffe und Konzepte bieten. Eine ausführliche Einführung in die Neurologie findet man zum Beispiel in dem Buch von Zilles und Rehkämper [ZR98].

Die Informationsverarbeitung im Gehirn erfolgt durch Neuronen. Diese können bei Stimulation über Fortsätze elektrische Impulse an andere, auch weit entfernt liegende Zellen weitergeben. Die Aufgabe der Neuronen ist also die Erregungsleitung und -verarbeitung. Dieser Prozeß bildet die Grundlage für alle Funktionen des Gehirns. Allerdings ist die Zahl der Neuronen und ihrer Verbindungen so groß, daß man dieses Gesamtnetzwerk mit heutigen Methoden nicht erfassen.

Im zerebralen Cortex, der auch Hirnrinde genannt wird, bilden Gruppen von nahe beieinanderliegenden Zellen Einheiten, die sich zum Beispiel durch ihre Architektur von anderen Zellverbänden in ihrer Umgebung unterscheiden. Häufig verfügen diese Einheiten auch über ähnliche Verbindungsstrukturen, insbesondere zu weit entfernten Hirnregionen. Dadurch entstehen zwischen den einzelnen Regionen Faserbahnen, die diese verbinden. Andersherum betrachtet kann man auch Zellverbände mit gleichen Verbindungsstrukturen zu einem Cluster zusammenfassen und darüber eine Einheit definieren. Der zerebrale Cortex läßt sich also in Bereiche einteilen, deren Neuronen eine funktionale Einheit darstellen. Unabhängig von den Kriterien für die Einteilung werden Zellverbände als Hirnregionen oder Areale bezeichnet. Kortikale Felder oder Areale sind normalerweise in jedem einzelnen Gehirn, unabhängig von der Größe oder Form, einer Spezies vorhanden. Deshalb bieten sie eine effiziente Beschreibung der Gehirnarhitektur.

Der Cortex besteht also aus vielen funktionell und strukturell verschiedenen Arealen, zwischen denen es eine komplexe Verbindungsstruktur gibt. Die Rolle, die Hirnregionen innerhalb dieses Netzwerkes haben, wird durch ihre jeweiligen Verbindungen definiert. Durch afferente Verbindungen wird der Input bestimmt, der erhalten werden kann, und durch efferente Verbindungen wird die verarbeitete Information an bestimmte Ziele weitergeleitet. Dabei zeigt jedes Areal eine individuelle Verbindungsstruktur [PSK02].

2.1 Parzellierungsschemata

Einer der ersten Autoren, die den Cortex in Regionen unterteilten war Brodmann [Bro05, Bro09]. Er verwendete dazu cytoarchitektonische Unterscheidungsmerkmale. Sein Ziel war ein vergleichendes System von Organen im zerebralen Cortex zu erstellen, das auf anatomischen Charakteristika basiert. Um es zu erreichen, nahm er eine regionale Unterteilung des Cortex in strukturelle kortikale Felder vor, die eine Einteilung in homogene und untereinander unverwechselbare Regionen der hemisphärischen Oberfläche darstellen sollte. Diese Regionen werden *Areae anatomicae* genannt.

Auch heute bildet die Kartierung des Gehirns mit seinen vielen strukturellen und funktionalen Aspekten die Grundlage der Bemühungen, zu verstehen, wie sich das Gehirn entwickelt, arbeitet und anpaßt. Bei einer Parzellierung des Cortex werden im Allgemeinen Gruppen von benachbarten Neuronen zu Arealen zusammengefaßt. Es gibt verschiedene Vorgehensweisen, um diese Gruppierungen vorzunehmen. Die ersten Ansätze orientierten sich, wie der von Brodmann, an mikrostrukturellen Unterschieden im Aufbau des Cortex. Modernere Ansätze beziehen auch funktionale Kriterien mit ein. Drei Beispiele für Parzellierungen sind in Abbildung 2.1 zu sehen. Die Parzellierungen des Cortex werden im folgenden auch Hirnkarten genannt.

Ziel der Einteilung des Cortex in Regionen ist es, funktionell zusammenhängende Strukturen zu finden. Die Untersuchung dieser Strukturen und ihrer Verbindungen soll helfen, die Architektur und funktionelle Organisation des zerebralen Cortex zu erschließen.

Die Einteilung des Cortex in zusammengehörige Regionen ist aber bei weitem nicht eindeutig. Wegen der verschiedenen angewandten Methoden und der hohen Beobachterabhängigkeit vieler mikrostrukturell-funktionaler Kriterien existiert eine große Anzahl von Parzellierungen, die sich in Nomenklatur, Anzahl, Position, Größe, Grenzen und Darstellung der beschriebenen Gehirnstrukturen unterscheiden [FE91, KW05, Zil04].

Verschiedene Hirnkarten sind somit oft inkompatibel in Hinsicht auf die Grenzen der beschriebenen Gehirnstrukturen und der Nomenklatur. Hinzu kommen noch die interindividuelle Variabilität der Gehirnform sowie interindividuelle Unterschiede in der Größe und Position der Gehirnstrukturen, die das eindeutige Identifizieren von Strukturen häufig schwierig gestalten. Diese Umstände können auch den Vergleich von Daten aus zwei verschiedenen Parzellierungsschemata sehr schwer machen. Für viele Hirnregionen ist nicht geklärt, wie sich die existierenden Einteilungen zueinander verhalten. Ebenso bleibt häufig fraglich, welche Einteilung optimal ist, sei es auch nur in einem bestimmten Kontext. Ein allgemein anerkanntes System ist also nicht in Sicht.

2.2 Funktionelle Systeme

Ein Netzwerk, das aus allen Arealen des Cortex besteht, ist für eine eingehende Analyse häufig zu komplex. Deshalb wird selten der ganze Cortex betrachtet, sondern Arealgruppen werden zu funktionellen Systemen zusammengefaßt. Wie der Name schon sagt, ist hängt diese Zuordnung davon ab, in welchem funktionellen Zusammenhang ein Areal aktiv wird. Im Primatencortex unterscheidet man unter anderem zwischen dem auditiven, dem visuellen und dem motorischen System. Eine mögliche schematische Darstellung der Organisation

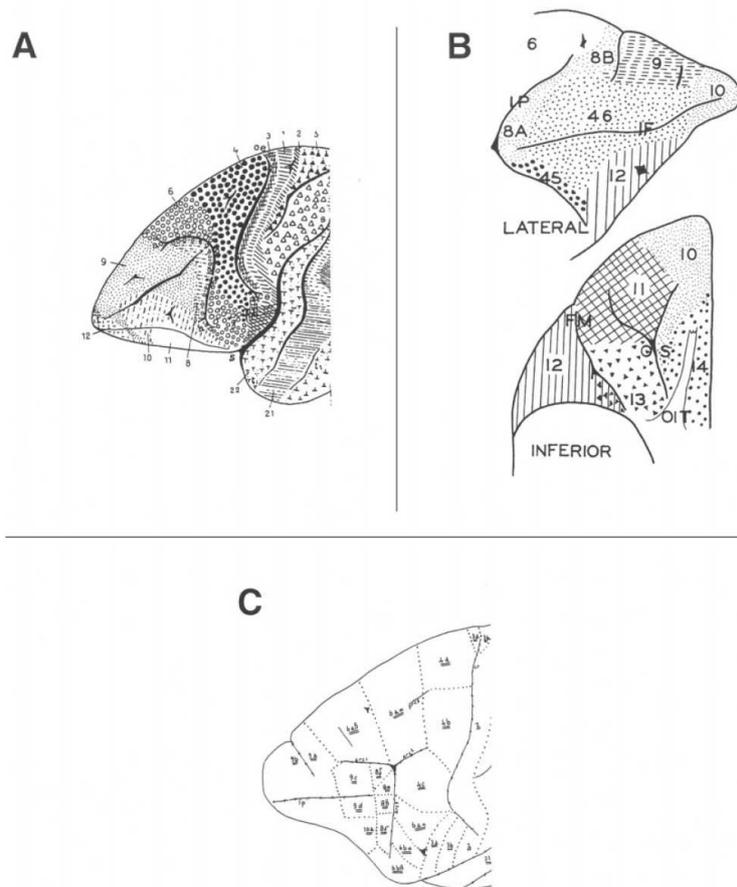


Abbildung 2.1: Parzellierungsschemata des präfrontalen Cortex des Makaken nach Brodmann (A) [Bro05], Walker (B) [Wal40] und Vogt & Vogt (C) [VV19]. (Abbildung entnommen aus Petrides und Pandya [PP01])

des Cortex in funktionelle Systeme sowie des Informationsaustausch zwischen den Systemen ist in Abbildung 2.2 zu sehen. Die Areale der verschiedenen Systeme sind jeweils über den Cortex verteilt und nicht etwa nur in einer Region zu finden. Einige Areale insbesondere des präfrontalen Cortex können auch zu mehreren funktionellen Systemen gezählt werden.

Bei Verbindungsanalysen beschränkt man sich also in der Regel auf die Areale und Verbindungen innerhalb von einem funktionellen System. Auch diese relativ kleinen Netzwerke mit in der Regel maximal 30 Arealen sind schon zu umfangreich und komplex, um intuitiv verstanden zu werden. Das Ziel aufzuklären, was für einen Beitrag eine individuellen Hirnregion zu den globalen Eigenschaften eines Netzwerkes leistet, läßt sich innerhalb solcher Netzwerke aber leichter realisieren, da zumindest klar ist, mit welcher Aufgabe das Netzwerk befaßt ist. Bei solchen Analysen werden dann allerdings Verbindungen von Arealen, die zu verschiedenen funktionellen Systemen gehören, vernachlässigt. Der Vorteil bei der Untersuchung funktioneller Systeme ist also, daß man schon konkrete Informationen über die Aufgabe des gesamten Netzwerkes hat und das Netzwerk relativ überschaubar ist.

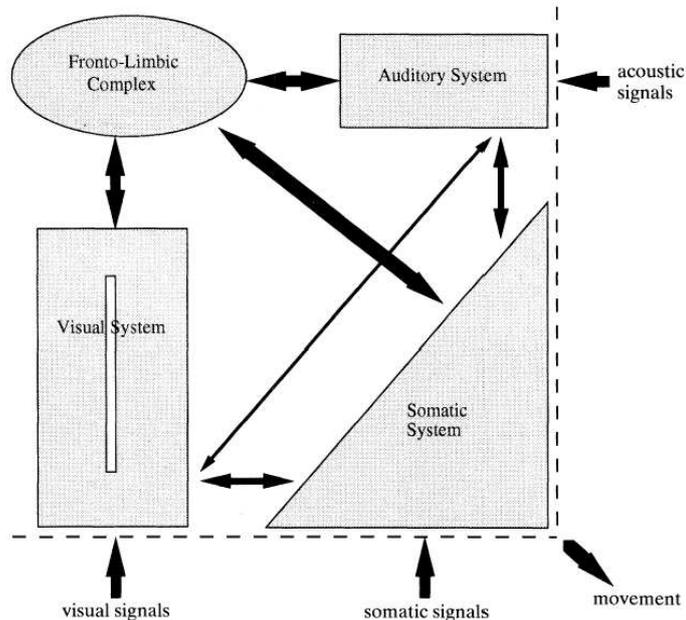


Abbildung 2.2: Organisation des Primaten Cortex nach Young entnommen aus [You93]

2.3 Tracing-Studien

Wie schon gesagt, definiert sich die Rolle eines Areals über seine Verbindungen zu anderen Arealen. Diese lassen sich aber nicht einfach erkennen, da die Faserbahnen im Gehirn durch bloßes Anschauen nicht zu trennen sind. Deshalb werden die Leitungseigenschaften der Verbindungen genutzt, um diese zu identifizieren. Dies passiert in sogenannten Tracing-Studien. Dabei wird in ein Areal, dessen Verbindung untersucht werden soll, eine Tracing-Substanz injiziert. Diese breitet sich dann im Gehirn entweder anterograd (in Richtung des Informationsflusses) oder retrograd (entgegen des Informationsflusses) aus. Nach einer angemessenen Überlebensperiode, die den Transport des Tracers sicher stellen soll, wird die Verteilung des Tracers im Gehirn untersucht. Alle Areale, die mit dem Tracer gefärbt wurden, haben also entweder Verbindungen zu dem Ausgangsareal (wenn ein retrograder Tracer verwendet wurde) oder es gibt Verbindungen vom Ausgangsareal zu diesen Arealen (wenn ein anterograder Tracer verwendet wurde). Ein Überblick über Methoden des neuroanatomischen Tracings sowie über verwendete Tracing-Substanzen findet man bei Köbbert et al. [KAB⁺00]. Eine graphische Darstellung des Ergebnisses eines Tracing-Experiment ist in Abbildung 2.3 zu sehen.

Je nach Studie wird aber nicht nur das Auftreten, sondern auch die Verteilung des Tracer in den markierten Arealen beschrieben. Dabei wird zuerst einmal registriert, ob das gesamte Areal oder nur Teile des Areals eingefärbt wurden. Außerdem wird häufig beschrieben, wie deutlich die Färbung war. Die Intensität eines Tracers wird auch als Indikator für die Verbindungsstärke angesehen, je mehr von dem Tracer zu sehen ist, desto stärker ist die Verbindung.

Es wird natürlich auch nicht nur die Oberfläche des Cortex betrachtet. Die Hirnrinde ist ein

Da die Parzellierungsschemata in der Regel nicht auf Verbindungsstrukturen basieren, ist auch bei einer vorhandenen Verbindung zwischen zwei Arealen nicht klar, ob diese sich über Neuronen des ganzen Areals erstreckt. Möglicherweise hat nur eine Teilregion des untersuchten Areals eine Verbindung zur Zielregion. Wird der Tracer in den anderen Teil des Areals injiziert, so läßt sich keine Verbindung nachweisen. Siehe Abbildung 2.4 für eine schematische Darstellung dieser Situation. Injektionen, die sich nicht über das ganze Areal erstrecken, können also zu scheinbar widersprüchlichen Ergebnissen bei Tracing-Studien führen.

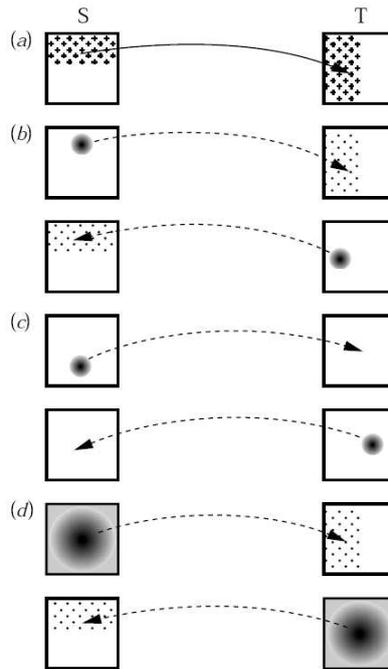


Abbildung 2.4: Beispiele für teilweise und vollständige Tracing-Injektionen, die zeigen, wie verschiedenen Injektionen in dasselbe Areal zu scheinbar widersprüchlichen Ergebnissen führen können. (Abbildung entnommen aus [SKB⁺01]). (a) die tatsächliche Art der Verbindung zwischen zwei hypothetischen Arealen. Nur ein Teil des Startareal (S) hat Verbindungen zum Zielareal (T). (b) Wenn anterograde und retrograde Injektionen nur in den Teilen appliziert werden, die in Verbindung stehen, dann deuten die Resultate auf eine Verbindung hin. (c) Werden anterograde und retrograde Injektionen in den anderen Teilen appliziert, so kommt man fälschlicherweise zu dem Ergebnis, daß es keine Verbindungen zwischen den Arealen gibt. (d) Dagegen liefern anterograde und retrograde Injektion, die sich über das ganze Gebiet des Start- beziehungsweise Zielareals ausdehnen, sichere Informationen.

Alle angeführten Probleme können zu widersprüchlichen Angaben über Verbindungen zwischen Arealen führen. Somit ist die Datenlage im Allgemeinen nicht eindeutig. Ein weiteres Problem für die Analyse von Verbindungsstrukturen ist, daß die vorhandenen Tracing-Studien nicht unbedingt alle möglichen Verbindungen abdecken. So ist bei der aktuellen Datenlage nicht immer klar, ob eine Verbindung nicht existiert oder ob sie nur noch nicht untersucht wurde. Dies ist insbesondere dann der Fall, wenn ein Areal aufgrund von neuen Erkenntnissen weiter unterteilt wird und für diese neue Unterteilung noch keine Untersuchung vorliegt. Da der Datenbestand aber schon sehr umfassend ist, wird im Allgemeinen bei Analysen ange-

nommen, daß nicht bekannte Verbindungen auch nicht existieren.

Alle diese Faktoren erschweren eine Analyse der Verbindungsstruktur. Wegen des hohen Aufwands der Tracing-Studien (und weil jedes Mal ein Affe sterben muß), können aber auch nicht so viele Daten erhoben werden, daß diese Unklarheiten verschwinden.

2.4 Die CoCoMac-Datenbank

Das ultimative Ziel von Tracing-Studien ist eine vollständige Darstellung des Kommunikationsnetzwerk des Gehirns, das durch seine Faserverbindungen errichtet wird. Der derzeitige Stand dieser Bemühungen steht aber nur in Form von Fragmenten zur Verfügung, die über tausende von Berichten in der Literatur verstreut sind. In dieser Arbeit werden deshalb vielfach Daten aus der CoCoMac-Datenbank ([SKB⁺01, KÖ4], <http://cocomac.org>) verwendet. Diese Datenbank ist eine umfangreiche Ressource, die literaturbasierte anatomische Kartierungen sowie publizierte experimentelle Primärdaten von hunderten von Studien über das Makakengehirn enthält. Ursprünglich stand CoCoMac für „cortical connectivity of the Macaque brain“. Da aber inzwischen auch subcortikale Verbindungen verzeichnet sind, wurde der Name zu „collation of connectivity data on the macaque brain“ umgedeutet.

Die Beschreibungen anatomischer Projektionen sind komplexe Daten. Zusätzlich zu Start- und Zielarealen sind noch andere Eigenschaften einer Verbindung von Interesse, zum Beispiel Stärke, laminare Muster und andere räumliche Verteilungen von sendenden und empfangenden Neuronen, oder quantitative Daten wie die absolute Anzahl von Neuronen. Dabei gibt es keinen Standard, gemäß dem anatomische Projektionen und ihre vielen Eigenschaften repräsentiert werden sollten. Ein weiteres Problem ist die schon angesprochene Vielzahl der Gehirnkarten. Um diese Probleme zu behandeln, wurde die CoCoMac-Datenbank nach strengen Richtlinien erstellt. Dabei wurden insbesondere die folgenden Punkte berücksichtigt:

1. objektive Darstellung von koordinatenunabhängigen, parzellierungsbasierten Daten
2. Einschätzung der Zuverlässigkeit und Genauigkeit der einzelnen Daten, insbesondere bei widersprüchlichen Forschungsergebnissen
3. Datamining und Integration von großen Mengen teilweise redundanter und widersprüchlicher Daten
4. automatische und reproduzierbare Transformation von Daten zwischen inkongruenten Hirnkarten

Der erste Punkt wird durch standardisierte Protokolle für die Datenextraktion und Dateneingabe realisiert. Dabei werden nur einzelne Ergebnisse und keine zusammengefaßten oder interpretierten Daten verwendet. Außerdem sind die Daten genau wie vom Autor beschrieben dargestellt, also insbesondere in seinem eigenen Parzellierungsschema. Zu jeder Quelle findet man bibliographische Daten, Kartierungsdaten und Experiment-bezogene Daten.

Der zweite Punkt wurde durch die Einführung von PDC-Werten umgesetzt. Dabei steht PDC für „Precision of Description Coding“; es wird mit diesem Wert also die Genauigkeit der Beschreibung in einer Literaturquelle beurteilt. Diese Bewertung der Daten wird ebenfalls nach einer standardisierten Prozedur ermittelt. Dabei wird nicht ein Wert für das ganze

Experiment vergeben, sondern fünf verschiedene Aspekte der Daten bewertet. Diese sind die Identifikation der Areale, die Ausbreitung des Tracers innerhalb eines identifizierten Areals, die Dichte oder Stärke der Markierung innerhalb eines identifizierten Areals, das laminare Muster der Markierung innerhalb eines identifizierten Areals sowie, unabhängig von konkreten Tracing-Experimenten, das Verhältnis von zwei Arealen in zwei verschiedenen Gehirnkarten. Insgesamt wird so die Präzision aller Datentypen bewertet. Dabei ist es durchaus möglich, daß verschiedene Aspekte der Information aus einer Quelle unterschiedliche PDC-Bewertungen erhalten.

Der dritte Punkt wird durch die hohe Flexibilität der Suche möglich. Es kann eine große Bandbreite von Daten gesucht und für viele Zwecke analysiert werden. So hat man unter anderem die Möglichkeit nach exakten Referenzen, experimentellen Protokollen, injizierten und gekennzeichneten Arealen, laminaren Mustern und Dichte der Markierung, quantitativen Daten aber auch Parzellierungsschemata und Beziehungen zwischen Gehirnkarten zu suchen. Es gibt auch verschiedene Einschränkungsmöglichkeiten für die Suche, wie zum Beispiel verwendete Tracer-Substanzen, rechte oder linke Hemisphäre, ipsi- oder contralaterale Verbindungen.

Der letzte Punkt beschäftigt sich nicht direkt mit der Dateneingabe sondern dem Problem inkongruenter Einteilungen von Hirnstrukturen (Parzellierungsproblem). Es sollen die Daten aus verschiedenen Studien besser vergleichbar gemacht werden. Um dies umzusetzen, wurde ein Algorithmus genannt „Objective Relational Transformation“ (ORT, [KW05, SKB⁺01]) implementiert. Dieser verwendet elementare logische Aussagen und Verkettungsregeln, um komplexere Aussagen abzuleiten, die die Beziehung zwischen Gehirnkarten und dargestellten Gehirnarealen beschreiben. Für jeweils zwei Areale aus verschiedenen Hirnkarten entscheidet der Algorithmus, ob diese identisch, ineinander enthalten, oder überlappend sind. Es handelt sich dabei um ein beobachterunabhängiges Verfahren zur Vergleichbarmachung von Originaldaten aus verschiedenen Hirnkarten.

The screenshot displays the CoCoMac database interface. The top section shows a search results page for 'Connectivity output list, PrimaryProjections' with 19 items. Below this is a table listing various projections with columns for Item, SourceSite, PDC, Hemisph, Density, PDC, Course, TargetSite, PDC, Hemisph, and Laminar. The bottom section shows a detailed view of a specific connection, including a citation: 'Corticostriatal connections of extrastriate visual areas in rhesus monkeys' by Pandya, DN (1985), and a table of 'Labelled Sites for FV91-V2v' with columns for Site, PDC, Hemisph, Extent, PDC, Density, PDC, Label type, Lam. pattern, PDC, Ref. text, and Ref. figures.

Abbildung 2.5: Ein Screenshot der CoCoMac-Datenbank

Kapitel 3

Grundlegende Begriffe der Graphentheorie

Ein *Graph* $G = (V, E)$ ist ein Paar bestehend aus einer Menge von *Knoten* $V = \{u_1, \dots, u_n\}$ und einer Menge E von *Kanten*. Wir betrachten in dieser Arbeit ausschließlich *endliche Graphen*, das heißt, sowohl die Knotenmenge als auch die Kantenmenge enthalten nur endlich viele Elemente.

In einem *gerichteten Graphen* ist jede Kante $e \in E$ ein geordnetes Paar (u, v) von zwei Knoten $u, v \in V$, u ist der *Startknoten* und v der *Zielknoten* der Kante (u, v) . Es gilt also $E \subseteq V \times V$.

In einem *ungerichteten Graphen* ist jede Kante $e \in E$ eine Menge $\{u, v\}$ von zwei verschiedenen Knoten $u, v \in V$, u und v sind *Endknoten* der Kante $\{u, v\}$. Es gilt also $E \subseteq \{\{u, v\} | u, v \in V, u \neq v\}$.

Grafisch stellt man die Knoten der Graphen als Kreise oder Punkte dar. Eine gerichtete Kante (u, v) zeichnet man als Pfeil von u nach v . Eine ungerichtete Kante zeichnet man als verbindende Linie zwischen u und v . Beispiele für einen gerichteten und einen ungerichteten Graphen sind in Abbildung 3.1 zu sehen.

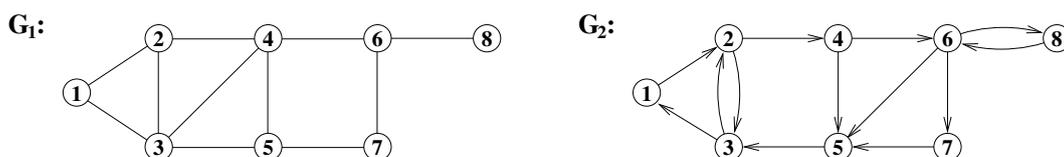


Abbildung 3.1: Links ein Beispiel für einen ungerichteten Graphen G_1 mit 8 Knoten und rechts ein Beispiel für einen gerichteten Graphen G_2 mit 8 Knoten.

Sei $e = (u, v)$ beziehungsweise $e = \{u, v\}$ eine Kante in einem Graphen. Die Knoten u und v sind zueinander adjazent und mit Kante e inzident.

Ein Graph $G' = (V', E')$ ist ein *Teilgraph* von G , falls $V' \subseteq V$ und $E' \subseteq E$.

Zwei Graphen $G = (V, E)$ und $J = (V', E')$ sind *isomorph*, falls es eine Bijektion $b : V \rightarrow V'$ gibt mit $(u, v) \in E \Leftrightarrow (b(u), b(v)) \in E'$, falls G und J gerichtet sind, bzw. $\{u, v\} \in E \Leftrightarrow \{b(u), b(v)\} \in E'$, falls G und J ungerichtet sind.

Eine *Graphklasse* ist eine Menge von Graphen. Spezielle Graphklassen entstehen durch Ein-

schränkungen der allgemeinen Graphdefinition.

3.1 Definitionen für ungerichtete Graphen

Sei $G = (V, E)$ ein ungerichteter Graph. Der *Knotengrad* eines Knotens $u \in V$, geschrieben $\deg(u)$, ist die Anzahl der mit u inzidenten Kanten. Die *Nachbarschaft* eines Knotens $u \in V$ ist die Menge $N(u) = \{v \in V \mid \{u, v\} \in E\}$. Ein Knoten $v \in N(u)$ heißt *Nachbar* von u .

$G' = (V', E')$ ist ein *induzierter Teilgraph* von G , falls $V' \subseteq V$ und $E' = E \cap \{\{u, v\} \mid u, v \in V', u \neq v\}$. Für eine Knotenmenge $V' \subseteq V$ ist $G|_{V'} = (V', E \cap (V' \times V'))$ der durch V' induzierte Teilgraph von G . Ein Beispiel ist in Abbildung 3.2 zu sehen.



Abbildung 3.2: Beispiele für induzierte Teilgraphen des Graphen G_1 aus Abbildung 3.1. Links der Teilgraph, der durch die Knotenmenge $\{v_1, v_2, v_3\}$ induziert wird und rechts der Teilgraph der durch die Knotenmenge $\{v_6, v_7, v_8\}$ induziert wird. Zusammen ergeben sie den Teilgraphen, der durch die Knotenmenge $\{v_1, v_2, v_3, v_6, v_7, v_8\}$ induziert wird.

Ein *ungerichteter Weg* p in G der Länge k von Knoten u nach Knoten w ist eine Folge von k Knoten $p = (v_1, \dots, v_k)$ mit $v_1 = u$ und $v_k = w$ und $\{v_{i-1}, v_i\} \in E$ für $1 < i \leq k$. Ein Weg p ist *einfach*, wenn alle k Knoten v_1, \dots, v_k verschieden sind. Ein (einfacher) ungerichteter *Kreis* der Länge $k \geq 3$ ist ein (einfacher) ungerichteter Weg $p = (v_1, \dots, v_k)$ der Länge k mit $\{v_k, v_1\} \in E$. Wir betrachten nur einfache Pfade und Kreise, bei denen alle Knoten verschieden sind.

Der *Abstand* zwischen zwei Knoten u, v ist die minimale Länge eines Weges zwischen u und v , wenn ein solcher existiert. Ansonsten ist der Abstand zwischen u und v nicht definiert.

In unserem ungerichteten Beispielgraphen G_1 aus Abbildung 3.1 ist $(v_1, v_2, v_3, v_4, v_5)$ ein Weg der Länge 5 von v_1 nach v_5 . (v_1, v_2, v_3) , (v_2, v_3, v_4) und (v_3, v_4, v_5) sind jeweils Kreise der Länge 3. Der Abstand zwischen v_1 und v_5 ist 3.

Ein *ungerichteter Wald* ist ein ungerichteter, kreisfreier Graph. Ein ungerichteter Wald in dem es zwischen jedem Knotenpaar einen Weg gibt, ist ein *ungerichteter Baum*. In einem ungerichteten Baum gibt es zwischen jedem Knotenpaar genau einen einfachen Weg.

Wir bezeichnen einen ungerichteten Graphen G als *zusammenhängend*, wenn es zwischen jedem Knotenpaar in G einen Weg gibt. Eine *Zusammenhangskomponente* von G ist ein maximaler zusammenhängender, induzierter Teilgraph von G .

Einen ungerichteten Graphen G , nennen wir *zweifach (k -fach) zusammenhängend*, wenn es zwischen jedem Knotenpaar zwei (k) knotendisjunkte Wege gibt. Dabei sind Wege *knoten-disjunkt*, wenn die von den Wegen besuchten Knoten alle (bis auf Start- und Endknoten) paarweise verschieden sind. Eine *k -fache Zusammenhangskomponente* von G ist ein maximaler k -fach zusammenhängender, induzierter Teilgraph von G .

Unser ungerichteter Beispielgraph G_1 aus Abbildung 3.1 ist zusammenhängend und bildet somit die einzige Zusammenhangskomponente von G_1 . Der Graph G_1 ist aber nicht zweifach zusammenhängend, jedoch bildet der Teilgraph von G_1 , der durch die Knotenmenge $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ induziert wird, eine zweifache Zusammenhangskomponente. Der durch die Knotenmenge $\{v_1, v_2, v_3, v_6, v_7, v_8\}$ induzierte Teilgraph von G_1 ist nicht zusammenhängend (siehe Abbildung 3.2).

3.2 Definitionen für gerichtete Graphen

Sei $G = (V, E)$ ein gerichteter Graph. Der *Eingangsgrad* bzw. *Ausgangsgrad* eines Knotens u , geschrieben $\text{indeg}(u)$ bzw. $\text{outdeg}(u)$, ist die Anzahl der in u einlaufenden bzw. auslaufenden Kanten. Der Grad eines Knotens u , geschrieben $\text{deg}(u)$, ist die Anzahl der mit u inzidenten Kanten ($\text{deg}(u) = \text{indeg}(u) + \text{outdeg}(u)$).

$G' = (V', E')$ ist ein *induzierter Teilgraph* von G , falls $V' \subseteq V$ und $E' = E \cap (V' \times V')$. Für eine Knotenmenge $V' \subseteq V$ ist $G|_{V'} = (V', E \cap (V' \times V'))$ der durch V' induzierte Teilgraph von G . Ein Beispiel ist in Abbildung 3.3 zu sehen.



Abbildung 3.3: Beispiele für induzierte Teilgraphen des Graphen G_2 aus Abbildung 3.1. Links der Teilgraph, der durch die Knotenmenge $\{v_1, v_2, v_3\}$ induziert wird und rechts der Teilgraph der durch die Knotenmenge $\{v_6, v_7, v_8\}$ induziert wird. Zusammen ergeben sie den Teilgraphen, der durch die Knotenmenge $\{v_1, v_2, v_3, v_6, v_7, v_8\}$ induziert wird.

Ein *gerichteter Weg* p in G der Länge k von Knoten u nach Knoten w ist eine Folge von k Knoten $p = (v_1, \dots, v_k)$ mit $v_1 = u$, $v_k = w$ und $(v_{i-1}, v_i) \in E$ für $1 < i \leq k$. Ein Weg p heißt *einfach*, wenn alle Knoten v_1, \dots, v_k verschieden sind. Ein (einfacher) gerichteter *Kreis* der Länge $k \geq 1$ ist ein (einfacher) gerichteter Weg $p = (v_1, \dots, v_k)$ der Länge k mit $(v_k, v_1) \in E$. In unserem gerichteten Beispielgraphen G_2 aus Abbildung 3.1 ist $(v_1, v_2, v_4, v_6, v_7, v_5)$ ein Weg der Länge 6 von v_1 nach v_5 . (v_1, v_2, v_3) ist ein Kreis der Länge 3.

Wir bezeichnen einen gerichteten Graphen G als *stark zusammenhängend*, wenn es zwischen jedem Knotenpaar in G einen Weg gibt. Eine *starke Zusammenhangskomponente* von G ist ein maximaler, stark zusammenhängender, induzierter Teilgraph von G .

Für einen gerichteten Graphen $G = (V, E)$ sei $G' = (V, \{(u, v) | (u, v) \in E \text{ oder } (v, u) \in E\})$. Man bezeichnet einen gerichteten Graphen G als *schwach zusammenhängend*, wenn G' stark zusammenhängend ist. Eine *schwache Zusammenhangskomponente* von G ist ein maximaler, schwach zusammenhängender, induzierter Teilgraph von G .

Der gerichtete Beispielgraph G_2 aus Abbildung 3.1 ist stark zusammenhängend und bildet die einzige starke Zusammenhangskomponente von G_2 . Damit ist G_2 auch schwach zusammenhängend. Der durch die Knotenmenge $\{v_6, v_7, v_8\}$ induzierte Teilgraph von G_2 ist ebenfalls stark zusammenhängend, während der durch die Knotenmenge $\{v_6, v_7, v_8\}$ induzierte Teilgraph von G_2 nur schwach zusammenhängend ist (siehe Abbildung 3.3). Der Teilgraph

von G_2 , der durch die Knotenmenge $\{v_1, v_2, v_3, v_6, v_7, v_8\}$ induziert wird, ist nicht zusammenhängend.

Eine *topologische Sortierung* eines gerichteten Graphen G mit n Knoten ist eine Anordnung der Knoten $h : V \rightarrow \{1, \dots, n\}$, so daß für jede gerichtete Kante gilt: $(u_i, u_j) \in E$, $h(u_i) < h(u_j)$. Eine solche kann es offensichtlich nur geben, wenn der Graph G kreisfrei ist. Ein Graph heißt *kreisfrei*, wenn er keinen Kreis enthält.

3.3 Darstellung von Gehirnetzwerken durch Graphen

Da wir in dieser Arbeit neurologische Fragen mit Hilfe der Graphentheorie untersuchen wollen, soll noch kurz darauf eingegangen werden, wie man die Netzwerke im Gehirn als Graphen darstellen kann. Auf der makroskopischen Ebene kann das Gehirn idealisiert als ein Netzwerk, das aus Hirnregionen und ihren Verbindungen besteht, betrachtet werden. Von einem graphentheoretischen Standpunkt aus werden Gehirnregionen als homogenen Knoten betrachtet, deren Kanten Verbindungen zwischen diesen Regionen ausdrücken. Da die Informationsweiterleitung im Gehirn immer eine Richtung hat, sind auch die Kanten gerichtet.

Wir betrachten ein Netzwerk im Gehirn also als gerichteten Graphen, bezeichnet mit $G = (V_G, E_G)$, dessen Knotenmenge V_G eine Menge von neuronalen Strukturen (Nuclei oder kortikale Regionen) und dessen gerichtete Kantenmenge E_G die Menge von unidirektionalen Verbindungen (Projektionen) zwischen diesen Strukturen repräsentieren. Als afferente Verbindungen werden diejenigen bezeichnet, die zu einer Struktur hinführen, wohingegen efferente Verbindungen von einer Struktur wegführen. Demnach entsprechen afferente Verbindungen einlaufenden und efferente Verbindungen auslaufenden Kanten in dem gerichteten Graphen.

In neurologischen Zusammenhängen werden Daten über Gehirnetzwerke und ihre Verbindungen in aller Regel durch eine Adjazenzmatrix dargestellt. Sei n die Anzahl der Knoten in einem gerichteten Graphen $G = (V, E)$. Die *Adjazenzmatrix* für G ist eine $n \times n$ -Matrix $A_G = (a_{i,j})$ mit

$$(a_{i,j}) = \begin{cases} 1 & \text{falls } (u_i, u_j) \in E \\ 0 & \text{falls } (u_i, u_j) \notin E \end{cases} .$$

Ein Beispiel für die Darstellung eines gerichteten Graphen durch eine Adjazenzmatrix ist in Abbildung 3.4 zu sehen.

Für die Darstellung von ungerichteten Graphen mit Adjazenzmatrizen wird jede ungerichteten Kante $e = \{u, v\}$ durch zwei gerichtete Kanten $(u, v), (v, u)$ gespeichert. Damit ergibt sich automatisch, daß Adjazenzmatrizen für ungerichtete Graphen symmetrisch zur Diagonalen sind. Es reicht dann also, die untere Dreiecksmatrix zu betrachten. Einen Überblick über die Verwendung von Graphentheorie in der neurologischen Forschung findet man bei [HKSS02], [Spo03] und [SCKH04]

	10	11	12	13	14	24	25	45	46	8A	8B	9
10	-	1	1	1	1	1	1	0	0	1	0	1
11	1	-	1	1	1	1	1	0	0	0	0	1
12	1	1	-	1	1	1	1	0	1	1	1	1
13	1	1	1	-	1	1	1	0	0	0	1	1
14	1	1	1	1	-	1	1	0	0	0	0	1
24	0	1	1	0	1	-	1	1	1	1	1	1
25	1	1	1	1	1	1	-	0	1	0	1	1
45	1	1	1	1	1	1	0	-	1	1	1	1
46	1	1	1	0	0	1	1	1	-	1	1	1
8A	1	0	1	0	0	1	0	1	1	-		1
8B						1			1	1	-	1
9	1	1	1	1	0	1	1	1	1	1	1	-

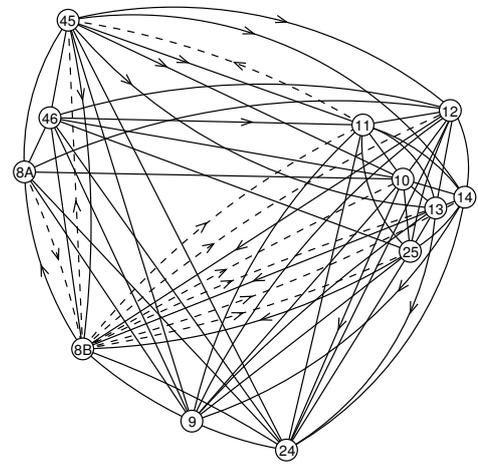


Abbildung 3.4: Links die Verbindungsmatrix für den präfrontalen Cortex nach Walker [Wal40] zusammengestellt aus der CoCoMac-Datenbank [SKB⁺01]. Nicht gefüllte Felder stehen für einen ungeklärten Verbindungszustand. Rechts der gerichtete Graph, der das Netzwerk des präfrontalen Cortex repräsentiert. Die gestrichelten Linien stellen Verbindungen dar, deren Existenz unbekannt ist.

Kapitel 4

Knotenbewertungen in Graphen

4.1 Einführung

Ein Teilgebiet der Neurologie ist die Untersuchung von Verhaltenseffekten, die aus regionalen Gehirnschädigungen, sogenannten Läsionen, resultieren. Dabei versucht man auch, die Bedeutung der geschädigten Hirnregion für die Informationsverarbeitung durch die auftretenden Wahrnehmungs- und Verhaltensänderungen herauszuarbeiten. Es ergibt sich also die Frage, welchen Beitrag eine einzelne Hirnregion zu einer globalen Eigenschaft des gesamten Netzwerks leistet.

Bei der Informationsverarbeitung im Gehirn müssen die neuronalen Strukturen innerhalb der verschiedenen funktionellen Systeme in der Lage sein, miteinander zu kommunizieren. Informationsweiterleitung ist dabei normalerweise mit der Möglichkeit kombiniert, zu antworten. Deshalb haben die Netzwerke, die diese funktionellen Systeme repräsentieren, eine dichte Verbindungsstruktur und insbesondere sind diese gerichteten Graphen stark zusammenhängend. Dies ermöglicht, daß jedes Areal mit jedem anderen, gegebenenfalls über Zwischenstationen, Informationen austauschen kann. Wenn ein Areal seine Funktionalität einbüßt, fällt nicht nur sein direkter Beitrag zur Informationsverarbeitung weg, sondern es werden auch die Informationswege zwischen noch intakten Arealen verändert. Deshalb ist der Beitrag eines Areals zum Zusammenhang des gesamten Netzwerk eine interessante Größe.

Wenn wir also das Gehirn makroskopischer Ebene idealisiert als einen gerichteten Graphen auffassen, so können wir aus diesem Graphen Knoten und damit natürlich auch alle Kanten, die mit diesem Knoten inzident sind, entfernen (analog zu einer lokalen Läsion). Im verbleibenden Graphen kann man überprüfen, wie sich Eigenschaften, wie zum Beispiel die Anzahl der Wege und die Erreichbarkeit der anderen Knoten über diese Wege, also der Zusammenhang des Graphen, verändern haben. Wenn ein Knoten aus einem Graphen entfernt wird, kann sich die Anzahl der Zusammenhangskomponenten vergrößern, und die Größe dieser Änderung kann als Beitrag des Knotens zum Zusammenhang angesehen werden. Die Bedeutung eines Knotens sollte von der Größe der Veränderung abhängig sein, also dem Effekt, den die Entfernung des Knoten hervorruft.

Wir wollen in diesem Kapitel die Auswirkung der Entfernung eines Knotens auf den starken Zusammenhang eines Graphens untersuchen, beziehungsweise ein Maß dafür finden, wie sehr ein Knoten zum Zusammenhang eines Graphen beiträgt. Natürlich lassen sich auf diesem

Wege nur topologische Veränderungen und keine Verhaltensänderungen erfassen, aber man kann so Hinweise auf den potentiellen Einfluß einer Gehirnregion auf andere Regionen erhalten. Von einem systemorientierten Standpunkt ist es eine große Herausforderung, den Einfluß eines einzelnen Areals auf die Verbindungsstruktur des gesamten Systems zu analysieren. Solche Informationen könnten hilfreich sein, um die funktionalen Konsequenzen von Läsionen zu verstehen.

Ein Teil der Ergebnisse dieses Kapitels wurden schon in [AKKW06] und [KRK⁺07] vorgestellt.

4.2 Shapleywerte und ihre Verwendung in der Graphentheorie

4.2.1 Shapleys Axiome

Wir suchen also eine Bewertungsfunktion für die Knoten in einem gerichteten Graphen, die ihre Wichtigkeit in Bezug auf den starken Zusammenhang des Graphen mißt. Diese erhalten wir mit Hilfe einiger Werkzeuge aus der Spieltheorie. Dazu benötigen wir zunächst einmal einige Definitionen.

Definition 4.1 *Sei N eine endliche Menge von Objekten. Eine charakteristische Funktion ist eine Mengenfunktion f von der Potenzmenge von N in die reellen Zahlen, also $f : \mathfrak{P}(N) \rightarrow \mathbb{R}$, für die gilt*

$$f(\emptyset) = 0.$$

Ein Träger von f ist jede Menge $T \subseteq N$ mit

$$f(S) = f(S \cap T) \text{ für alle } S \subseteq N.$$

Die Summe von zwei charakteristischen Funktionen f und g ist durch $(f+g)(S) = f(S)+g(S)$ für alle $S \subseteq N$ definiert und bildet wieder eine charakteristische Funktion. In der Spieltheorie wird die charakteristische Funktion (dort auch Spiel genannt) normalerweise mit v bezeichnet. Da in der Graphentheorie der Buchstabe v jedoch für Knoten reserviert ist, wird hier von dieser Notation abgesehen. Offensichtlich ist jede Obermenge eines Trägers T von f wieder ein Träger von f . Damit ist N immer ein Träger von f . Objekte, die nicht zu einem Träger $T \subsetneq N$ von f gehören, tragen nichts zum Wert von f bei.

Beispiel 4.2 *Sei N eine endliche Menge von Objekten und $a \in N$. Dann ist die Funktion $1_a : \mathfrak{P}(N) \rightarrow \mathbb{R}$ mit*

$$1_a(S) = \begin{cases} 1 & \text{wenn } a \in S \\ 0 & \text{wenn } a \notin S \end{cases}$$

für $S \subseteq N$ eine charakteristische Funktion. Jede Menge $T \subseteq N$ mit $a \in T$ ist Träger von 1_a .

Sei $N = \{a, b, c\}$. Dann ist die Funktion $f : \mathfrak{P}(N) \rightarrow \mathbb{R}$ definiert durch

$$\begin{array}{llll} f(\emptyset) = 0 & f(\{a\}) = 2 & f(\{b\}) = 3 & f(\{c\}) = 4 \\ f(\{a, b\}) = 6 & f(\{a, c\}) = 8 & f(\{b, c\}) = 12 & f(\{a, b, c\}) = 24 \end{array}$$

eine charakteristische Funktion. In diesem Fall ist N der einzige Träger von f .

Die charakteristische Funktion $(1_a + f) : \mathfrak{P}(N) \rightarrow \mathbb{R}$ hat dann nach Definition die Werte

$$\begin{aligned} (1_a + f)(\emptyset) &= 0 & (1_a + f)(\{a\}) &= 3 & (1_a + f)(\{b\}) &= 3 & (1_a + f)(\{c\}) &= 4 \\ (1_a + f)(\{a, b\}) &= 7 & (1_a + f)(\{a, c\}) &= 9 & (1_a + f)(\{b, c\}) &= 12 & (1_a + f)(\{a, b, c\}) &= 25. \end{aligned}$$

Auch hier ist N der einzige Träger.

Sei N eine Menge von Objekten und $\Pi(N)$ die Menge aller Permutationen von N . Für $\pi \in \Pi(N)$ und $S \subseteq N$ sei $\pi^{-1}(S)$ das Urbild von S unter π , also $\pi^{-1}(S) = \{x \mid \pi(x) \in S\}$. Wir definieren die Funktion $f_\pi : \mathfrak{P}(N) \rightarrow \mathbb{R}$ für eine charakteristische Funktion f durch

$$f_\pi(S) = f(\pi^{-1}(S)) \text{ für alle } S \subseteq N.$$

Damit ist f_π wieder eine charakteristische Funktion.

Um die Objekte der Menge N in Bezug auf eine charakteristische Funktion f zu bewerten, wird eine Funktion ϕ verwendet, die jeder charakteristischen Funktion f eine Bewertungsfunktion $\phi_f : N \rightarrow \mathbb{R}$ zuordnet. Shapley hat die folgenden Bedingungen an diese Funktion gestellt [Sha53]:

Axiom 1: für jede Permutation $\pi \in \Pi(N)$ und alle $x \in N$ ist

$$\phi_{f_\pi}(\pi(x)) = \phi_f(x),$$

Axiom 2: für jeden Träger T von N gilt

$$\sum_{x \in T} \phi_f(x) = f(T),$$

und

Axiom 3: für je zwei charakteristische Funktionen f und g gilt

$$\phi_{f+g} = \phi_f + \phi_g.$$

In der Spieltheorie ist ein Argument von ϕ eine charakteristische Funktion f und das Ergebnis ein Vektor $\phi(f) = (y_1, \dots, y_{|N|}) \in \mathbb{R}^{|N|}$. Es gibt dann eine Abbildung $\tau : N \rightarrow \{1, \dots, |N|\}$, so daß jedem Objekt $x \in N$ der Wert $y_{\tau(x)} \in \mathbb{R}$ zugeordnet wird. Da hier aber eine vorgegebene charakteristische Funktion f_G für alle Graphen G verwendet wird, ist es einfacher, als Argument von ϕ die Objekte zu verwenden.

Wir wollen kurz darauf eingehen, was mit diesen Axiomen bewirkt werden soll. Das erste Axiom wird von Shapley auch als Symmetrie bezeichnet. Es wird dadurch gewährleistet, daß $\phi(f)$ eine Eigenschaft der charakteristischen Funktion f ist, und nicht von der Reihenfolge τ , in der die Elemente der Menge N betrachtet werden, abhängt. Diese Formulierung ergibt sich aus der Tatsache, daß Shapley einen Vektor als Wert der Funktion betrachtet und die Reihenfolge in denen die Ergebnisse angeordnet werden, nicht die Werte von ϕ beeinflussen sollen. Das impliziert auch, daß zwei Elemente, die sich nur durch die Reihenfolge, in der sie betrachtet werden, unterscheiden, sich aber in Bezug auf die charakteristische Funktion f austauschbar verhalten, durch ϕ denselben Wert erhalten.

Bemerkung 4.3 Sei $a, b \in N$ und $\tau_{a,b} : N \rightarrow N$ die Transposition von a und b in N , also die Permutation, die nur a und b vertauscht. Gilt für $\tau_{a,b}$ und alle $S \subseteq N$, daß $f(S) = f_{\tau_{a,b}}(S) = f(\tau_{a,b}^{-1}(S))$, dann ist $f = f_{\tau_{a,b}}$ und damit folgt aus dem ersten Axiom $\phi_f(a) = \phi_f(b)$.

Das zweite Axiom bezeichnet Shapley mit Effizienz. Der der gesamten Menge N zugeordnete Wert $f(N)$ soll auf die einzelnen Objekte der Menge verteilt werden. Das ist in sofern effizient, daß alles verteilt wird und nichts übrig bleibt. Das dritte Axiom (Summierungsaxiom) drückt aus, daß, wenn zwei charakteristische Funktionen kombiniert werden, ihre Werte Objekt für Objekt addiert werden müssen. Die Bewertungen beeinflussen sich also nicht gegenseitig, sondern sind voneinander unabhängig.

Shapley hat gezeigt, daß die Funktion ϕ existiert und durch diese drei Axiome eindeutig definiert ist [Sha53]. Desweiteren hat er gezeigt, daß die Bewertung eines Objektes durch ϕ in Bezug auf eine charakteristische Funktion f berechenbar ist durch

$$\phi_f(x) = \sum_{S \subseteq N, x \in S} \frac{(|S| - 1)! (|N| - |S|)!}{|N|!} (f(S) - f(S - \{x\})), \quad (4.1)$$

wobei $|S|$ und $|N|$ die Anzahl der Elemente in S bzw. N sind, oder alternativ durch

$$\phi_f(x) = \frac{1}{|N|!} \sum_{\pi \in \Pi_N} (f(\lambda(\pi, x) \cup \{x\}) - f(\lambda(\pi, x))), \quad (4.2)$$

wobei Π_N die Menge aller Anordnungen $\pi : N \rightarrow \{1, \dots, |N|\}$ ist und

$$\lambda(\pi, x) = \{y \in N \mid \pi(y) < \pi(x)\}$$

die Menge aller $y \in N$ ist, die durch π links von x angeordnet werden.

Die Bewertung $\phi_f(x)$ eines Objektes $x \in N$ wird auch als *Shapleywert* von x bezeichnet. Um den Beitrag eines Objektes zu dem gemeinsamen Ertrag zu berechnen, werden also Teilmengen von N in allen Größen betrachtet. An Gleichung 4.1 läßt sich erkennen, daß Erträge von Teilmengen derselben Größe i , $i \in \{1, \dots, |N|\}$ mit demselben Gewicht $\frac{(i-1)! (|N|-i)!}{|N|!}$ in die Berechnung eingehen. Die Summe der Gewichte über alle Teilmengen einer festen Größe i ist $\frac{1}{|N|}$: Es gibt

$$\binom{|N| - 1}{i - 1} = \frac{|N| - 1!}{(i - 1)! (|N| - i)!}$$

Möglichkeiten eine Menge S aus x und $i - 1$ weiteren Elementen von N zu bilden. Damit erhalten wir

$$\sum_{S \subseteq N, |S|=i, x \in S} \frac{(i - 1)! (|N| - i)!}{|N|!} = \frac{|N| - 1!}{(i - 1)! (|N| - i)!} \cdot \frac{(i - 1)! (|N| - i)!}{|N|!} = \frac{1}{|N|}.$$

Veranschaulichen wir uns diese Begriffe mit den Funktionen aus Beispiel 4.2.

Beispiel 4.4 Seien $a, b, c \in N$. Da $\{a\}$ Träger von 1_a ist, folgt aus dem zweiten Axiom $\phi_{1_a}(a) = 1 = 1_a(\{a\})$. Weil aber auch $\{a, b\}$ Träger von 1_a ist, folgt aus $\phi_{1_a}(a) = 1$ mit dem zweiten Axiom außerdem $\phi_{1_a}(b) = 0$. Da b und c in Bezug auf die Werte von 1_a austauschbar

sind, also für alle Transpositionen $\tau \in T_{b,c} : N \rightarrow N$ und alle $S \subseteq N$ gilt $f(S) = f(\tau^{-1}(S))$, folgt aus dem ersten Axiom $\phi_{1_a}(c) = 0$.

Für die Menge $N = \{a, b, c\}$ gibt es die folgenden sechs Permutationen: $abc, acb, bac, bca, cab, cba$. Entsprechend gibt es auch sechs Anordnungen der Elemente. Für die Funktion f aus Beispiel 4.2 erhalten wir mit Formel 4.2

$$\begin{aligned}\phi_f(a) &= \frac{1}{6} \cdot ((f(\{a\}) - f(\emptyset)) + (f(\{a\}) - f(\emptyset)) + (f(\{b, a\}) - f(\{b\})) + (f(\{c, a\}) - f(\{c\})) \\ &\quad + (f(\{b, c, a\}) - f(\{b, c\})) + (f(\{c, b, a\}) - f(\{c, b\}))) \\ &= \frac{1}{6} \cdot ((2 - 0) + (2 - 0) + (6 - 3) + (8 - 4) + (24 - 12) + (24 - 12)) \\ &= \frac{35}{6},\end{aligned}$$

$$\begin{aligned}\phi_f(b) &= \frac{1}{6} \cdot ((f(\{b\}) - f(\emptyset)) + (f(\{b\}) - f(\emptyset)) + (f(\{a, b\}) - f(\{a\})) + (f(\{c, b\}) - f(\{c\})) \\ &\quad + (f(\{a, c, b\}) - f(\{a, c\})) + (f(\{c, a, b\}) - f(\{c, a\}))) \\ &= \frac{1}{6} \cdot ((3 - 0) + (3 - 0) + (6 - 2) + (12 - 4) + (24 - 8) + (24 - 8)) \\ &= \frac{50}{6},\end{aligned}$$

$$\begin{aligned}\phi_f(c) &= \frac{1}{6} \cdot ((f(\{c\}) - f(\emptyset)) + (f(\{c\}) - f(\emptyset)) + (f(\{a, c\}) - f(\{a\})) + (f(\{b, c\}) - f(\{b\})) \\ &\quad + (f(\{a, b, c\}) - f(\{a, b\})) + (f(\{b, a, c\}) - f(\{b, a\}))) \\ &= \frac{1}{6} \cdot ((4 - 0) + (4 - 0) + (8 - 2) + (12 - 3) + (24 - 6) + (24 - 6)) \\ &= \frac{59}{6}.\end{aligned}$$

Insgesamt also $\phi_f(a) + \phi_f(b) + \phi_f(c) = \frac{35}{6} + \frac{50}{6} + \frac{59}{6} = \frac{144}{6} = 24 = f(\{a, b, c\})$.

Für die Funktion $(1_a + f)$ ergeben sich damit die folgenden Shapleywerte:

$$\phi_{(1_a+f)}(a) = 1 + \frac{35}{6} = \frac{41}{6}, \quad \phi_{(1_a+f)}(b) = 0 + \frac{50}{6} = \frac{50}{6}, \quad \phi_{(1_a+f)}(c) = 0 + \frac{59}{6} = \frac{59}{6}.$$

4.2.2 Übertragung von Shapleys Axiomen auf Graphen

Da im Folgenden ausschließlich charakteristische Funktionen auf Knotenmengen betrachtet werden, soll jetzt überlegt werden, was Shapley Axiome für diese Funktionen bedeuten.

Sei $G = (V, E)$ ein Graph und $f_G : \mathfrak{P}(V) \rightarrow \mathbb{R}$ eine Funktion auf der Potenzmenge der Knoten mit $f_G(\emptyset) = 0$. Für diese Situation kann man Shapleys Axiome folgendermaßen formulieren:

(Axiom 1:) Seien $G = (V, E)$ und $G' = (V', E')$ isomorphe Graphen und $b : V \rightarrow V'$ eine Bijektion mit $\{u, v\} \in E \Leftrightarrow \{b(u), b(v)\} \in E'$. Dann gilt für alle Knoten $v \in V$

$$\phi_{f_G}(v) = \phi_{f_{G'}}(b(v)).$$

(Axiom 2:)

$$\sum_{v \in V} \phi_{f_G}(v) = f_G(V),$$

und

(Axiom 3:) für je zwei charakteristische Funktionen f_G und g_G auf der Knotenmenge V gilt

$$\phi_{f_G+g_G}(v) = \phi_{f_G}(v) + \phi_{g_G}(v) \text{ für alle } v \in V.$$

Das 1. Axiom sagt, daß isomorphe Knoten dieselbe Bewertung erhalten sollen. Solche Knoten sind in Bezug auf die Graphenstruktur nicht unterscheidbar. Dies entspricht der Aussage von Bemerkung 4.3. Da wir die Shapleywerte einzeln und nicht als Vektor für alle Knoten berechnen, ist der Bezug auf eine Reihenfolge nicht notwendig. Das 2., daß die Summe der einzelnen Knotenbewertungen dem Wert entsprechen soll, den f_G der gesamten Knotenmenge V zuordnet. Oder anders gesagt, dieser Wert soll auf die einzelnen Knoten verteilt werden. Dies stellt keine Einschränkung dar, da wir auch nur charakteristische Funktionen betrachten werden, bei denen V der einzige Träger ist. Die Umformulierung dient lediglich der Vereinfachung der Notation. Das 3. Axiom ist unverändert geblieben.

Wenn die Funktion f eine Grapheigenschaft mißt, soll die Bewertung ϕ beschreiben, wie sehr ein einzelner Knoten eines Graphens an einer Grapheigenschaft beteiligt ist.

Beispiel 4.5 Sei $G = (\{v_1, v_2, v_3\}, \{\{v_1, v_2\}, \{v_2, v_3\}\})$, also ein ungerichteter Weg mit 3 Knoten (siehe Abbildung) und $f_G : \mathfrak{P}(V) \rightarrow \mathbb{R}$ die Funktion, die jeder Knotenmenge $S \subseteq \{v_1, v_2, v_3\}$ die Länge des längsten Weges in dem durch S induzierten Teilgraph $G|_S$ zuordnet.



Dann ist $f(\{v_1, v_2, v_3\}) = 2$, $f(\{v_1, v_2\}) = 1$, $f(\{v_2, v_3\}) = 1$ und für alle anderen Teilmengen S von V ist $f(S) = 0$. Damit ergeben sich nach Formel 4.2 folgende Shapleywerte: (Summanden, die 0 sind, sind nicht angegeben.)

$$\begin{aligned} \phi_{f_G}(v_1) &= \frac{1}{6} \cdot ((f_G(\{v_1, v_2, v_3\}) - f_G(\{v_2, v_3\})) + 2 \cdot (f_G(\{v_1, v_2\}) - f_G(\{v_2\}))) \\ &= \frac{1}{6} \cdot (1 + 2) = \frac{1}{2} \end{aligned}$$

$$\begin{aligned} \phi_{f_G}(v_2) &= \frac{1}{6} \cdot ((f_G(\{v_1, v_2, v_3\}) - f_G(\{v_1, v_3\})) + 2 \cdot (f_G(\{v_1, v_2\}) - f_G(\{v_1\})) \\ &\quad + 2 \cdot (f_G(\{v_2, v_3\}) - f_G(\{v_3\}))) \\ &= \frac{1}{6} \cdot (2 + 2 + 2) = 1 \end{aligned}$$

$$\begin{aligned} \phi_{f_G}(v_3) &= \frac{1}{6} \cdot ((f_G(\{v_1, v_2, v_3\}) - f_G(\{v_1, v_2\})) + 2 \cdot (f_G(\{v_2, v_3\}) - f_G(\{v_2\}))) \\ &= \frac{1}{6} \cdot (1 + 2) = \frac{1}{2} \end{aligned}$$

4.2.3 Arbeiten zu Shapleywerten und Graphen

Myerson war der erste, der Graphentheorie und Shapleywerte zusammenbrachte, indem er graph-beschränkte charakteristische Funktionen auf den Knoten von ungerichteten Graphen wie folgt definierte [Mye77]:

Definition 4.6 Sei $G = (V_G, E_G)$ ein ungerichteter Graph, $h : \mathfrak{P}(V_G) \rightarrow \mathbb{R}$ eine charakteristische Funktion und $CC(G|_S)$ für $S \subseteq V_G$ die Menge der Zusammenhangskomponenten des

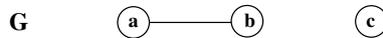
durch S induzierten Teilgraphen von G . Die graph-beschränkte charakteristische Funktion h^G ist definiert durch

$$h^G(S) := \sum_{T \in CC(G|_S)} h(T) \quad \text{für alle } S \subseteq V_G.$$

Der Shapleywert einer solchen graph-beschränkten Funktion h^G heißt *Myersonwert* der Funktion h . Myerson wollte damit die charakteristische Funktion auf solche Mengen einschränken, die miteinander verbunden sind. Dabei mißt die Funktion h nicht unbedingt eine Grapheigenschaft. Die Zusammenhangsstruktur geht nur über die Einschränkung der Funktion h auf in G zusammenhängende Teilmengen von V ein. Für die später in Abschnitt 4.5 definierte charakteristische Funktion f_G auf der Knotenmenge eines ungerichteten Graphen G ist ϕ_{f_G} äquivalent mit dem Myersonwert für eine Funktion h mit $h(V') = 1$ für alle $V' \subseteq V_G$. Außerdem gilt für die in Abschnitt 4.5 betrachtete charakteristische Funktion f , daß $f_G^G(S) = f_G(S)$ für alle $S \in V_G$ ist. Also ist der Shapleywert für diese Funktion gleich dem Myersonwert.

Myerson Ansatz unterscheidet sich also von der hier vorgestellten Vorgehensweise, da wir für eine Menge $S \subseteq V_G$ die Anzahl der Zusammenhangskomponenten in dem durch S induzierten Teilgraphen $G|_S$ betrachten, um den Wert unserer charakteristischen Funktion f_G zu definieren und nicht um eine Funktion zu beschränken. Als Beispiel für Meyersonwerte betrachten wir die Funktion f aus Beispiel 4.2 kombiniert mit einem Graph aus drei Knoten.

Beispiel 4.7 Sei $G = (\{a, b, c\}, \{\{a, b\}\})$ ein ungerichteter Graph mit 3 Knoten (siehe Abbildung) und f wie in Beispiel 4.2.



Dann gelten

$$\begin{aligned} f^G(\{a, b, c\}) &= f(\{a, b\}) + f(\{c\}) = 10, \\ f^G(\{a, c\}) &= f(\{a\}) + f(\{c\}) = 6, \\ f^G(\{b, c\}) &= f(\{b\}) + f(\{c\}) = 7 \end{aligned}$$

und für alle anderen Teilmengen $S \subseteq \{a, b, c\}$ ist $f^G(S) = f(S)$. Damit ergeben sich die folgenden Meyersonwerte.

$$\begin{aligned} \phi_{f^G}(a) &= \frac{1}{6} \cdot ((2 - 0) + (2 - 0) + (6 - 3) + (6 - 4) + (10 - 7) + (10 - 7)) = \frac{15}{6}, \\ \phi_{f^G}(b) &= \frac{1}{6} \cdot ((3 - 0) + (3 - 0) + (6 - 2) + (7 - 4) + (10 - 6) + (10 - 6)) = \frac{21}{6}, \\ \phi_{f^G}(c) &= \frac{1}{6} \cdot ((4 - 0) + (4 - 0) + (6 - 2) + (7 - 3) + (10 - 6) + (10 - 6)) = \frac{24}{6}. \end{aligned}$$

Daraus folgt $\phi_{f^G}(a) + \phi_{f^G}(b) + \phi_{f^G}(c) = \frac{15}{6} + \frac{21}{6} + \frac{24}{6} = \frac{60}{6} = 10 = f^G(\{a, b, c\})$.

Mehrere Autoren haben seit der Arbeit von Myerson die Berechnung von Shapleywerten für auf Knoten von Graphen analysiert. Owen [Owe86] definierte für eine Teilmenge $T \subseteq N$, $T \neq \emptyset$ eine charakteristische Funktion p_T durch

$$p_T(S) = \begin{cases} -1, & \text{wenn } S \cap T \neq \emptyset \\ 0, & \text{wenn } S \cap T = \emptyset \end{cases}.$$

Ist $T = N$, dann gilt $p_N(S) = -1$ für alle $S \subseteq N$. Die graph-beschränkte Funktion für einen Graphen $G = (V, E)$ und $T = N = V$ ist

$$p_V^G(S) = \sum_{T \in \text{CC}(G|_S)} p_{V_G}(T) = -|\text{CC}(G|_S)| \quad \text{für alle } S \subseteq V,$$

wobei $\text{CC}(G|_S)$ für $S \subseteq V_G$ wieder die Menge der Zusammenhangskomponenten des durch S induzierten Teilgraphen von G ist.

Owen konnte zeigen, daß für einen ungerichteten Baum $G = (V, E)$ und $v \in V$ für den Myersonwert der Knoten

$$\Phi_{p_V^G}(v) = -1 + \frac{\deg(v)}{2}$$

gilt, wobei $\deg(v)$ der Grad von v ist [Owe86].

Die Funktion p_V wird auch von Gómez et al. [GGAM⁺04] untersucht, die eine einfache Separationseigenschaft für ungerichtete Graphen zeigen, die verwendet werden kann, um einige Myersonwerte effizienter zu berechnen.

Deng und Papadimitriou betrachten in [DP94] eine charakteristische Funktion für einen kantenbewerteten Graphen $G = (V, E)$, die die Gewichte aller Kanten zwischen zwei Knoten von $V' \subseteq V$ aufsummiert.

Van den Brink und Borm in [vdBP02] analysieren eine charakteristische Funktion auf Knoten eines ungerichteten Graphen, die über Nachbarn von Knoten in einer Menge definiert sind. Diese Funktionen berücksichtigen allerdings nur lokale Eigenschaften des Graphen, können aber dafür effizient berechnet werden. In diesem Kapitel soll im Gegensatz dazu eine Funktion, die den Einfluß eines Knoten auf die gesamte Zusammenhangsstruktur eines Graphen mißt, untersucht werden.

4.3 Knotenbewertungen für gerichtete Graphen

Wir definieren zuerst die charakteristische Funktion f_G , um die Knoten in einem gerichteten Graphen zu bewerten. Diese Bewertung mißt den Einfluß eines Knotens auf die Zusammenhangsstruktur.

Sei $G = (V, E)$ ein gerichteter Graph und $\text{SCC}(G)$ die Menge aller *starken Zusammenhangskomponenten* von G und f_G eine Funktion aus den Teilmengen von V_G in die natürlichen Zahlen \mathbb{N}_0 , so daß für jede Teilmenge $V' \subseteq V_G$,

$$f_G(V') = |\text{SCC}(G|_{V'})|.$$

Das heißt, $f_G(V')$ ist die Anzahl der starken Zusammenhangskomponenten in dem Teilgraphen von G , der durch die Knoten von V' induziert wird. Es handelt sich bei f_G um eine charakteristische Funktion, da $f_G(\emptyset)$ immer Null ist. Die gesamte Knotenmenge V_G ist der einzige Träger von f_G für jeden gerichteten Graph G . Mit Axiom 2 erhalten wir

$$\sum_{v \in V_G} \phi_{f_G}(v) = f_G(V_G) = |\text{SCC}(G)|.$$

Da ein kleinerer Wert von f_G einen größeren Zusammenhang bedeutet, gilt dasselbe auch für ϕ_{f_G} . Je niedriger die Bewertung eines Knotens desto größer ist seine Bedeutung für den Zusammenhang des Graphens.

Abbildung 4.1 zeigt ein Beispiel der Knotenbewertung ϕ_{f_G} für einen gerichteten Graphen G mit Knotenmenge $V_G = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$ sowie die Knotenbewertung $\phi_{f_{G'}}$ für den gerichteten Graphen G' mit Knotenmenge $V_{G'} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_8\}$, der aus G durch Entfernen von Knoten v_7 und seinen inzidenten Kanten entstanden ist. Da G stark zusammenhängend ist, ist $f_G(V_G) = \sum_{v \in V_G} \phi_{f_G}(v) = 1$.

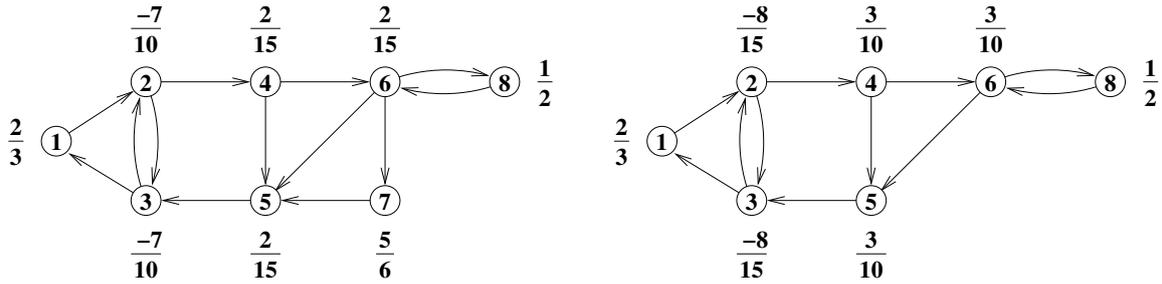


Abbildung 4.1: Links die Knotenbewertung ϕ_{f_G} für einen gerichteten Graphen G mit 8 Knoten und rechts die Knotenbewertung $\phi_{f_{G'}}$ für einen gerichteten Graphen G' mit 7 Knoten. Je niedriger die Bewertung eines Knotens desto größer ist seine Bedeutung für den Zusammenhang des Graphens.

Bei einer Berechnung von ϕ_{f_G} nach Formel 4.2 erhält der Knoten v_8 die Bewertung $\phi_{f_G}(v_8) = \frac{1}{2}$, weil $f_G(\lambda(\pi, v_8) \cup \{v_8\}) - f_G(\lambda(\pi, v_8)) = 0$ genau dann gilt, wenn $\pi(v_6) < \pi(v_8)$ ist. Sonst erhalten wir $f_G(\lambda(\pi, v_8) \cup \{v_8\}) - f_G(\lambda(\pi, v_8)) = 1$. Jeder der beiden Fälle $\pi(v_6) < \pi(v_8)$ und $\pi(v_6) > \pi(v_8)$ tritt genau in genau der Hälfte aller $8!$ Anordnungen π von V_G auf.

Knoten v_1 hat die Bewertung $\phi_{f_G}(v_1) = \frac{2}{3}$, da $f_G(\lambda(\pi, v_1) \cup \{v_1\}) - f_G(\lambda(\pi, v_1)) = 0$ genau dann, wenn $\pi(v_2) < \pi(v_1)$ und $\pi(v_3) < \pi(v_1)$. Sonst erhalten wir $f_G(\lambda(\pi, v_1) \cup \{v_1\}) - f_G(\lambda(\pi, v_1)) = 1$. Hier tritt der erste Fall bei genau einem Drittel der $8!$ Anordnungen π von V_G auf.

Die gleichen Überlegungen können verwendet werden, um die Bewertung $\phi_{f_{G'}}(v_1) = \frac{2}{3}$ und $\phi_{f_{G'}}(v_8) = \frac{1}{2}$ im Graphen G' zu erhalten. Die Bewertungen aller anderen Knoten unterscheiden sich in G von denen in G' . Dies ist nicht weiter überraschend, da für jeden stark zusammenhängenden Graphen $G = (V, E)$ die Gleichung $f_G(V) = \sum_{v \in V} \phi_{f_G}(v) = 1$ gilt, unabhängig von der Größe der Knotenmenge V . Also ist es nicht möglich Knotenbewertungen von Graphen zu vergleichen, die nicht dieselbe Anzahl Knoten haben, wie zum Beispiel G und G' aus der Abbildung 4.1.

Seien $G = (V_G, E_G)$ und $G' = (V_{G'}, E_{G'})$ zwei gerichtete Graphen. Wenn G und G' isomorph sind und b ein Isomorphismus zwischen G und G' ist, dann ist $f_G(V') = f_{G'}(b(V'))$ für jede Knotenmenge $V' \subseteq V_G$. Dabei ist $b(V') = \{b(u) \mid u \in V'\}$ das Bild von V' unter b . Dies impliziert $\phi_{f_G}(v) = \phi_{f_{G'}}(b(v))$ für alle Knoten $v \in V$. Damit ist das zweite Axiom aus Abschnitt 4.2.2 erfüllt. Sei $V' \subseteq V_G$ eine Teilmenge der Knotenmenge von G . Ein Graph G heißt V' -symmetrisch, wenn es für jedes Knotenpaar $v_1, v_2 \in V'$ einen Isomorphismus b von G nach G gibt, so daß $b(v_1) = v_2$. In V' -symmetrischen Graphen haben alle Knoten $v \in V'$ dieselbe Knotenbewertung. Wenn zwei Knoten v_1, v_2 dieselbe Nachbarschaft haben, daß heißt,

wenn $\{u \mid (u, v_1) \in E_G\} = \{u \mid (u, v_2) \in E_G\}$ und $\{u \mid (v_1, u) \in E_G\} = \{u \mid (v_2, u) \in E_G\}$, dann ist G offensichtlich $\{v_1, v_2\}$ -symmetrisch. Abbildung 4.2 zeigt zwei Beispiele für teilweise symmetrische Graphen.

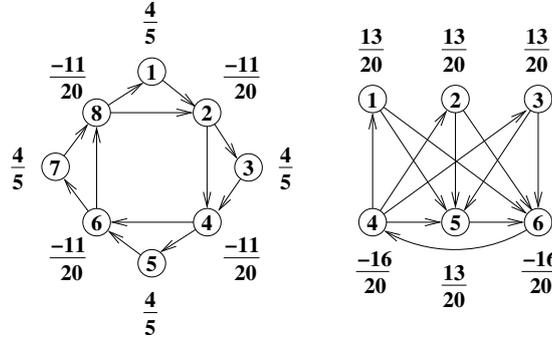


Abbildung 4.2: Der linke Graph ist $\{v_1, v_3, v_5, v_7\}$ -symmetrisch sowie $\{v_2, v_4, v_6, v_8\}$ -symmetrisch und der rechte Graph ist $\{v_1, v_2, v_3\}$ -symmetrisch.

In einem stark zusammenhängenden Graphen G mit Knotenmenge V_G , die V_G -symmetrisch ist, sind die Bewertungen für alle Knoten gleich. Da $\sum_{v \in V_G} \phi_{f_G}(v) = f_G(V_G) = 1$, ist $\phi_{f_G}(v) = \frac{1}{|V_G|}$ für jeden Knoten $v \in V_G$, siehe auch die Beispiele in Abbildung 4.3. Obwohl die Bewertungen für alle Knoten in den beiden Graphen links in der Abbildung gleich sind, kommen sie durch unterschiedliche Berechnungen nach Formel 4.2 zustande. Für einen Knoten v des ersten Graphen in Abbildung 4.3 (eine Clique mit sechs Knoten) ist $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = 1$, wenn $\lambda(\pi, v) = \emptyset$ ist, was in $\frac{1}{6}$ aller $6!$ Anordnungen der Knotenmenge der Fall ist. In allen anderen Fällen gilt $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = 0$. Für einen Knoten v des zweiten Graphen in Abbildung 4.3 (ein Kreis mit sechs Knoten) ist $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = -4$, wenn $\lambda(\pi, v)$ alle Knoten des Kreises außer v enthält, was in $\frac{1}{6}$ aller $6!$ Anordnungen der Knotenmenge der Fall ist. In allen anderen Fällen ist $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = 1$, also erhalten wir insgesamt $-4 \cdot \frac{1}{6} + 1 \cdot \frac{5}{6} = \frac{1}{6}$.

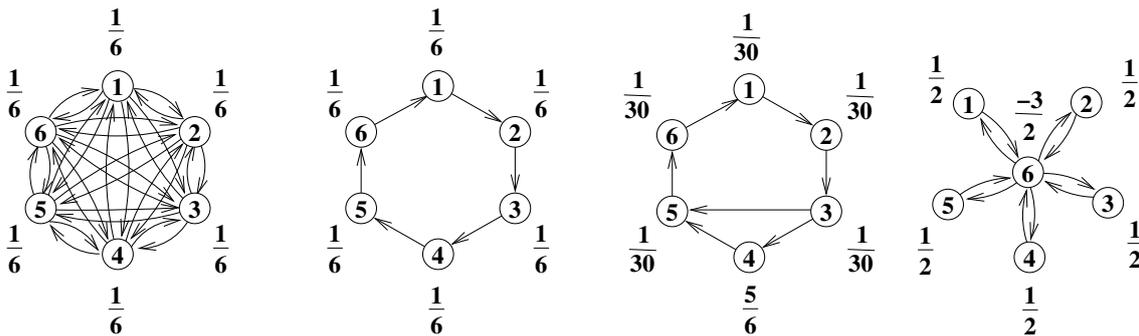


Abbildung 4.3: Vier Beispiele für die Knotenbewertung ϕ_{f_G} in Graphen mit sechs Knoten. Die zwei linken Graphen (eine Clique und ein Kreis) sind beide $\{v_1, v_2, v_3, v_4, v_5, v_6\}$ -symmetrisch. In dem dritten Graph hat v_4 die größte und im vierten Graph hat v_6 die kleinste Knotenbewertung, die in einem Graphen mit sechs Knoten möglich ist.

Der Wertebereich, den ϕ_{f_G} annehmen kann, ist von der Anzahl der Knoten im Graph G abhängig. Wenn der Graph nur aus einem Knoten besteht, hat dieser die Bewertung 1. Für

Graphen mit größeren Knotenmengen erhalten wir die folgende Abschätzung:

Lemma 4.8 *Sei $G = (V_G, E_G)$ ein stark zusammenhängender, gerichteter Graph mit $|V_G| \geq 2$. Dann gilt für jeden Knoten $v \in V_G$*

$$1 - \frac{|V_G| - 1}{2} \leq \phi_{f_G}(v) \leq \frac{|V_G| - 1}{|V_G|}.$$

Beweis: Sei Π_{V_G} die Menge aller Anordnungen $\pi : V_G \rightarrow \{1, \dots, |V_G|\}$. Für einen Knoten $v \in V_G$ und für jedes $i \in \{0, \dots, |V_G| - 1\}$ ist die Anzahl der Anordnungen in Π_{V_G} , in denen genau i Knoten links von v stehen, $(|V_G| - 1)!$. Dies sind alle Anordnungen $\pi \in \Pi$ mit $\pi(v) = i + 1$ und $|\lambda(\pi, v)| = i$.

Für einen Knoten $v \in V$ gilt für alle Anordnungen $\pi \in \Pi$

$$1 - i \leq f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) \leq 1$$

mit $i = |\lambda(\pi, v)|$. Die obere Grenze ergibt sich daraus, daß durch das hinzufügen eines Knotens zu einem Graphen maximal eine weitere starke Zusammenhangskomponente entsteht. Die untere, weil $G|_{\lambda(\pi, v)}$ maximal i starke Zusammenhangskomponenten haben kann und jeder Graph, der mindestens einen Knoten enthält auch mindestens eine starke Zusammenhangskomponente besitzt.

Zuerst zeigen wir die untere Grenze. Diese wird für einen Knoten $v \in V_G$ erreicht, wenn $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = 1 - i$ für alle $\pi \in \Pi$. Dann ist jeder induzierte Teilgraph von G , der v enthält, stark zusammenhängend und jeder induzierte Teilgraph von G , der v nicht enthält, hat so viele starke Zusammenhangskomponenten wie Knoten. (Ein Beispiel für diese Situation ist der zentrale Knoten in einem Stern, siehe Abbildung 4.3.) In diesem Fall verbindet der Knoten v verbindet also alle anderen Knoten in G . Damit erhalten wir für einen beliebigen Knoten $v \in V$ die folgende Abschätzung:

$$\begin{aligned} \phi_f(v) &= \frac{1}{|V_G|!} \sum_{\pi \in \Pi_{V_G}} (f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v))) \\ &\stackrel{(*)}{\geq} \frac{1}{|V_G|!} \sum_{i=0}^{|V_G|-1} (1 - i)(|V_G| - 1)! \\ &= \frac{1}{|V_G|!} (|V_G| \cdot (|V_G| - 1)! - \frac{|V_G| \cdot (|V_G| - 1)}{2} (|V_G| - 1)!) \\ &= 1 - \frac{|V_G| - 1}{2}. \end{aligned}$$

(*) Wenn wir die Aufzählungen nach der Anzahl $i = 1, \dots, |V_G| - 1$ der Knoten links von v anordnen, dann kann sich die Anzahl der starken Zusammenhangskomponenten durch das Hinzufügen von v maximal von i zu 1 reduzieren.

Betrachten wir jetzt die obere Grenze. Ein Knoten $v \in V_G$ hat den höchsten Wert $\phi_f(v)$, wenn $f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)) = 1$ für möglichst viele $\pi \in \Pi$. (Da der Graph mindestens zwei Knoten hat und stark zusammenhängend ist, kann es nicht für alle $\pi \in \Pi$ gelten. $G|_{V_G - v}$ hat mindestens eine Zusammenhangskomponente und G genau eine Zusammenhangskomponente.) Die obere Grenze wird erreicht, wenn v allein eine starke Zusammenhangskomponente in jedem induzierten echten Teilgraph $G' = G|_{V'}$, $v \in V' \subsetneq V_G$ bildet und der induzierte Teilgraph

$G' = G|_{V_G - v}$ stark zusammenhängend ist. (Siehe Abbildung 4.3 für ein Beispiel.) In diesem Fall erhöht das Hinzufügen von v zu einem induzierten Teilgraphen den Zusammenhang also nie. Damit erhalten wir für einen beliebigen Knoten $v \in V$ die folgende Abschätzung:

$$\begin{aligned} \phi_f(v) &= \frac{1}{|V_G|!} \sum_{\pi \in \Pi_{V_G}} (f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v))) \\ &\stackrel{(*)}{\leq} \frac{1}{|V_G|!} \sum_{i=0}^{|V_G|-2} 1 \cdot (|V_G| - 1)! + 0 \cdot (|V_G| - 1) \\ &= \frac{|V_G| - 1}{|V_G|} \end{aligned}$$

(*) Wenn wir wieder die Aufzählungen geordnet nach der Anzahl $i = 1, \dots, |V_G| - 1$ von Knoten links von v betrachten, dann erhöht das Hinzufügen von v die Anzahl der starken Zusammenhangskomponenten für $i = 1, \dots, |V_G| - 2$ maximal um 1. Für $i = |V_G| - 1$ erhöht das Hinzufügen von v die Anzahl der starken Zusammenhangskomponenten nicht, da G stark zusammenhängend ist. \square

Die Berechnung der Knotenbewertung $\phi_{f_G}(v)$ nach Gleichung 4.1 oder Gleichung 4.2 ist sehr ineffizient. Die Anzahl der zu betrachtenden Teilmengen nimmt exponentiell mit der Anzahl der Knoten in G zu. Um die Berechnung von ϕ_{f_G} für viele praktisch interessante Fälle zu bewältigen, stellen wir eine Methode vor, um einen großen Graphen in kleinere Teile zu zerlegen. Diese Zerlegung wird es ermöglichen, die Knotenbewertungen des Ausgangsgraphen effizient zu berechnen, indem die Bewertungen der kleineren Teilgraphen verwendet werden. Unsere Zerlegungsmethode wird durch die folgenden drei Lemmata und Satz 4.14 vorgestellt.

Das erste Lemma zeigt, daß die Berechnung einer Bewertung $\phi_{f_G}(v)$, für die die Argumente von f_G auf die Knoten von einer Teilmenge $V' \subseteq V_G$ beschränkt sind, der Berechnung von $\phi_{f_{G|_{V'}}}(v)$ entspricht.

Lemma 4.9 *Sei $G = (V_G, E_G)$ ein Graph, $V' \subseteq V_G$, und $G' = G|_{V'}$. Sei Π_{V_G} die Menge aller Anordnungen $\pi : V_G \rightarrow \{1, \dots, |V_G|\}$. Dann gilt für jeden Knoten $v \in V'$*

$$\phi_{f_{G'}}(v) = \frac{1}{|V_G|!} \sum_{\pi \in \Pi_{V_G}} (f_G((\lambda(\pi, v) \cup \{v\}) \cap V') - f_G(\lambda(\pi, v) \cap V')).$$

Beweis: Sei $\Pi_{V'}$ die Menge aller Anordnungen $\pi' : V' \rightarrow \{1, \dots, |V'|\}$. Wir zeigen zuerst, daß es für jede Anordnung $\pi' \in \Pi_{V'}$ genau $(|V'| + 1) \cdot (|V'| + 2) \cdot \dots \cdot |V_G|$ eindeutige Anordnungen $\pi \in \Pi_{V_G}$ gibt, so daß für jedes Knotenpaar $v_1, v_2 \in V'$, $\pi'(v_1) < \pi'(v_2)$ genau dann, wenn $\pi(v_1) < \pi(v_2)$. Sei $p = (v_{i_1}, \dots, v_{i_{|V'|}})$ eine Folge der Knoten in V' in der durch π' definierten Reihenfolge, daß heißt

$$\pi'(v_{i_1}) < \pi'(v_{i_2}) < \dots < \pi'(v_{i_{|V'|}}).$$

Wenn wir die Knoten aus der Menge $V_G - V'$ in einer beliebigen Reihenfolge betrachten, dann kann der erste Knoten aus $V_G - V'$ an $|V'| + 1$ Stellen in die Folge p eingefügt werden, um eine Folge mit $|V'| + 1$ Knoten zu erhalten. Danach kann der nächste Knoten an $|V'| + 2$ Stellen in der erweiterten Folge eingefügt werden, um eine Folge mit $|V'| + 2$ Knoten zu erhalten, und so weiter. Der letzte Knoten von $V_G - V'$ kann an $|V_G|$ Stellen in die Folge eingefügt werden, die durch das Einfügen der anderen Knoten entstanden ist, um eine Folge mit allen $|V_G|$ Knoten

von G zu erhalten. Für alle diese $(|V'| + 1) \cdot (|V'| + 2) \cdots |V_G|$ Anordnungen π , die bezüglich der durch π' vorgegebenen Anordnung der Knoten aus V' übereinstimmen, erhalten wir

$$\begin{aligned} & f_{G'}((\lambda(\pi, v) \cup \{v\}) \cap V') - f_{G'}(\lambda(\pi, v) \cap V') \\ = & f_{G'}(\lambda(\pi', v) \cup \{v\}) - f_{G'}(\lambda(\pi', v)) \end{aligned}$$

für jeden Knoten $v \in V'$, und somit

$$\begin{aligned} \phi_{f_{G'}}(v) &= \frac{1}{|V'|!} \sum_{\pi' \in \Pi_{V'}} (f_{G'}(\lambda(\pi', v) \cup \{v\}) - f_{G'}(\lambda(\pi', v))) \\ &= \frac{1}{|V'|!} \sum_{\pi \in \Pi_{V_G}} \frac{(f_{G'}((\lambda(\pi, v) \cup \{v\}) \cap V') - f_{G'}(\lambda(\pi, v) \cap V'))}{(|V'|+1) \cdot (|V'|+2) \cdots |V_G|} \\ &= \frac{1}{|V_G|!} \sum_{\pi \in \Pi_{V_G}} (f_{G'}((\lambda(\pi, v) \cup \{v\}) \cap V') - f_{G'}(\lambda(\pi, v) \cap V')) \\ &= \frac{1}{|V_G|!} \sum_{\pi \in \Pi_{V_G}} (f_G((\lambda(\pi, v) \cup \{v\}) \cap V') - f_G(\lambda(\pi, v) \cap V')). \end{aligned}$$

Die letzte Gleichheit folgt aus der Tatsache, daß $f_{G'}(V'' \cap V') = f_G(V'' \cap V')$ für jede Teilmenge $V'' \subseteq V_G$ gilt. \square

Die Bewertung eines Knotens in einem Graphen G hängt nur von der Zusammenhangsstruktur der starken Zusammenhangskomponente ab, zu der der Knoten gehört, wie folgende Beobachtung zeigt. Ist $G' = (V_{G'}, E_{G'})$ eine starke Zusammenhangskomponente von $G = (V_G, E_G)$, dann gilt für jeden Knoten $v \in V_{G'}$ und jede Knotenmenge $V'' \subseteq V_G$

$$f_G((V'' \cup \{v\}) \cap V_{G'}) - f_G(V'' \cap V_{G'}) = f_G(V'' \cup \{v\}) - f_G(V''),$$

und somit nach Lemma 4.9 $\phi_{f_{G'}}(v) = \phi_{f_G}(v)$.

Wir definieren jetzt eine Eigenschaft für ein Knotenpaar u, v , die es uns erlaubt, die Bewertungen für die zwei Knoten u und v unabhängig voneinander zu berechnen. Das heißt, die Bewertung von u in G ist gleich der Bewertung von u in G ohne v , also in $G|_{V_G - \{v\}}$.

Definition 4.10 Sei $G = (V_G, E_G)$ ein gerichteter Graph und seien $u, v \in V_G$ zwei nicht-adjazente Knoten, das heißt weder (u, v) noch (v, u) ist eine Kante von G . Knoten u und Knoten v sind stark separierbar in G , wenn es zu jedem Paar von Wegen $p_1 = (u, w_1, \dots, w_k, v)$ von u nach v und $p_2 = (v, x_1, \dots, x_l, u)$ von v nach u einen stark zusammenhängenden Teilgraphen $S = (V_S, E_S)$ in $G|_{W \cup X}$ gibt, $W = \{w_1 \dots w_k\}$, $X = \{x_1, \dots, x_l\}$, so daß

1. $V_S \cap W \neq \emptyset$ und $V_S \cap X \neq \emptyset$ und
2. $W \cap X \subseteq V_S$ oder
- 2.' $V_S \cap W \cap X \neq \emptyset$ und es einen Knoten $y \in V_S \cap W \cap X$, $y = w_i = x_j$, gibt, so daß für jeden Knoten $z \in W \cap X$, $z = w_{i'} = x_{j'}$, mit $z \notin V_S$ gilt:
 $i' < i \Leftrightarrow j' > j$.

Wir sagen der stark zusammenhängende Teilgraph V_S separiert die Knoten u und v . Da stark separierbare Knoten nicht adjazent sind, enthalten die Mengen W und X jeweils mindestens einen Knoten. Ist $W \cap X \neq \emptyset$, so gehören alle Knoten aus $W \cap X$, die in p_1 in umgekehrter Richtung wie in p_2 durchlaufen werden, immer zu derselben Zusammenhangskomponente in

$G|_{W \cup X}$. Wichtig für die Frage des Zusammenhangs sind also in erster Linie die Knoten in $W \cap X$, die in p_1 in derselben Reihenfolge durchlaufen werden wie in p_2 . Ist $W \cap X = \emptyset$, dann ist Bedingung 2. erfüllt. Daß die beiden Bedingungen 2. und 2'. für $W \cap X \neq \emptyset$ äquivalent sind, zeigt die folgende Bemerkung.

Bemerkung 4.11 Sei $G = (V_G, E_G)$ ein gerichteter Graph und $u, v \in V_G$ seien zwei nicht-adjazente Knoten, $p_1 = (u, w_1, \dots, w_k, v)$ ein Weg von u nach v , $p_2 = (v, x_1, \dots, x_l, u)$ ein Weg von v nach u , $W = \{w_1 \dots w_k\}$, $X = \{x_1, \dots, x_l\}$ und $W \cap X \neq \emptyset$. Dann sind die folgenden Aussagen äquivalent:

1. Es gibt eine starke Zusammenhangskomponente $S = (V_S, E_S)$ in $G|_{W \cup X}$ mit $W \cap X \subseteq V_S$.
2. Es gibt einen stark zusammenhängenden Teilgraphen $T = (V_T, E_T)$ in $G|_{W \cup X}$ mit $V_T \cap W \cap X \neq \emptyset$ und einen Knoten $y \in V_T \cap W \cap X$, $y = w_i = x_j$, so daß für jeden Knoten $z \in (W \cap X) - V_T$, $z = w_{i'} = x_{j'}$ gilt:
 $i' < i$ und $j' > j$ oder $i' > i$ und $j' < j$.

Es gilt dann $V_T \subseteq V_S$.

Beweis: 1. \Rightarrow 2. gilt offensichtlich. 2. \Rightarrow 1.: Sei Teilgraphen $T = (V_T, E_T)$ wie in 2. beschrieben und $y \in V_T \cap W \cap X$, $y = w_i = x_j$. Sei $z \in (W \cap X) - V_T$, $z = w_{i'} = x_{j'}$. Dann gilt nach Voraussetzung $i' < i$ und $j' > j$ oder $i' > i$ und $j' < j$. Sei ohne Beschränkung der Allgemeinheit $i' > i$ und $j' < j$. Dann ist der Teilgraph, der durch die Knotenmengen $V_T \cup \{w_i, \dots, w_{i'} = x_{j'}, \dots, x_j\}$ induziert wird, stark zusammenhängend. Daraus folgt, daß alle Knoten aus $W \cap X$ in $G|_{W \cup X}$ zu derselben Zusammenhangskomponente gehören. \square

Das folgende Lemma verdeutlicht die Struktur von Graphen, die stark separierbare Knoten enthalten.

Lemma 4.12 Sei $G = (V_G, E_G)$ ein gerichteter, stark zusammenhängender Graph und $V_H, V_J \subseteq V_G$ seien zwei Knotenmengen, so daß

1. $V_H \cup V_J = V_G$ und
2. für jede Kante $(v_1, v_2) \in E_G$ beide Knoten in V_H oder in V_J , oder in beiden Mengen sind.

Sei $H = G|_{V_H}$, $J = G|_{V_J}$, und $I = G|_{V_H \cap V_J}$, $V_I = V_H \cap V_J$, und jedes Knotenpaar $u \in V_H - V_J$, $v \in V_J - V_H$ sei stark separierbar in G .

Dann gibt es für jedes Knotenpaar $u \in V_H - V_J$, $v \in V_J - V_H$ zu jedem Paar von Wegen $p_1 = (u = w_0, w_1, \dots, w_k, w_{k+1} = v)$ von u nach v und $p_2 = (v = x_0, x_1, \dots, x_l, x_{l+1} = u)$ von v nach u einen stark zusammenhängenden Teilgraphen $S = (V_S, E_S)$ in $G|_{W \cup X}$, $W = \{w_1, \dots, w_k\}$, $X = \{x_1, \dots, x_l\}$, so daß

1. $V_S \subseteq (W \cup X) \cap V_I$
2. $V_S \cap W \neq \emptyset$ und $V_S \cap X \neq \emptyset$ sowie

3. wenn für einen Knoten $z \in (W \cap X) \cap V_I$, $z = w_{i'} = x_{j'}$, für jeden Knoten $y \in V_S \cap W \cap X$, $y = w_i = x_j$, $y \neq z$, gilt, daß $i' < i$ und $j' < j$ beziehungsweise $i' > i$ und $j' > j$, dann ist $z \in V_S$.

Es gibt also für die Knoten u und v einen zusammenhängenden Teilgraphen S , der vollständig in I enthalten ist und die Knoten u und v separiert.

Beweis: Wir betrachten nur einfache Wege. Die durch zwei Wege gebildeten Kreise müssen allerdings nicht einfach sein. Sei $p_1 = (u = w_0, w_1, \dots, w_k, w_{k+1} = v)$ von u nach v und $p_2 = (v = x_0, x_1, \dots, x_l, x_{l+1} = u)$ von v nach u ein Paar von Wegen in G . Ohne Beschränkung der Allgemeinheit kann angenommen werden, daß es Zahlen $m, n \in \{1, \dots, k\}$, $m \leq n$, gibt, so daß $w_1, \dots, w_{m-1} \in V_H - V_J$, $w_m, \dots, w_n \in V_H \cap V_J$ und $w_{n+1}, \dots, w_k \in V_J - V_H$, sowie Zahlen $o, p \in \{1, \dots, l\}$, $o \leq p$, so daß $x_1, \dots, x_{o-1} \in V_J - V_H$, $x_o, \dots, x_p \in V_H \cap V_J$ und $x_{p+1}, \dots, x_l \in V_H - V_J$. Betrachte die beiden stark separierbaren Knoten w_{m-1} und w_{n+1} und die Wege $p'_1 = (w_{m-1}, \dots, w_{n+1})$ und $p'_2 = (w_{n+1}, \dots, w_k, x_0, \dots, x_l, w_0, \dots, w_{m-1})$. Dabei nehmen wir ohne Beschränkung der Allgemeinheit an, daß p'_2 ein einfacher Weg ist. Wir wollen zeigen, daß es einen stark zusammenhängenden Teilgraphen gibt, der die Knoten w_{m-1} und w_{n+1} separiert und die Anforderungen erfüllt. Damit separiert er auch die Knoten u und v . Sei $W' = \{w_m, \dots, w_n\} \subseteq V_H \cap V_J$. Da wir nur einfache Wege betrachten, kann W' nur gemeinsame Knoten mit X haben. Wir unterscheiden die folgenden Fälle:

1. $W' \cap X \neq \emptyset$
2. $W' \cap X = \emptyset$

Sei also $S = (V_S, E_S)$ ein stark zusammenhängender Teilgraph, der die beiden Knoten w_{m-1} und w_{n+1} separiert.

1. Fall: Betrachte einen separierenden Teilgraphen S , der alle Knoten aus $W' \cap X \subseteq V_I$ enthält und in Bezug auf die Knoten, die nicht in V_I sind, minimal ist. Es gilt $W' \cap X \subseteq V_I$, da $W' \subseteq V_I$. Liegt der Teilgraph S , der w_{m-1} und w_{n+1} separiert, vollständig in I , ist nichts mehr zu zeigen. Wenn nicht, dann gibt es keinen stark zusammenhängenden Teilgraphen in I , der alle Knoten aus $W' \cap X$ enthält. Wir betrachten die drei folgenden Situationen:

- i) $|W' \cap X| = 1$. Dann erfüllt dieser Knoten die geforderten Bedingungen für den Teilgraphen S , also $V_S = W' \cap X$.
- ii) $|W' \cap X| > 1$ und es gibt einen Knoten $y \in W' \cap X$, $y = w_i = x_j$, so daß für jeden Knoten $z \in W \cap X$, $z = w_{i'} = x_{j'}$, entweder $i < i'$ und $j > j'$ oder $i > i'$ und $j < j'$ gilt. Dann kann man analog zum Beweis von Bemerkung 4.11 einen stark zusammenhängenden Teilgraphen konstruieren, der die Bedingungen erfüllt.
- iii) $|W' \cap X| > 1$, es gibt keinen Knoten wie in ii) beschrieben und V_S enthält mindestens einen Knoten $u' \notin V_I$. Dann gibt es zwei Knoten $y \in W' \cap X$, $y = w_i = x_j$, $z \in W \cap X$, $z = w_{i'} = x_{j'}$, die in $S|_I$ nicht in derselben Zusammenhangskomponente sind, und für die entweder $i < i'$ und $j < j'$ oder $i > i'$ und $j > j'$ gelten muß. Sei ohne Beschränkung der Allgemeinheit $i < i'$ und $j < j'$. Dann gibt es einen in Bezug auf die Anzahl der Knoten in $V_G - V_I$ minimalen einfachen Weg p von z nach y in S , der y , z und einen Knoten u' enthält, der nicht in V_I ist (siehe Abbildung 4.4).

Sei ohne Beschränkung der Allgemeinheit $u' \in V_J$, und zwischen z und u' liegen in p nur Knoten aus V_I . Wir konstruieren aus p'_1 , p'_2 und p zwei neue Wege: Konstruiere einen neuen

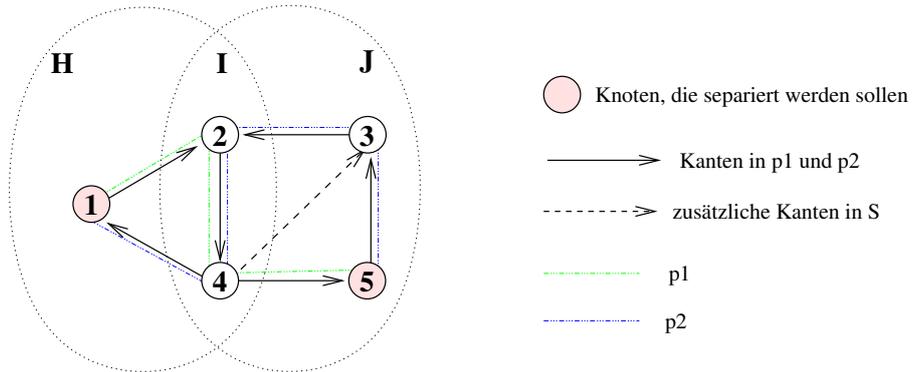


Abbildung 4.4: Beispiel für Fall 1, iii) von Beweis 4.12 mit $p_1 = (v_1, v_2, v_4, v_5)$ und $p_2 = (v_5, v_3, v_2, v_4, v_1)$

Weg \hat{p}'_2 , der Knoten aus $V_J - V_H$ und $V_H - V_J$ enthält: von u' nach $y = x_i$ wie in p , dann $x_{i+1}, \dots, x_k, v, w_1, \dots, w_{m-1}$ und \hat{p}'_1 von $w_{m+1} \dots w_j = x_i, \dots, x_{i'}$, und dann von $x_{i'} = z$ nach u' wie in p . Dann enthält \hat{p}'_2 weniger Knoten aus $V_G - V_I$ als p'_2, \hat{p}'_1 nur Knoten aus V_I und die beiden Knoten u' und w_{m-1} sind stark separierbar. Dann kann ein separierender Teilgraph gefunden werden, der auch die Knoten y und z enthält, und \hat{p}'_2 kann nach demselben Verfahren wie oben weiter in Bezug auf die Knoten, die nicht in I liegen, verkürzt werden. Dies führt zu einem Widerspruch.

2. Fall: Betrachte die einfachen Kreise in S , die mindestens einen Knoten aus p'_1 und einen Knoten aus p'_2 enthalten und somit auch eine Kante von einem Knoten aus p'_1 zu einem Knoten aus p'_2 sowie eine Kante von einem Knoten aus p'_2 zu einem Knoten aus p'_1 enthalten. Gibt es einen Kreis K in S , für den alle Knoten in I liegen und K somit mindestens einen Knoten aus $\{w_m, \dots, w_n\}$ und mindestens einen Knoten aus $\{x_o, \dots, x_p\}$ enthält, sind wir fertig. Ansonsten enthält jeder Kreis mindestens einen Knoten, der nicht aus I ist und es liegt einer der folgenden drei Fälle vor (siehe auch Abbildung 4.5):

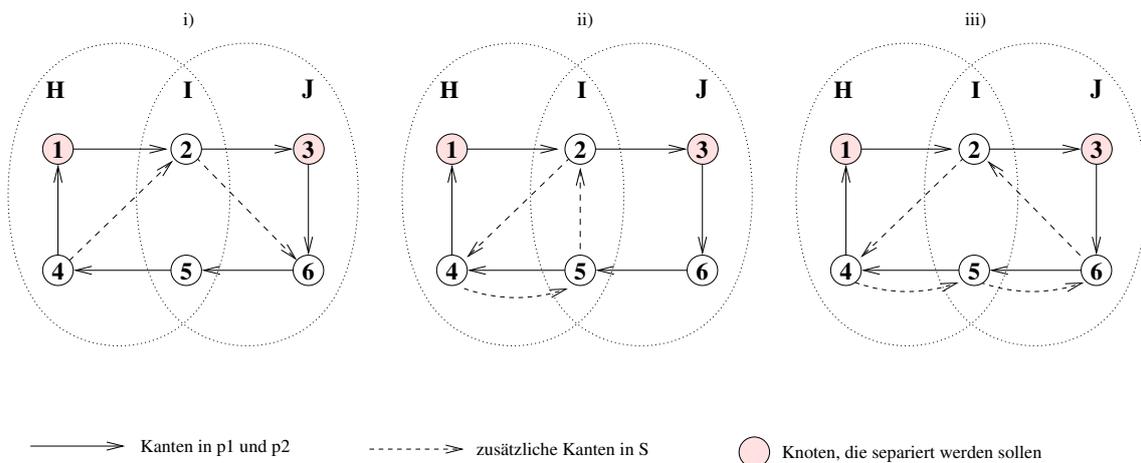


Abbildung 4.5: Beispiele für die drei Möglichkeiten aus Fall 2 von Beweis 4.12 mit jeweils $p_1 = (v_1, v_2, v_3)$ und $p_2 = (v_3, v_4, v_5, v_6, v_1)$

i) es gibt in K eine Kante (u', w_i) oder eine Kante (w_i, v') mit $w_i \in \{w_m, \dots, w_n\}$, $u' \in \{x_{p+1}, \dots, x_l, w_0, \dots, w_{m-2}\}$, $v' \in \{w_{n+2}, \dots, w_k, x_0, \dots, x_{o-1}\}$. Dann kann man die Wege p'_1 und p'_2 verkürzen. Sei zum Beispiel (u', w_i) in E_G mit $u' \in \{w_0, \dots, w_{m-2}\}$. Dann erhält man die Wege $\hat{p}'_1 = (u', w_i, \dots, w_{n+1})$ und $\hat{p}'_2 = w_{n+1} \dots w_k, x_0, \dots, x_l, w_0, \dots, u'$.

ii) es gibt einen Kreis K in S , der eine Kante (w_i, x_j) oder eine Kante (x_j, w_i) , mit $w_i \in \{w_m, \dots, w_n\}$ und $x_j \in \{x_o, \dots, x_p\}$, sowie einen Knoten $u' \in \{x_{p+1}, \dots, x_l, w_0, \dots, w_{m-2}\}$ oder $v' \in \{w_{n+2}, \dots, w_k, x_0, \dots, x_{o-1}\}$ enthält. Auch hier lassen sich die Wege \hat{p}'_1 und \hat{p}'_2 konstruieren: Zum Beispiel den Fall, daß K ein Kreis mit der Kante (x_j, w_i) ist, betrachte den Knoten u' , der der letzte Knoten in K vor der Kante (x_j, w_i) ist, der nicht in V_I liegt: Konstruiere dann \bar{p}'_1 durch u' bis x_j aus K , dann w_i, w_i, \dots, w_{n+1} und \bar{p}'_2 durch $w_{n+1}, \dots, w_k, x_0, \dots, x_j$ und wieder w_j bis u' aus K . Damit kann jeder stark zusammenhängende Teilgraph, der u' und w_{n+1} separiert, zu einem stark zusammenhängenden Teilgraphen erweitert werden, der auch x_j, w_i enthält und damit die Bedingungen für den ursprünglichen Kreis aus p_1, p_2 erfüllt. Der neue Kreis ist zwar nicht notwendigerweise kürzer als der Kreis aus p'_1 und p'_2 , enthält aber weniger Knoten aus $V_G - V_I$. Er entspricht Bedingung iii) im 1. Fall.

iii) Wenn keiner der anderen beiden Fälle auftritt, muß es einen Kreis K geben, der die Kanten (v', w_i) und (w_j, u') mit $w_i, w_j \in \{w_m, \dots, w_n\}$, $u' \in \{x_{p+1}, \dots, x_l, w_0, \dots, w_{m-2}\}$, $v' \in \{w_{n+2}, \dots, w_k, x_0, \dots, x_{o-1}\}$ enthält. Dieser Kreis K muß auch mindestens einen Knoten aus $\{x_o, \dots, x_p\}$ enthalten. Jeder Teilgraph, der die Knoten u' und v' in diesem Kreis K separiert, separiert auch die Knoten w_{m-1} und w_{n+1} im ursprünglichen Kreis.

In allen Fällen wird ein Kreis mit weniger Knoten aus $V_G - V_I$ konstruiert, der immer noch zwei stark separierbare Knoten enthält sowie Knoten aus $\{w_m, \dots, w_n\}$ und aus $\{x_o, \dots, x_p\}$, die auf den beiden Wegen dazwischen liegen. Entweder ist dieses Ergebnis bereits der gesuchte Graph, oder das beschriebene Verfahren muß auf das Ergebnis erneut angewandt werden. Da der Graph nur endlich viele Knoten besitzt, kann das Verfahren nur endlich oft iteriert werden. Das Ergebnis des letzten Iterationsschrittes muß der gesuchte Teilgraph sein. \square

Mit Hilfe dieser Eigenschaft läßt sich das nächste Lemma zeigen, welches deutlich macht, wie sich die starke Separierbarkeit von Knoten auf die Berechnung der Funktion f_G auswirkt.

Lemma 4.13 *Sei $G = (V_G, E_G)$ ein gerichteter Graph und $V_H, V_J \subseteq V_G$ zwei Knotenmengen, so daß $V_H \cup V_J = V_G$ ist und für jede Kante $(v_1, v_2) \in E_G$ sind beide Knoten in V_H oder in V_J oder in beiden Mengen. Sei $H = G|_{V_H}$, $J = G|_{V_J}$, und $I = G|_{V_H \cap V_J}$. Wenn jedes Knotenpaar $u \in V_H - V_J$, $v \in V_J - V_H$ stark separierbar in G ist, dann gilt für jede Knotenmenge $V' \subseteq V_G$*

$$f_G(V') = f_H(V' \cap V_H) + f_J(V' \cap V_J) - f_I(V' \cap V_I).$$

Beweis: Sei $V' \subseteq V_G$ eine Knotenmenge von G . Betrachte den folgenden ungerichteten Graphen $T = (V_T, E_T)$ mit Knotenmenge

$$V_T = \text{SCC}(H|_{V'}) \cup \text{SCC}(J|_{V'})$$

und Kantenmenge E_T , so daß zwei Knoten von V_T genau dann durch eine ungerichtete Kante verbunden sind, wenn die beiden starken Zusammenhangskomponenten mindestens einen gemeinsamen Knoten haben. Wenn zwei verschiedene starke Zusammenhangskomponenten aus der Knotenmenge V_T durch eine Kante in T verbunden sind, dann muß eine von ihnen

aus $\text{SCC}(H|_{V'})$ und die andere aus $\text{SCC}(J|_{V'})$ sein. Außerdem gibt es für jede starke Zusammenhangskomponente Z aus $\text{SCC}(I|_{V'})$ genau eine starke Zusammenhangskomponente Z_H in $\text{SCC}(H|_{V'})$ und genau eine starke Zusammenhangskomponente Z_J in $\text{SCC}(J|_{V'})$ und die gemeinsamen Knoten von Z_H und Z_J sind genau die Knoten von Z .

Da die Vereinigung von zwei stark zusammenhängenden Teilgraphen, die einen gemeinsamen Knoten haben, wieder stark zusammenhängend ist und jedes Paar von Knoten $u \in V_H - V_J$, $v \in V_J - V_H$ in G stark separierbar ist, folgt mit Lemma 4.12, daß der ungerichtete Graph T keine Kreise enthält: T ist also kreisfrei und somit ein Wald.

Die Anzahl der Zusammenhangskomponenten in T (also die Anzahl der Bäume in T) ist äquivalent zu der Anzahl von starken Zusammenhangskomponenten in $G|_{V'}$. Die Anzahl der Zusammenhangskomponenten in einem Wald ist gleich der Anzahl der Knoten minus der Anzahl der Kanten. Sei $CC(T)$ die Menge der Zusammenhangskomponenten in T . Da T genau eine Kante für jede starke Zusammenhangskomponente von $\text{SCC}(I|_{V'})$ und genau einen Knoten für jede starke Zusammenhangskomponente in $\text{SCC}(H|_{V'})$ und $\text{SCC}(J|_{V'})$ hat, erhalten wir

$$|\text{SCC}(G|_{V'})| = |CC(T)| = |\text{SCC}(H|_{V'})| + |\text{SCC}(J|_{V'})| - |\text{SCC}(I|_{V'})|,$$

also

$$f_G(V') = f_H(V' \cap V_H) + f_J(V' \cap V_J) - f_I(V' \cap V_I).$$

□

Der folgende Satz gibt an, wie die Bewertung von Knoten aus G durch die Bewertungen von denselben Knoten in gewissen Teilgraphen von G berechnet werden kann.

Satz 4.14 *Sei $G = (V_G, E_G)$ ein gerichteter Graph und $V_H, V_J \subseteq V_G$ seien zwei Knotenmengen, so daß $V_H \cup V_J = V_G$ ist und für jede Kante $(v_1, v_2) \in E_G$ beide Knoten in V_H oder in V_J oder in beiden Mengen sind. Sei $H = G|_{V_H}$, $J = G|_{V_J}$ und $I = G|_{V_H \cap V_J}$. Wenn jedes Knotenpaar $u \in V_H - V_J$, $v \in V_J - V_H$ stark separierbar in G ist, dann gilt*

1. für jeden Knoten $w \in V_H \cap V_J$: $\phi_{f_G}(w) = \phi_{f_H}(w) + \phi_{f_J}(w) - \phi_{f_I}(w)$,
2. für jeden Knoten $w \in V_H - V_J$: $\phi_{f_G}(w) = \phi_{f_H}(w)$ und
3. für jeden Knoten $w \in V_J - V_H$: $\phi_{f_G}(w) = \phi_{f_J}(w)$.

Beweis: Sei w ein Knoten von V_G . Nach Lemma 4.13 gilt für jede Knotenmenge $V' \subseteq V_G$:

$$\begin{aligned} f_G(V' \cup \{w\}) - f_G(V') &= (f_H((V' \cup \{w\}) \cap V_H) - f_H(V' \cap V_H)) \\ &+ (f_J((V' \cup \{w\}) \cap V_J) - f_J(V' \cap V_J)) \\ &- (f_I((V' \cup \{w\}) \cap V_I) - f_I(V' \cap V_I)). \end{aligned}$$

Wenn $w \in V_H \cap V_J$, dann erhalten wir mit Lemma 4.9

$$\phi_{f_G}(w) = \phi_{f_H}(w) + \phi_{f_J}(w) - \phi_{f_I}(w).$$

Wenn w ein Knoten aus $V_H - V_J$ ist, dann gilt $(V' \cup \{w\}) \cap V_J = V' \cap V_J$ und $(V' \cup \{w\}) \cap V_I = V' \cap V_I$ und somit

$$f_G(V' \cup \{w\}) - f_G(V') = f_H((V' \cup \{w\}) \cap V_H) - f_H(V' \cap V_H),$$

4.4 Das Entscheidungsproblem „starke Separierbarkeit von zwei Knoten“ ist NP-vollständig

was mit Lemma 4.9 impliziert:

$$\phi_{f_G}(w) = \phi_{f_H}(w).$$

Wenn w ein Knoten aus $V_J - V_H$ ist, dann folgt mit einer analogen Argumentation $\phi_{f_G}(w) = \phi_{f_J}(w)$. \square

Abbildung 4.6 zeigt ein Beispiel für einen gerichteten Graphen G , in dem alle Bewertungen $\phi_{f_G}(v)$ mit Satz 4.14 aus den Bewertungen für die drei Teilgraphen H , I und J berechnet wurden.

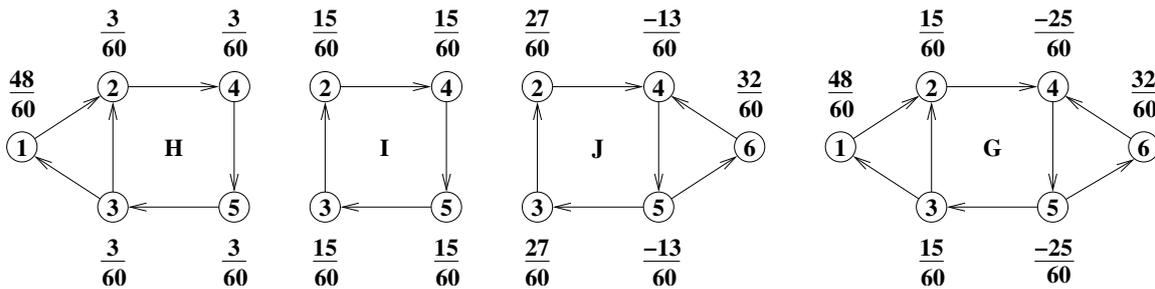


Abbildung 4.6: Vier Graphen $G = (V_G, E_G)$, $H = G|_{V_H}$, $J = G|_{V_J}$ und $I = G|_{V_H \cap V_J}$, für die $V_H, V_J \subseteq V_G$, $V_H \cup V_J = V_G$, und für jede Kante $(v_1, v_2) \in E_G$ beide Knoten in V_H oder in V_G oder in beiden Mengen sind. Das Knotenpaar $v_1 \in V_H - V_J$, $v_6 \in V_J - V_H$ ist stark separierbar in G .

4.4 Das Entscheidungsproblem „starke Separierbarkeit von zwei Knoten“ ist NP-vollständig

Um die Formel aus Satz 4.14 zu verwenden, muß man Knotenmengen in einem Graphen finden, die jeweils stark separierbar sind. Wir werden jetzt zeigen, daß das Problem zu entscheiden, ob zwei Knoten in einem gerichteten Graphen nicht stark separierbar sind, NP-vollständig ist. Daraus folgt, daß es keinen Polynomzeitalgorithmus gibt, der entscheidet, ob zwei Knoten in einem Graphen nicht stark separierbar sind, es sei denn $P = NP$. Die NP-Schwere folgt aus einer einfachen Reduktion des Satisfiability-Problems. Eine Einführung in die Theorie der NP-Vollständigkeit findet man zum Beispiel in [GJ79].

Wir werden die folgenden Begriffe verwenden: Sei $X = \{x_1, \dots, x_n\}$ eine Menge von *booleschen Variablen*. Eine Wahrheitswertbelegung für X ist eine Funktion $t : X \rightarrow \{\text{wahr}, \text{falsch}\}$. Ist $t(x_i) = \text{wahr}$, dann sagen wir, die Variable x_i ist *wahr* unter t ; Ist $t(x_i) = \text{falsch}$, dann sagen wir, die Variable x_i ist *falsch* unter t . Ist x_i eine Variable von X , dann sind x_i und \bar{x}_i *Literale* über X . Literal x_i ist genau dann *wahr* unter t wenn die Variable x_i unter t *wahr* ist; Literal \bar{x}_i ist genau dann *wahr* unter t , wenn die Variable x_i unter t *falsch* ist. Eine *Klausel* C über X ist eine Menge von Literalen über X . Eine Klausel C wird von einer Belegung t für X *erfüllt*, wenn C mindestens ein Literal enthält, das unter t *wahr* ist. Eine Menge \mathcal{C} von Klauseln über X ist genau dann *erfüllbar*, wenn es eine Wahrheitswertbelegung t gibt, die jede Klausel von \mathcal{C} erfüllt.

Das Satisfiability-Problem, bezeichnet mit SAT, wird folgendermaßen definiert: Gegeben eine

Menge X von Variablen und eine Menge \mathcal{C} von Klauseln über X . Gibt es eine erfüllende Belegung für \mathcal{C} ?

Dieses Problem ist NP-vollständig, selbst wenn jede Klausel von \mathcal{C} nur genau drei Literale enthält.

Satz 4.15 *Das Problem zu entscheiden, ob zwei Knoten u, v eines gerichteten Graphen G nicht stark separierbar sind, ist NP-vollständig.*

Beweis: Zuerst zeigen wir, daß das Problem in NP liegt. Zwei Knoten u und v sind nicht stark separierbar in G , wenn sie die Bedingungen aus Definition 4.10 nicht erfüllen.

Wir können also nicht-deterministisch alle Paare von Wegen $p_1 = (u, w_1, \dots, w_k, v)$ von u nach v und $p_2 = (v, x_1, \dots, x_l, u)$ von v nach u betrachten. Dann können wir für jede starke Zusammenhangskomponente $S = (V_S, E_S)$ von $G|_{W \cup X}$, $W = \{w_1 \dots w_k\}$, $X = \{x_1, \dots, x_l\}$ in Polynomzeit verifizieren, ob $V_S \cap W \neq \emptyset$, $V_S \cap X \neq \emptyset$ und $W \cap X \subseteq V_S$. Also gehört das Problem zu entscheiden, ob zwei Knoten u, v nicht stark separierbar sind, zu NP.

Die NP-Schwere folgt aus einer einfachen Transformation von 3-SAT. Sei $X = \{x_1, \dots, x_n\}$ eine Menge boolescher Variablen und $\mathcal{C} = \{C_1, \dots, C_m\}$ eine Menge von m Klauseln. Wir definieren einen Graphen $G(X, \mathcal{C})$ mit zwei Knoten u und v , so daß es genau dann eine Wahrheitswertbelegung t für X gibt, die jede Klausel von \mathcal{C} erfüllt, wenn u und v in $G(X, \mathcal{C})$ nicht stark separierbar sind. Abbildung 4.7 zeigt ein Beispiel der Konstruktion für vier Variablen x_1, x_2, x_3, x_4 und vier Klauseln $\{x_2, x_3, \bar{x}_4\}$, $\{x_1, \bar{x}_2, x_4\}$, $\{\bar{x}_1, \bar{x}_2, \bar{x}_3\}$, $\{\bar{x}_1, x_2, x_3\}$.

Graph $G(X, \mathcal{C})$ hat sechs Knoten u, a, b, v, c, d , zwei Literalknoten x_i, \bar{x}_i für jede Variable x_i , $1 \leq i \leq n$, und drei Literalknoten $c_{j,1}, c_{j,2}, c_{j,3}$ für jede Klausel $C_j = \{c_{j,1}, c_{j,2}, c_{j,3}\}$, $1 \leq j \leq m$. $G(X, \mathcal{C})$ hat die Kanten (u, a) , (a, x_1) , (a, \bar{x}_1) , die Kanten (x_i, x_{i+1}) , (x_i, \bar{x}_{i+1}) , (\bar{x}_i, x_{i+1}) , $(\bar{x}_i, \bar{x}_{i+1})$ für $i = 1, \dots, n-1$, die Kanten (x_n, b) , (\bar{x}_n, b) , (b, v) , (v, c) , $(c, c_{1,1})$, $(c, c_{1,2})$, $(c, c_{1,3})$, die Kanten $(c_{j,k}, c_{j+1,l})$ für $j = 1, \dots, m-1$ und $k, l \in \{1, 2, 3\}$, und die Kanten $(c_{m,1}, d)$, $(c_{m,2}, d)$, $(c_{m,3}, d)$, (d, u) , und (d, a) . Zusätzlich gibt es sogenannte *Brückenkanten* von jedem Literalknoten x_i (\bar{x}_i) für Variable x_i zu jedem Literalknoten \bar{x}_i (beziehungsweise x_i) für eine Klausel. In Abbildung 4.7 sind die Brückenkanten gestrichelt gezeichnet.

Jedes Paar von Wegen zwischen den Knoten u und u in $G(X, \mathcal{C})$, besteht aus zwei knotendisjunkten Wegen $p_1 = (u, a, \dots, b, v)$ und $p_2 = (v, c, \dots, d, u)$. Weg p_1 durchläuft für jede Variable genau einen Literalknoten und definiert so eine Wahrheitswertbelegung t für die Variablen. Weg p_2 durchläuft für jede Klausel genau einen Literalknoten. Wegen der Kante (d, a) gibt es genau dann einen u und v separierenden zusammenhängenden Teilgraphen in dem durch die Knoten der beiden Wege ohne u und v induzierten Teilgraphen, also einen Teilgraphen der die Bedingung 1. aus Definition 4.10 erfüllt, wenn es eine Brückenkante von einem Literalknoten in p_1 zu einem Literalknoten in p_2 gibt.

Angenommen es gibt eine Wahrheitswertbelegung t für X , die jede Klausel erfüllt. Dann gibt es zwei Wege $p_1 = (u, a, \dots, b, v)$ und $p_2 = (v, c, \dots, d, u)$, so daß der durch die Knoten der Wege p_1 und p_2 ohne u und v induzierte Teilgraph keine Brückenkanten enthält, beispielsweise zwei Wege $p_1 = (u, a, \dots, b, v)$ und $p_2 = (v, c, \dots, d, u)$, die außer u, v, a, b beziehungsweise u, v, c, d nur *wahren* Literalknoten durchlaufen. In diesem Fall gibt es keinen stark zusammenhängenden Teilgraphen, in dem durch die Knoten der beiden Wege ohne u und v induzierten Teilgraphen, der aus beiden Wegen mindestens einen Knoten enthält. Also sind u und v nicht stark separierbar.

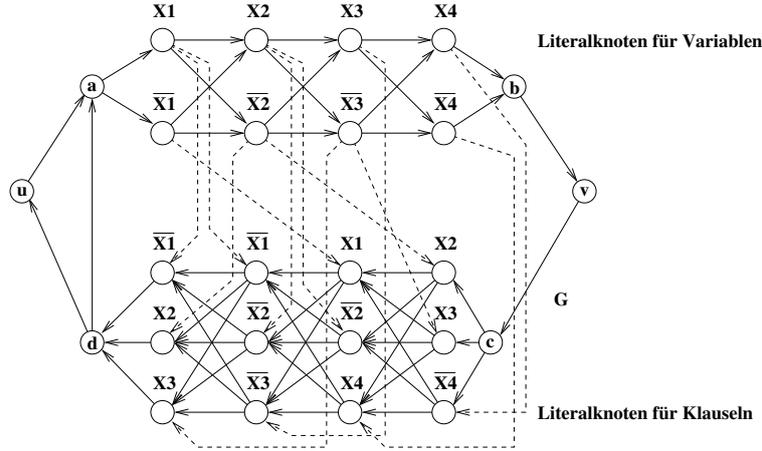


Abbildung 4.7: Der Graph $G(X, \mathcal{C})$ für $X = \{x_1, x_2, x_3, x_4\}$ und $\mathcal{C} = \{ \{x_2, x_3, \bar{x}_4\}, \{x_1, \bar{x}_2, x_4\}, \{\bar{x}_1, \bar{x}_2, \bar{x}_3\}, \{\bar{x}_1, x_2, x_3\} \}$.

Angenommen es gibt keine Wahrheitswertbelegung für X , die jede Klausel erfüllt. Dann hat jeder Teilgraph, der durch die Knoten von einem Paar von Wegen $p_1 = (u, a, \dots, b, v)$ und $p_2 = (v, c, \dots, d, u)$ induziert wird, mindestens eine Brückenkante (u', v') . In diesem Fall bildet der Kreis bestehend aus einem Teilweg von p_1 von a bis u' und einem Teilweg von p_2 von v' bis d durch die Kante (d, a) und die Brückenkante (u', v') einen stark zusammenhängenden Teilgraphen der aus jedem der beiden Wege Knoten enthält. Also sind u und v stark separierbar. \square

Theorem 4.15 kann verwendet werden, um zu zeigen, daß das Entscheidungsproblem, ob zwei Knoten eine unterschiedliche Knotenbewertung durch ϕ_{f_G} erhalten, NP-schwer ist. Betrachte wieder den Graphen $G(X, \mathcal{C})$ mit den zwei Knoten u und v konstruiert für die Instanz (X, \mathcal{C}) von 3-SAT wie im Beweis von 4.15. Sei $G'(X, \mathcal{C})$ der Graph $G(X, \mathcal{C})$ ohne den Knoten v und seine inzidenten Kanten. Dann ist $\phi_{f_{G(X, \mathcal{C})}}(u) = \phi_{f_{G'(X, \mathcal{C})}}(u)$, wenn u und v stark separierbar in G sind, und $\phi_{f_{G(X, \mathcal{C})}}(u) < \phi_{f_{G'(X, \mathcal{C})}}(u)$, wenn u und v nicht stark separierbar in G sind.

Satz 4.16 *Das Problem zu entscheiden, ob $\phi_{f_G}(u) < \phi_{f_G}(v)$ für zwei Knoten u, v eines gerichteten Graphen G , ist NP-schwer.*

Somit kann ein Algorithmus zur Berechnung von ϕ_{f_G} verwendet werden, um ein NP-schweres und auch ein co-NP-schweres Problem zu entscheiden.

4.5 Knotenbewertungen für ungerichtete Graphen

Die Knotenbewertung ϕ_{f_G} für gerichtete Graphen kann leicht für ungerichtete Graphen erweitert werden. Für einen ungerichteten Graph G sei $\text{dir}(G)$ der gerichtete Graph, der entsteht, wenn wir jede ungerichtete Kante $\{u, v\}$ durch zwei gerichtete Kanten (u, v) und (v, u) ersetzen. Sei f_G die Funktion von den Teilmengen von V_G in die reellen Zahlen \mathbb{R} , die jeder Teilmenge $V' \subseteq V_G$ die Anzahl der Zusammenhangskomponenten in dem Teilgraphen von G

zuordnet, der durch die Knoten von V' induziert wird. Das heißt, die Bewertung eines Knotens v in einem ungerichteten Graphen G ist gleich der Bewertung von v in dem zugehörigen gerichteten Graphen $\text{dir}(G)$.

Für ungerichtete Graphen erhalten wir die folgenden Grenzen in Bezug auf die Knotenbewertung ϕ_{f_G} .

Lemma 4.17 *Sei $G = (V_G, E_G)$ ein zusammenhängender, ungerichteter Graph mit $|V_G| \geq 2$. Dann gilt für jeden Knoten $v \in V_G$*

$$1 - \frac{|V_G| - 1}{2} \leq \phi_{f_G}(v) \leq \frac{1}{2}.$$

Beweis: Der Beweis der unteren Schranke entspricht dem Beweis von Lemma 4.8. Es bleibt also die obere Schranke zu zeigen. Sei Π_{V_G} die Menge aller Anordnungen $\pi : V_G \rightarrow \{1, \dots, |V_G|\}$. Es gilt wieder für einen Knoten $v \in V$

$$1 \geq f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v))$$

für alle Anordnungen $\pi \in \Pi$, wobei $\lambda(\pi, v) = \{w \in V_G \mid \pi(w) < \pi(v)\}$ wieder die Menge aller $w \in V_G$ ist, die durch π links von v angeordnet werden. Es gilt

$$1 = f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v))$$

genau dann, wenn $\lambda(\pi, v)$ keine Nachbarn von v enthält und

$$0 \geq f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v)),$$

wenn $\lambda(\pi, v)$ mindestens einen Nachbarn von v enthält. Daraus folgt, daß die Bewertung eines Knotens umso höher ist, je weniger Nachbarn er hat. Da G zusammenhängend ist, hat jeder Knoten mindestens einen Nachbarn. Die Anzahl der Anordnungen $\pi \in \Pi_{V_G}$, in denen ein fester Knoten $w \in V_G$ links von einem Knoten $v \in V_G$ angeordnet wird, ist genau $\frac{1}{2}|\Pi_{V_G}| = \frac{|V_G!|}{2}$. Damit erhalten wir:

$$\begin{aligned} \phi_f(v) &= \frac{1}{|V_G!|} \sum_{\pi \in \Pi_{V_G}} (f_G(\lambda(\pi, v) \cup \{v\}) - f_G(\lambda(\pi, v))) \\ &\leq \frac{1}{|V_G!|} \left(\frac{|V_G!|}{2} \cdot 1 + \frac{|V_G!|}{2} \cdot 0 \right) \\ &= \frac{1}{2} \end{aligned}$$

□

Abbildung 4.8 zeigt ein Beispiel für die Knotenbewertung ϕ_{f_G} für einen ungerichteten Graphen G mit Knotenmenge $V_G = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$.

Es ist leicht zu sehen, daß zwei Knoten u, v von $\text{dir}(G)$ genau dann nicht stark separierbar sind, wenn G einen Kreis ohne Chord besitzt, der u und v enthält. Ein *Chord* für einen Kreis $c = (u_1, \dots, u_k)$ ist eine Kante $\{u_i, u_j\}$, so daß $2 \leq |i - j| \leq k - 2$. Das Entscheidungsproblem, ob ein ungerichteter Graph G einen Kreis ohne Chord enthält, kann in Linearzeit gelöst werden [Dir61, RTL76, TY84]. Dies ist das bekannte Erkennungsproblem für chordale Graphen. Ein Graph G ist ein *chordaler Graph* wenn jeder Kreis von G mit einer Länge von mindestens vier

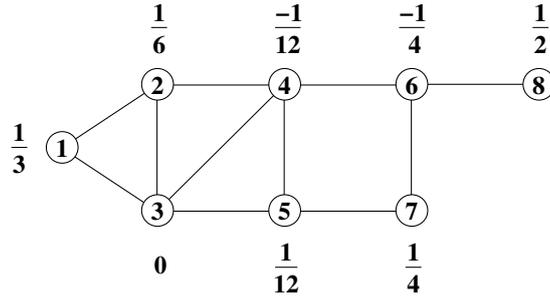


Abbildung 4.8: Die Knotenbewertung ϕ_{f_G} für einen ungerichteten Graph G mit acht Knoten.

einen Chord enthält oder, alternativ, wenn G keinen Kreis ohne Chord enthält, siehe [HS58]. Das Problem zu entscheiden, ob G einen Kreis ohne Chord mit einer Länge von $k \geq 5$ enthält, kann in $O(|V_G| + |E_G|^2)$ Zeit auf $O(|V_G| \cdot |E_G|)$ Platz gelöst werden, siehe [NP04].

Satz 4.14 angewendet auf ungerichtete Graphen entspricht dem folgenden Satz, der eine allgemeinere Version von Proposition 2 von [GGAM⁺04] ist.

Satz 4.18 Sei $G = (V_G, E_G)$ ein ungerichteter Graph und seien $V_H, V_J \subseteq V_G$ zwei Knotenmengen, so daß $V_H \cup V_J = V_G$ ist und für jede Kante $\{v_1, v_2\} \in E_G$ gilt beide Knoten in V_H oder in V_J oder in beiden Mengen sind. Sei $H = G|_{V_H}$, $J = G|_{V_J}$, und $I = G|_{V_H \cap V_J}$. Wenn G keinen Kreis ohne Chord mit einem Knoten aus $V_H - V_J$ und einem Knoten aus $V_J - V_H$ besitzt, dann gilt

1. für jeden Knoten $w \in V_H \cap V_J$: $\phi_{f_G}(w) = \phi_{f_H}(w) + \phi_{f_J}(w) - \phi_{f_I}(w)$,
2. für jeden Knoten $w \in V_H - V_J$: $\phi_{f_G}(w) = \phi_{f_H}(w)$ und
3. für jeden Knoten $w \in V_J - V_H$: $\phi_{f_G}(w) = \phi_{f_J}(w)$.

Angenommen, ein zusammenhängender Graph G kann in $k > 1$ Zusammenhangskomponenten G_1, \dots, G_k zerlegt werden, indem man einen vollständigen Teilgraphen I von G entfernt. Graph I heißt *Cliquenseparator* von G . Sei $G'_i = G|_{V_{G_i} \cup V_I}$ für $i = 1, \dots, k$ der Teilgraph von G , der von den Knoten der Zusammenhangskomponente G_i und den Knoten des entfernten Teilgraphen I induziert wird. Dann ist nach Satz 4.18 für jedes $i = 1, \dots, k$ für einen Knoten w von G'_i die Knotenbewertung $\phi_{f_G}(w) = \phi_{f_{G'_i}}(w)$ und für einen Knoten w aus I die Knotenbewertung

$$\phi_{f_G}(w) = \left(\sum_{j=1}^k \phi_{f_{G'_j}}(w) \right) - (k-1) \cdot \phi_{f_I}(w).$$

Tarjan führte in [Tar85] eine Zerlegung von Graphen definiert durch einen Cliquenseparator ein, die zu einer baumstrukturierten Zerlegung bis zu Teilgraphen, die keine weiteren Cliquenseparatoren enthalten, führt. Die verbleibenden Teilgraphen heißen Atomgraphen. Eine solche Zerlegung kann für einen Graphen G in Zeit $O(|V_G| \cdot |E_G|)$ gefunden werden. Die obige Diskussion impliziert, daß die Knotenbewertungen $\phi_{f_G}(v)$ von cliquenseparierbaren Graphen in Polynomzeit berechnet werden können, wenn die Bewertungen in den Atomgraphen in Polynomzeit berechnet werden können.

In den nächsten drei Abschnitten wollen wir Graphklassen untersuchen, bei denen eine effiziente Berechnung der Knotenbewertung durch Zerlegung möglich ist. Bei den cycle-composed und den chordalen Graphen werden dazu Cliquenseparatoren verwendet.

4.5.1 Cycle-composed Graphen

Ein Beispiel für eine Graphklasse für die die Bewertung ϕ_{f_G} effizient berechenbar ist, ist die Klasse der *cycle-composed* Graphen, die wie folgt rekursiv definiert werden können.

Definition 4.19 : Cycle-composed Graph

1. Der Kreis C_n mit $n \geq 3$ Knoten ist cycle-composed.
2. Sei $G = (V_G, E_G)$ ein cycle-composed Graph und $e_1 = \{u_1, v_1\}$ eine Kante von G . Sei $C_n = (V_{C_n}, E_{C_n})$ ein Kreis mit $n \geq 3$ Knoten und sei $e_2 = \{u_2, v_2\}$ eine Kante von C_n . Dann ist der Graph, der durch eine knotendisjunkte Vereinigung von G und C_n entsteht, wenn man u_2 mit u_1 und v_2 mit v_1 identifiziert, cycle-composed.

Der zusammengesetzte Graph hat die Knotenmenge $V_G \cup V_{C_n} - \{u_2, v_2\}$ und die Kantenmenge $\{\{h(u), h(v)\} \mid \{u, v\} \in E_G \cup E_{C_n}\}$, wobei $h(u) = u$ für jedes $u \in V_G \cup V_{C_n} - \{u_2, v_2\}$, und $h(u_2) = u_1$ und $h(v_2) = v_1$.

Cycle-composed Graphen sind zweifach zusammenhängend und haben eine Baumweite von höchstens 2, siehe [RS86, Bod98] für eine Definition von Baumweite. Graphen mit einer Baumweite von höchstens 2 können in Linearzeit erkannt werden, indem man Knoten mit einem Grad von höchstens 2 entfernt. Wenn ein Knoten u mit einem Grad von 2 entfernt wird, werden die beiden Nachbarn von u mit einer Kante verbunden, wenn sie nicht schon adjazent sind. Ein Graph hat genau dann Baumweite 2, wenn er durch das Entfernen von Knoten von Grad 2 auf die eben beschriebene Art vollständig abgebaut werden kann, siehe zum Beispiel [Wie87].

Sei \mathcal{C} die Knotenmenge der Kreise, die verwendet wurden, um den cycle-composed Graphen G aufzubauen. \mathcal{C} enthält also eine Knotenmenge C für jeden Kreis, der verwendet wurde, um G aufzubauen. Die Knotenbewertung $\phi(u)$ für alle Knoten u von G ist durch die folgende einfache Prozedur in Linearzeit berechenbar.

1. für alle $u \in V$ {
2. sei $\phi_{f_G}(u) := (\deg(u) - 2) * (-\frac{1}{2});$ }
3. für alle $C \in \mathcal{C}$ {
4. für alle $u \in C$ {
5. sei $\phi_{f_G}(u) := \phi_{f_G}(u) + \frac{1}{|C|};$ }

In diesem Fall bestehen die separierenden Cliques jeweils aus zwei Knoten, die innerhalb dieses Teilgraphen beide die Bewertung $\frac{1}{2}$ erhalten. Ein solcher separierender Teilgraph besteht immer aus zwei Knoten, die an derselben Knotenidentifikation beteiligt sind und der inzidenten Kante. Da jeder Knoten u an $\deg(u) - 2$ Knotenidentifikationen beteiligt ist, kann

die Bewertung von u mit $(\deg(u) - 2) * (-\frac{1}{2})$ initialisiert werden. Danach addiert der Algorithmus für jeden Kreis C den Bruch $\frac{1}{|C|}$ zum Anteil jedes Knotens in C , da in einem Kreis C jeder Knoten die Bewertung $\frac{1}{|C|}$ hat. Da die Anzahl der Knoten in den Mengen von \mathcal{C} der Summe $\sum_{C \in \mathcal{C}} |C| = 2|E_G| - |V_G|$ entspricht, kann die Bewertung für alle Knoten in einem cycle-composed Graphen in Linearzeit berechnet werden, wenn \mathcal{C} gegeben ist.

Die Knotenmengen der Kreise können mit dem folgenden Algorithmus berechnet werden. Wir nehmen an, daß zu jeder Kante eine leere Knotenliste initial zugeordnet wird. Das heißt, jede Kante $\{u, v\}$ wird initial durch ein Paar $(\{u, v\}, \emptyset)$ repräsentiert. Eine Kante $e = (\{u, v\}, L)$ mit einer nicht-leeren Knotenliste L repräsentiert einen Pfad zwischen u und v , der alle Knoten in L durchläuft. Wenn G einen Knoten vom Grad 2 besitzt, dessen beide Nachbarn u, v nicht adjazent sind, können wir u und seine beiden inzidenten Kanten $(\{u, v\}, L_1)$, $(\{u, w\}, L_2)$ entfernen und eine neue Kante $(\{v, w\}, L_1 \cup L_2 \cup \{u\})$ zwischen u und v einfügen.

Wenn G einen Knoten u vom Grad 2 besitzt, dessen beide Nachbarn v, w adjazent sind, dann können die Knoten eines Kreises ausgegeben werden. Seien $(\{u, v\}, L_1)$, $(\{u, w\}, L_2)$, $(\{v, w\}, L_3)$ die Kanten zwischen den drei Knoten u, v und w . Dann gibt der Algorithmus die Knotenmenge $L_1 \cup L_2 \cup L_3 \cup \{u, v, w\}$ aus. Wenn der Graph G keine weiteren Kanten als die drei obigen hat, dann sind alle Kreise ausgegeben und der Algorithmus endet. Wenn der Graph G noch weitere Kanten besitzt und L_3 nicht leer ist, dann ist der Graph nicht cycle-composed. In jedem anderem Fall entfernt der Algorithmus nach der Ausgabe des Kreises die beiden Kanten $(\{u, v\}, L_1)$, $(\{u, w\}, L_2)$ und arbeitet weiter. Wenn dieser Prozeß abbricht, weil es keine weiteren Knoten vom Grad 2 gibt, dann ist der Graph auch nicht cycle-composed.

Dieser Algorithmus berechnet die Knotenmengen aller Kreise, die verwendet wurden, um einen cycle-composed Graph aufzubauen. Die Laufzeit des Algorithmus ist in $O(|V_G|^2)$, da wir für jeden Knoten überprüfen müssen, ob seine beiden Nachbarn adjazent sind. Dieses Problem kann aber durch einen einfachen Trick umgangen werden, der auch in [Wie87] für das Erkennen von außenplanaren Graphen verwendet wurde. Der Trick ist, erst dann zu überprüfen, ob zwei Nachbarn v, w von u adjazent sind, wenn einer dieser beiden Knoten v, w einen Grad von 2 oder weniger erhält. In dieser Situation kann der Test in einer festen Anzahl von Schritten ausgeführt werden, und dann wird entweder eine neue Kante eingefügt oder ein Kreis ausgegeben. Mit dieser Modifikation erhält man einen Linearzeitalgorithmus für die Berechnung aller Kreise eines cycle-composed Graphen. Das folgende Beispiel zeigt ein mögliche Implementierung.

```

erstelle-neue-Kante (Knoten  $u$ )
{
  seien  $e_1 = (\{u, v\}, L_1)$ ,  $e_2 = (\{u, w\}, L_2) \in E_G$  die beiden mit  $u$  inzidenten Kanten;
  füge  $(\{v, w\}, L_1 \cup L_2 \cup \{u\})$  in  $E_{\text{neu}}$  ein;
  entferne  $e_1$  und  $e_2$  aus  $E_G$ ;
  wenn  $(\deg_{E_G}(v) = 2)$  dann
    füge  $v$  in  $M$  ein;
  wenn  $(\deg_{E_G}(w) = 2)$  dann
    füge  $w$  in  $M$  ein;
}

```

verschiebe-neue-Kante (Kante $e_{\text{neu}} = (\{u, v\}, L_{\text{neu}})$)

```

{
  wenn es eine Kante  $e = (\{u, v\}, L) \in E_G$  gibt, dann {
    gib  $L \cup L_{\text{neu}} \cup \{u, v\}$  aus;
    entferne  $e_{\text{neu}}$  aus  $E_{\text{neu}}$ ;
    wenn  $(|E_G| = 1)$  und  $(|E_{\text{neu}}| = 0)$  dann
      halt „alle Kreise ausgegeben“;
    sonst wenn  $(L \neq \emptyset)$  dann
      halt „ $G$  ist nicht cycle-composed“;
  }
  sonst {
    entferne  $e_{\text{neu}}$  aus  $E_{\text{neu}}$ ;
    füge  $e_{\text{neu}}$  in  $E_G$  ein;
    wenn  $(\deg_{E_G}(u) = 3)$  dann
      entferne  $u$  aus  $M$ ;
    wenn  $(\deg_{E_G}(v) = 3)$  dann
      entferne  $v$  aus  $M$ ; }
}

```

berechne-Kreise (Graph $G = (V_G, E_G)$)

```

{
  sei  $M := \emptyset$ ;
  für alle  $u \in V_G$  {
    wenn  $(\deg_{E_G}(u) = 2)$  dann {
      füge  $u$  in  $M$  ein; } }
  solange  $(M \neq \emptyset)$  {
    sei  $u \in M$ ;
    wenn es eine Kante  $e_{\text{neu}} \in E_{\text{neu}}$  inzident mit  $u$  gibt, dann
      verschiebe-neue-Kante ( $e_{\text{neu}}$ );
    sonst {
      wenn  $(\deg_{E_G}(u) = 2)$  dann
        erstelle-neue-Kante ( $u$ );
      entferne  $u$  aus  $M$ ; } }
  halt „ $G$  ist nicht cycle-composed“;
}

```

Der obige Algorithmus speichert in einer Menge M alle Knoten vom Grad 2. Dabei ist zu beachten, daß der Grad eines Knotens immer durch die Kanten in E_G bestimmt ist. Für jeden Knoten u , der mit genau zwei Knoten v und w adjazent ist, wird eine neue Kante in eine Menge, bezeichnet mit E_{neu} , eingefügt, aber noch nicht in die Kantenmenge E_G des Graphen G . Wenn ein Knoten u aus M für die Verarbeitung in Betracht kommt, wird zunächst überprüft, ob es Kanten inzident mit u in der Menge E_{neu} gibt. Wenn E_{neu} eine Kante e enthält, die inzident mit u ist, dann wird e entweder in E_G eingefügt (wenn die beiden Knoten nicht durch eine Kante in E_G adjazent sind), oder ein Kreis wird ausgegeben (wenn die beiden Knoten von e durch eine Kante in E_G adjazent sind). Der Test, ob die beiden Knoten von e adjazent sind, kann in Zeit $O(1)$ durchgeführt werden, da u einer der beiden Endknoten von e ist und den Grad 2 hat. Das beweist den folgenden Satz.

Satz 4.20 Die Knotenbewertung $\phi_{f_G}(u)$ für alle Knoten u eines cycle-composed Graphen G ist in Linearzeit berechenbar.

4.5.2 Chordale Graphen

Die Knotenbewertung ϕ_{f_G} ist auch für *chordale Graphen* in Linearzeit berechenbar. Eine interessante Charakterisierung von chordalen Graphen ist die Existenz einer *perfekten Eliminationsordnung*.

Definition 4.21 Sei $p = (u_1, \dots, u_n)$ eine Ordnung der $|V_G| = n$ Knoten von $G = (V_G, E_G)$, und sei $N(G, p, i)$ für $i = 1, \dots, n$ die Menge der Nachbarn u_j des Knoten u_i mit $i < j$,

$$N(G, p, i) := \{u_j \mid \{u_i, u_j\} \in E_G \wedge i < j\}.$$

Die Knotenordnung $p = (u_1, \dots, u_n)$ wird *perfekte Eliminationsordnung (PEO)* genannt, wenn die Knoten aus $N(G, p, i)$ für $i = 1, \dots, n-1$ einen vollständigen Teilgraphen induzieren.

Dirac [Dir61], Fulkerson und Gross [FG65], und Rose [Ros70] haben gezeigt, daß ein Graph G genau dann chordal ist, wenn er eine perfekte Eliminationsordnung besitzt. Rose, Tarjan und Lueker haben in [RTL76] gezeigt, daß eine perfekte Eliminationsordnung in Linearzeit gefunden werden kann, wenn sie existiert. Wird eine perfekte Eliminationsordnung $p = (v_1, \dots, v_n)$ der Knoten von $G = (V_G, E_G)$ gegeben, dann kann die Knotenbewertung ϕ_{f_G} nach Satz 4.14 und mit dem folgenden Algorithmus berechnet werden. Die separierenden Cliques sind hier die durch die Knoten aus $N(G, p, i)$ induzierten vollständigen Teilgraphen. Man beachte, daß in einem vollständigen Graphen mit n Knoten $\phi_{f_G}(v) = \frac{1}{n}$ für jeden Knoten von G ist, da G V_G -symmetrisch ist.

1. Sei $\phi_{f_G}(v_n) := 1$;
2. für $i = n-1, \dots, 1$ {
3. sei $\phi_{f_G}(v_i) := \frac{1}{|N(G, p, i)|+1}$;
4. für alle $v \in N(G, p, i)$ {
5. sei $\phi_{f_G}(v) := \phi_{f_G}(v) + \frac{1}{|N(G, p, i)|+1} - \frac{1}{|N(G, p, i)|}$; }

Die Laufzeit dieses Algorithmus ist linear in der Größe von G , weil die Zuordnung in Zeile 3 genau $|V_G| - 1$ mal und die Zuordnung in Zeile 5 genau $|E_G|$ mal durchgeführt wird. Da die perfekte Eliminationsordnung in Linearzeit gefunden werden kann, erhalten wir das folgende Theorem.

Satz 4.22 Die Knotenbewertung $\phi_{f_G}(v)$ für alle Knoten v eines chordalen Graphen G ist in Linearzeit berechenbar.

4.5.3 Module-composed Graphen

Ein weiteres Beispiel für eine Graphklasse für die die Bewertung ϕ_{f_G} effizient berechenbar ist, ist die Klasse der *bipartiten module-composed Graphen*, die rekursiv so in zwei disjunkte

Teilgraphen zerlegt werden können, daß die Knoten aller Kanten zwischen diesen zwei Teilgraphen einen vollständigen bipartiten Graphen G' induzieren. Um diese Graphen genau zu definieren, benötigen wir vorher noch die folgenden Definitionen.

Definition 4.23 Eine Knotenmenge $V' \subseteq V_G$ heißt eine unabhängige Menge von G , wenn es keine Kante zwischen zwei Knoten von V' gibt.

Ein Graph G heißt bipartit, wenn die Knotenmenge V_G von G in zwei unabhängige Mengen $V_1 \subseteq V_G$ und $V_2 = V_G - V_1$ geteilt werden kann.

Für einen Knoten $u \in V_G$ sei $N(u)$ die Menge seiner Nachbarn, also aller Knoten $v \in V_G$, so daß $\{u, v\}$ eine Kante in E_G ist. Eine Knotenmenge $M \subseteq V_G$ heißt ein Modul von G , wenn für alle Knoten $u, v \in M$ gilt, $N(u) - M = N(v) - M$.

Somit haben alle Knoten eines Moduls M dieselben Nachbarn außerhalb von M . Damit können wir nun auch die nächste Definition formulieren.

Definition 4.24 : Module-composed Graph Ein Graph $G = (V_G, E_G)$ ist module-composed, wenn die $|V_G| = n$ Knoten von V_G in einer Folge u_1, \dots, u_n so angeordnet werden können, so $M_1 = \emptyset$ und für $i = 2, \dots, n$ die Knotenmenge

$$M_i = \{u_j \mid \{u_j, u_i\} \in E_G, j < i\}$$

ein Modul in dem Teilgraph ist, der durch die Knoten u_1, \dots, u_{i-1} induziert wird. Die Folge

$$(M_1, u_1), \dots, (M_n, u_n)$$

heißt eine Modul-Aufbaufolge.

Eine Modul-Aufbaufolge bestimmt einen Graphen eindeutig. Allerdings hat ein module-composed Graph nicht unbedingt nur eine eindeutige Modul-Aufbaufolge.

Beschäftigen wir uns zunächst mit den Eigenschaften von module-composed Graphen. Dazu betrachten wir die folgenden Graphen. (Siehe Abbildung 4.9 für Illustrationen.)

1. Ein Kreis ohne Chord mit einer Länge ≥ 5 heißt *Hole*.
2. Der Graph mit 5 Knoten u_1, \dots, u_5 und 6 Kanten $\{u_1, u_2\}, \{u_2, u_3\}, \{u_3, u_4\}, \{u_4, u_1\}, \{u_3, u_5\}, \{u_5, u_4\}$ heißt *House*.
3. Der Graph mit 5 Knoten u_1, \dots, u_5 und 7 Kanten $\{u_1, u_2\}, \{u_2, u_3\}, \{u_3, u_4\}, \{u_4, u_5\}, \{u_5, u_1\}, \{u_2, u_5\}, \{u_3, u_5\}$ heißt *Gem*.
4. Und der Graph mit 6 Knoten u_1, \dots, u_6 und 7 Kanten $\{u_1, u_2\}, \{u_2, u_3\}, \{u_3, u_4\}, \{u_4, u_5\}, \{u_5, u_6\}, \{u_6, u_1\}, \{u_2, u_5\}$ heißt *Domino*.

Wir sagen ein Graph ist hole-frei, house-frei, gem-frei, oder domino-frei, wenn er keinen induzierten Teilgraph besitzt, der ein Hole, House, Gem beziehungsweise Domino ist.

Lemma 4.25 Sei $(M_1, u_1), (M_2, u_2), \dots, (M_n, u_n)$ eine Modul-Aufbaufolge für einen Graphen $G = (V_G, E_G)$ mit n Knoten. Es gilt:

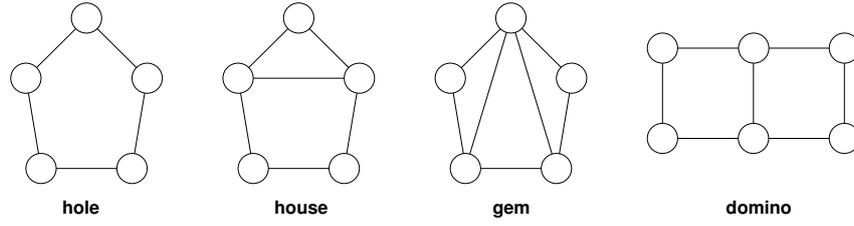


Abbildung 4.9: Ein Hole mit 5 Knoten, das House, der Gem, und der Domino.

1. Die Menge aller module-composed Graphen ist bezüglich induzierter Teilgraphen abgeschlossen.
2. Jeder module-composed Graph ist hole-frei, house-frei, und domino-frei.
3. Wenn die Module M_2, \dots, M_n alle nicht leer sind, dann ist G zusammenhängend.
4. Der Graph G ist genau dann bipartit, wenn alle Module M_i , $1 \leq i \leq n$, unabhängige Mengen sind.

Beweis:

1. Wenn M ein Modul eines Graphen G ist, dann ist für jedes $u \in M$ auch $M - \{u\}$ ein Modul im Graphen $G|_{V_G - \{u\}}$.
2. Holes, das House sowie der Domino sind nicht module-composed. Also können sie auch keine induzierten Teilgraphen von module-composed Graphen sein.
3. Sei $G_i = G|_{\{u_1, \dots, u_i\}}$, $1 \leq i \leq n$, der Teilgraph von G , der von den Knoten u_1, \dots, u_i induziert wird. Wenn G_i , $1 \leq i < n$, zusammenhängend und M_{i+1} nicht leer ist, dann ist auch G_{i+1} zusammenhängend.
4. Wenn G bipartit ist, dann ist jeder Modul M_i eine unabhängige Menge, weil u_i mit allen Knoten in M_i verbunden ist. Angenommen, alle Module M_i , $1 \leq i \leq n$, sind unabhängige Mengen und G_i ist bipartit für ein $1 \leq i < n$. Wenn $|M_{i+1}| \leq 1$ oder wenn die Knoten von M_{i+1} keine Nachbarn in G_i haben, dann ist G_{i+1} auch bipartit. Andernfalls besteht M_{i+1} aus mindestens zwei Knoten, die mindestens einen gemeinsamen Nachbarn haben. In diesem Fall gehören alle Knoten von M_{i+1} auf dieselbe Seite jeder Bipartition von G_i , also ist G_{i+1} bipartit.

□

Definition 4.26 Ein zusammenhängender Graph G heißt *distanzerhaltend*, wenn für jedes Knotenpaar $u, v \in V_G$ der Abstand zwischen u und v in jedem induzierten zusammenhängenden Teilgraphen von G , der u und v enthält, derselbe ist.

Wenn G distanzerhaltend ist, dann gibt es eine *Pruning-Sequenz*, die G definiert, siehe [BM86, HM90]. Eine Pruning-Sequenz für einen Graphen $G = (V_G, E_G)$ mit n Knoten ist eine Folge von geordneten Paaren

$$(v_1, l_1), \dots, (v_n, l_n),$$

wobei l_i entweder das Symbol \bullet oder ein Paar aus $\{v_1, \dots, v_{i-1}\} \times \{\text{leaf}, \text{true-twin}, \text{false-twin}\}$ ist.

Ein Graph ist durch eine Pruning-Sequenz wie folgt definiert: Sei G_i , $1 \leq i \leq n$ der Graph, der durch die Pruning-Sequenz $(v_1, l_1), \dots, (v_i, l_i)$ definiert ist. Dann erhält man den Graphen G_i aus dem Graphen G_{i-1} , indem man den Knoten v_i und Kanten entsprechend der folgenden Regeln einfügt:

1. Ist $l_i = \bullet$, dann werden keine Kanten eingefügt.
2. Ist $l_i = (v_j, \text{leaf})$, dann wird v_i mit v_j verbunden.
3. Ist $l_i = (v_j, \text{false-twin})$, dann wird v_i mit allen Nachbarn von v_j verbunden.
4. Ist $l_i = (v_j, \text{true-twin})$, dann wird v_i mit v_j und allen Nachbarn von v_j verbunden.

Ein bipartiter, distanzerhaltender Graph G besitzt immer eine Pruning-Sequenz ohne true twins, da eine true-twin-Operation $(v_i, (v_j, \text{true-twin}))$ immer einen Kreis erzeugt, wenn v_j mindestens einen Nachbarn hat. Wenn v_j keinen Nachbarn hat, dann kann $(v_i, (v_j, \text{true-twin}))$ durch $(v_i, (v_j, \text{leaf}))$ ersetzt werden.

Lemma 4.27 *Sei G ein bipartiter Graph. Dann sind folgende Aussagen äquivalent:*

1. G ist module-composed.
2. G ist domino-frei und hole-frei.
3. G ist distanzerhaltend.

Beweis:

1.) \Rightarrow 2.) Siehe Lemma 4.25.

2.) \Rightarrow 3.) Wenn G bipartit ist, dann ist er house-frei und gem-frei. Jeder domino-freie, house-freie, gem-freie und hole-freie Graph ist distanzerhaltend, siehe [BM86, HM90].

3.) \Rightarrow 1.) Ist G bipartit, dann ist es leicht, eine Modul-Aufbaufolge für G aus einer Pruning-Sequenz $(v_1, l_1), \dots, (v_n, l_n)$ für G zu definieren. Sei S_i , $1 \leq i \leq n$, die Modul-Aufbaufolge, die aus der Pruning-Sequenz $(v_1, l_1), \dots, (v_i, l_i)$ gewonnen wurde. Dann ist $S_1 = (\emptyset, v_1)$ und S_i , $i > 1$, erhält man aus S_{i-1} folgendermaßen:

Ist $l_i = \bullet$, dann wird S_{i-1} am Ende erweitert durch (\emptyset, v_i) . Ist $l_i = (v_j, \text{leaf})$, dann wird S_{i-1} am Ende erweitert durch $(\{v_j\}, v_i)$. Ist $l_i = (v_j, \text{false-twin})$, dann sei (M, u) das Paar für den Knoten $u = v_j$ in der Modul-Aufbaufolge S_{i-1} . Das Paar (M, v_i) wird in S_{i-1} direkt hinter (M, u) eingefügt. \square

Da eine Pruning-Sequenz für distanzerhaltende Graphen in Linearzeit berechenbar ist (siehe [BM86, HM90]), ist auch eine Modul-Aufbaufolge ein Linearzeit berechenbar.

Als nächstes zeigen wir, wie man die Knotenbewertung $\phi_{f_G}(u)$ für die Knoten u eines vollständigen bipartiten Graphen $G = (V_G, E_G)$ berechnet.

Definition 4.28 Sei $G = (V_G, E_G)$ mit $V_{G_1} \subseteq V_G$, $V_{G_2} = V_G - V_{G_1}$, $E_G = \{\{u, v\} | u \in V_{G_1}, v \in V_{G_2}\}$, $n = |V_{G_1}|$ und $m = |V_{G_2}|$. Dann ist G ein vollständiger bipartiter Graph, der auch mit $K_{n,m}$ bezeichnet wird.

In diesen Graphen lassen sich die Knotenbewertungen durch eine einfache Formel bestimmen.

Lemma 4.29 Sei $G = K_{n,m}$. Für einen Knoten $u \in V_{G_1}$ kann die Knotenbewertung $\phi_{f_G}(u)$ durch

$$\begin{aligned} \phi_{f_G}(u) &= \sum_{U \subseteq V_{G_1} - \{u\}} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} + \sum_{U \subseteq V_{G_2}, U \neq \emptyset} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} \cdot (1 - |U|) \\ &= \frac{1}{m+1} - \frac{(m-1)m}{n(n+1)(n+m)} \end{aligned} \tag{4.3}$$

berechnet werden.

Beweis: Die Knotenbewertung $\phi_{f_G}(u)$ für einen Knoten $u \in V_{G_1}$ wird nach Gleichung 4.1 durch

$$\phi_{f_G}(u) = \sum_{U \subseteq V_G, u \in U} \frac{(|U| - 1)! (|V_G| - |U|)!}{|V_G|!} (f_G(U) - f_G(U - \{u\}))$$

oder äquivalent durch

$$\phi_{f_G}(u) = \sum_{U \subseteq V_G - \{u\}} \frac{(|U|)! (n + m - |U| - 1)!}{(n + m)!} (f_G(U \cup \{u\}) - f_G(U))$$

definiert. Wir unterscheiden die folgenden drei Fälle:

1. Wenn $U \subseteq V_{G_1}$, dann ist $f_G(U \cup \{u\}) - f_G(U) = 1$,
2. wenn $U \subseteq V_{G_2}$ und $U \neq \emptyset$, dann ist $f_G(U \cup \{u\}) - f_G(U) = 1 - |U|$ und wenn $U \subseteq V_G$, $V_{G_1} \cap U \neq \emptyset$ und
3. wenn $V_{G_2} \cap U \neq \emptyset$, dann ist $f_G(U \cup \{u\}) - f_G(U) = 0$.

Damit erhalten wir:

$$\begin{aligned}
\phi_{f_1, G}(u) &= \sum_{U \subseteq N - \{u\}} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} + \sum_{U \subseteq M, U \neq \emptyset} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} \cdot (1 - |U|) \\
&= \sum_{U \subseteq N - \{u\}} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} - \sum_{U \subseteq M, U \neq \emptyset} \frac{|U|! (n+m-|U|-1)!}{(n+m)!} \cdot (|U| - 1) \\
&= \sum_{k=0}^{n-1} \binom{n-1}{k} \frac{k! (n+m-k-1)!}{(n+m)!} - \sum_{k=1}^m \binom{m}{k} \frac{k! (n+m-k-1)! (k-1)}{(n+m)!} \\
&= \frac{1}{(n+m)!} \sum_{k=0}^{n-1} \frac{(n-1)! k! (n+m-k-1)!}{k! (n-1-k)!} - \frac{1}{(n+m)!} \sum_{k=1}^m \frac{m! k! (n+m-k-1)! (k-1)}{k! (m-k)!} \\
&= \frac{(n-1)!}{(n+m)!} \sum_{k=0}^{n-1} \frac{(n+m-k-1)! m!}{(n-1-k)! m!} - \frac{m!}{(n+m)!} \sum_{k=1}^m \frac{(n+m-k-1)! (k-1) (n-1)!}{(m-k)! (n-1)!} \\
&= \frac{(n-1)! m!}{(n+m)!} \sum_{k=0}^{n-1} \binom{n+m-k-1}{m} - \frac{m! (n-1)!}{(n+m)!} \sum_{k=1}^m \binom{n+m-k-1}{n-1} (k-1) \\
&\stackrel{(*)}{=} \frac{(n-1)! m! (n+m)}{(n+m)! (m+1)} - \frac{m! (n-1)! (n+m-1)}{(n+m)! (n+1)} \\
&= \frac{(n-1)! m! (n+m)!}{(n+m)! (m+1)! (n-1)!} - \frac{m! (n-1)! (n+m-1)!}{(n+m)! (n+1)! (m-2)!} \\
&= \frac{1}{m+1} - \frac{(m-1)m}{n(n+1)(n+m)}
\end{aligned}$$

Gleichung (*) kann man in zwei Schritten zeigen.

Schritt 1. Da

$$\sum_{k=0}^{n-1} \binom{n+m-k-1}{m} = \sum_{k=0}^{n-1} \binom{m+k}{m},$$

zeigen wir durch Induktion über n , daß

$$\sum_{k=0}^{n-1} \binom{m+k}{m} = \binom{n+m}{m+1}.$$

Induktionsanfang: Sei $n = 1$.

$$\binom{m}{m} = 1 = \binom{1+m}{m+1}$$

Induktionsschritt: Sei $n > 1$.

$$\begin{aligned}
\sum_{k=0}^{n-1} \binom{m+k}{m} &= \binom{m+n-1}{m} + \sum_{k=0}^{n-2} \binom{m+k}{m} \\
&\stackrel{(1)}{=} \binom{m+n-1}{m} + \binom{n-1+m}{m+1} \\
&= \binom{n+m}{m+1}
\end{aligned}$$

(1) folgt aus der Induktionsannahme.

Schritt 2. Da

$$\sum_{k=1}^m \binom{n+m-k-1}{n-1} (k-1) = \sum_{k=0}^{m-2} \binom{n+k-1}{n-1} (m-k-1),$$

zeigen wir durch Induktion über m , daß

$$\sum_{k=0}^{m-2} \binom{n+k-1}{n-1} (m-k-1) = \binom{n+m-1}{n+1}.$$

Induktionsanfang: Sei $m = 2$.

$$\binom{n-1}{n-1} \cdot 1 = 1 = \binom{n+1}{n+1}$$

Induktionsschritt: Sei $m > 2$.

$$\begin{aligned} \sum_{k=0}^{m-2} \binom{n+k-1}{n-1} \cdot (m-k-1) &= \sum_{k=0}^{m-3} \binom{n+k-1}{n-1} \cdot (m-k-1) + \binom{n+m-3}{n-1} \\ &= \sum_{k=0}^{m-3} \binom{n+k-1}{n-1} \cdot (m-k-2) + \sum_{k=0}^{m-3} \binom{n+k-1}{n-1} + \binom{n+m-3}{n-1} \\ &\stackrel{(1)}{=} \binom{n+m-2}{n+1} + \sum_{k=0}^{m-3} \binom{n+k-1}{n-1} + \binom{n+m-3}{n-1} \\ &= \binom{n+m-2}{n+1} + \sum_{k=0}^{m-2} \binom{n+k-1}{n-1} \\ &\stackrel{(2)}{=} \binom{n+m-2}{n+1} + \binom{n-2+m}{n} \\ &= \binom{n+m-1}{n+1} \end{aligned}$$

(1) folgt aus der Induktionsannahme, (2) folgt aus der Gleichheit, die im 1. Schritt gezeigt wurde. \square

Dies zeigt, daß $\phi_{f_G}(u)$ für einen Knoten u in einem vollständig bipartiten Graphen in Polynomzeit berechenbar ist.

Sei $(M_1, u_1), (M_2, u_2), \dots, (M_n, u_n)$ eine Modul-Aufbaufolge für einen bipartiten Graphen G . Sei $G_i = (V_{G_i}, E_{G_i}) = G|_{\{u_1, \dots, u_i\}}$, $1 \leq i \leq n$, der Teilgraph von G , der durch die Knoten $\{u_1, \dots, u_i\}$ induziert wird. Die Knoten von M_i und $N_i = \{w \in V_{G_{i-1}} \mid \{w, v\} \in E_{G_{i-1}}, v \in M_i\}$ induzieren einen vollständigen bipartiten Graphen $K_{n,m}$ mit $n = |M_i|$ und $m = |N_i|$. Dieser Teilgraph trennt den Knoten u_i von allen Knoten aus $V_G - (N_i \cup M_i \cup \{u_i\})$ in G_i .

Somit hat jeder Kreis in G_i , der u_i und einen Knoten aus $V_G - (N_i \cup M_i \cup \{u_i\})$ enthält, auch mindestens zwei Knoten aus N_i und mindestens zwei Knoten aus M_i und somit mindestens zwei Chords. Also kann Satz 4.18 zur Berechnung der Knotenbewertungen verwendet werden.

Sei

$$h(n, m) = \frac{1}{m+1} - \frac{(m-1)m}{n(n+1)(n+m)}.$$

Mit dieser Bezeichnung, Satz 4.18 und Lemma 4.29 erhalten wir:

1. Die Bewertung von u_i in G_i ist $h(n+1, m)$.
2. Die Bewertungen der Knoten aus M_i in G_i unterscheiden sich von ihrer Bewertung in G_{i-1} durch die Differenz $h(m, n+1) - h(m, n)$.

3. Die Bewertungen der Knoten aus N_i in G_i unterscheiden sich von ihren Bewertungen in G_{i-1} durch die Differenz $h(n+1, m) - h(n, m)$.
4. Die Bewertungen aller anderen Knoten bleiben unverändert.

Der folgende Algorithmus berechnet die Knotenbewertungen für alle Knoten eines bipartiten module-composed Graphen in Polynomzeit. Dabei kann N_i in Zeit $O(|N_i|)$ bestimmt werden, weil M_i ein Modul von G_{i-1} ist und deshalb alle Knoten aus M_i dieselben Nachbarn in G_{i-1} haben. Die vollständige Berechnung ist in Zeit $O(|V_G|^2)$.

1. sei $\phi_{f_G}(u_1) := 1$;
2. für $i = 2, \dots, n$ {
3. sei $m = |M_i|$;
4. sei $N_i = \{w \in V_{G_{i-1}} \mid \{w, v\} \in E_{G_{i-1}}, v \in M_i\}$;
5. sei $n = |N_i|$;
6. sei $\phi_{f_G}(u_i) := h(n+1, m)$;
7. für alle $v \in M_i$ {
8. sei $\phi_{f_G}(v) := \phi_{f_G}(v) + h(m, n+1) - h(m, n)$; }
9. für alle $v \in N_i$ {
10. sei $\phi_{f_G}(v) := \phi_{f_G}(v) + h(n+1, m) - h(n, m)$; }

Damit ist der folgende Satz gezeigt.

Satz 4.30 *Die Knotenbewertungen $\phi_{f_G}(v)$ für alle Knoten v eines bipartiten, module-composed Graphen $G = (V_G, E_G)$ lassen sich in Zeit $O(|V_G|^2)$ berechnen.*

4.6 Berechnungen von Knotenbewertungen in neuronalen Netzwerken

In diesem Abschnitt werden Knotenbewertungen für zwei Netzwerke des Makakengehirns berechnet. Das eine repräsentiert die kortikalen Areale des visuellen Systems und das andere präfrontale kortikale Areale. Für die Berechnungen wird die in Abschnitt 4.3 definierte charakteristische Funktion f_G für gerichtete Graphen G verwendet. Die Knotenbewertungen werden im folgenden auch als Shapleywerte bezeichnet.

4.6.1 Knotenbewertungen im visuellen System

Dem ersten untersuchten Graphen liegt eine schon veröffentlichte Verbindungsmatrix des visuellen Cortex des Makaken von Young ([You92], basierend auf Daten die von Felleman und Van Essen zusammengestellt wurden [FE91]) zu Grunde. Der Graph besteht aus 30 disjunkten Arealen. Verbindungen zwischen diesen Arealen wurden durch binäre Einträge in der Verbindungsmatrix repräsentiert, wobei alle leeren Felder (unbekannter Verbindungsstatus) als 0 interpretiert wurden. Die Ergebnisse der Shapleywert-Berechnung für diesen Graphen

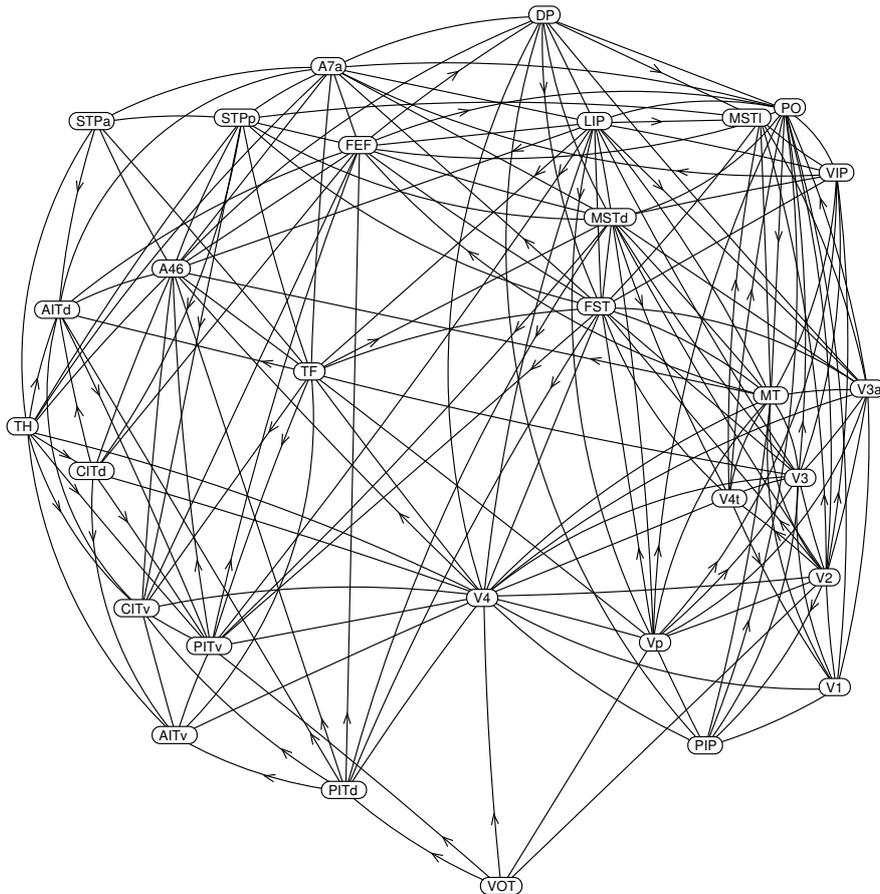


Abbildung 4.10: Der gerichtete Graph, der das visuelle System nach Young darstellt. Es wurden nur die Verbindungen eingezeichnet, die bekanntermaßen vorhanden sind.

findet man in Tabelle 4.1, die Verbindungsmatrix nach Young im Anhang A.1. Eine graphische Darstellung dieses Netzwerkes ist in Abbildung 4.10 zu sehen.

Da dieser Graph stark zusammenhängend ist, ist die Summe aller Knotenbewertungen nach Shapleys 2. Axiom gleich 1 und damit der durchschnittliche Shapleywert der Knoten $\frac{1}{30}$. Wie oben dargestellt, bedeutet eine numerisch kleinere Knotenbewertung einen größeren Einfluß auf den Zusammenhang, also die Erreichbarkeit anderer Knoten, wogegen ein hoher Wert darauf hinweist, daß der Knoten ohne große Auswirkungen auf die Struktur des verbleibenden Netzwerkes entfernt werden kann.

Der negativste Wert wurde für das Areal V4 berechnet, ein dicht vernetztes farbsensitives Areal im ventralen visuellen Verarbeitungs-Stream. Die beiden Areale mit den nächstniedrigsten Werten sind das frontale Augenfeld (FEF) und Area 46 (zentral-lateraler präfrontaler Cortex). Dieses Ergebnis sollte mit Vorsicht behandelt werden, da diese beiden Areale die einzigen präfrontalen Areale in diesem visuellen Netzwerk sind und somit eine spezielle Position innerhalb des Netzwerkes einnehmen. Andere Areale mit niedrigen Knotenbewertungen sind die am dichtesten verbundenen Strukturen des Occipitallappens (V2, V3, VP) und der dorsalen Stream Regionen (MSTd, 7a, MT, STPp, LIP, DP), sowie das temporale Areal TF.

$\phi_{f_{Y92}}(v)$					
V4	-0.380586	V3	-0.0227672	V1	0.090548
FEF	-0.149840	STPp	0.0036237	VIP	0.105730
A46	-0.135431	LIP	0.0120692	PIP	0.120754
V2	-0.104353	DP	0.0300926	AITd	0.157156
Vp	-0.094883	V3a	0.0359315	PITv	0.157774
MSTd	-0.087045	MSTl	0.0430511	STPa	0.160592
A7a	-0.085710	P0	0.0545006	V4t	0.170802
FST	-0.061905	TH	0.0678206	CITd	0.210864
MT	-0.047566	CITv	0.0858998	PITd	0.303108
TF	-0.045754	AITv	0.0877731	VOT	0.317751

Tabelle 4.1: Knotenbewertungen für die Areale des visuellen System basierend auf den Verbindungsdaten von Young [You92]

Der unerwartet hohe Wert von Areal V1 ist darauf zurückzuführen, daß sein subkortikaler Input in diesem Netzwerk nicht berücksichtigt wird.

4.6.2 Knotenbewertungen im präfrontalen Cortex

Das zweite untersuchte Netzwerk ist der Präfrontalcortex des Makaken, der ursprünglich von Walker definiert wurde [Wal40]. Es wurden hier für die zwölf dem präfrontalen Cortex angehörenden Areale drei verschiedene Versionen der Verbindungsmatrix verwendet. Diese Matrizen wurden aus Daten von anatomischen Tracing-Studien aus der CoCoMac Datenbank zusammengestellt. Die Tabellen mit den Verbindungsmatrizen befinden sich im Anhang (A.2, A.3, A.4, A.5). Für eine graphische Darstellung dieser Netzwerke siehe Abbildung 4.11.

Als erstes wurden die Shapleywerte für eine Verbindungsmatrix des präfrontalen Cortex berechnet, die auch schon in früheren Studien verwendet wurde ([KS03, PSK02, SKB⁺01]). Passingham et al. [PSK02] untersuchten dabei die Ähnlichkeiten von Knoten und die Clusterstrukturen in dieser Verbindungsmatrix und kamen dabei zu dem Schluß, daß jeder Knoten eine einmalige Menge von afferenten und efferenten Verbindungen hat. Sie unterschieden aufgrund von Gemeinsamkeiten in den Verbindungsmustern drei Gruppen von Arealen im präfrontalen Cortex: ein dicht vernetztes Cluster von orbitomedialen Arealen (Areas 10-14, 25), eine Gruppe von lateralen präfrontalen Arealen (45, 46, 8A), sowie eine dritte, dicht vernetzte Gruppe aus den übrigen Arealen (8B, 9, 24), deren Verbindungsmuster ungefähr gleich viele Ähnlichkeiten mit jeder der beiden anderen Gruppen hatten und die zwischen den anderen beiden Gruppen zu vermitteln schienen.

Wie in dieser Untersuchung wurde auch in der vorliegenden Arbeit zunächst angenommen, daß die Abwesenheit von Verbindungsinformationen ein Hinweis auf eine tatsächliche Abwesenheit der entsprechenden Verbindungen ist. Dementsprechend wurden die nach der Auswertung der Daten aus CoCoMac-Datenbank noch leeren Zellen der Verbindungsmatrix mit Nullen gefüllt (Verbindungsmatrix W40-0, siehe dazu im Anhang A.3). Die Shapleywerte für die Knoten des durch diese Adjazenzmatrix beschriebenen Graphen sind in Tabelle 4.2 zu sehen. Auch für diesen Graph gilt, daß er starkzusammenhängend ist, und somit ist der ist der

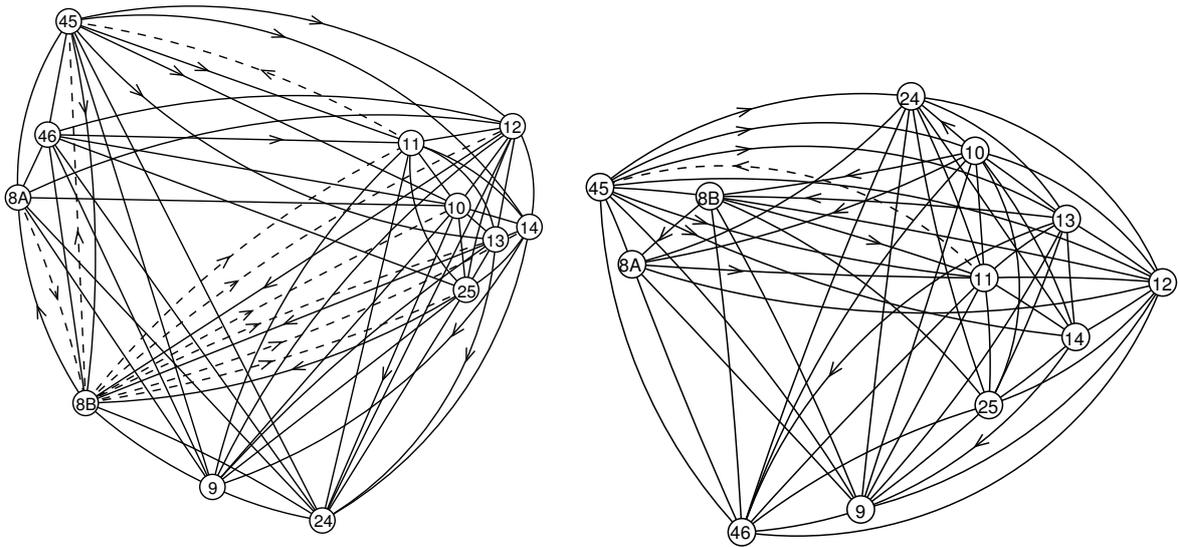


Abbildung 4.11: Gerichtete Graphen, die das Netzwerk des präfrontalen Cortex repräsentieren. Links das W40-Netzwerk, das schon in früheren Studien verwendet wurde, rechts das neue W40-Netzwerk. Die gestrichelten Linien stellen in beiden Fällen Verbindungen dar, bei denen unbekannt ist, ob sie als neuronale Verbindung tatsächlich existieren.

durchschnittliche Shapleywert der Knoten $\frac{1}{12}$.

Für diese erste Version des W40-Netzwerks sind die Knoten mit den deutlich niedrigsten Shapleywerten die Areale 9 und 24. Diese beiden Areale haben jeweils 21 von 22 bei zwölf Arealen möglichen Verbindungen und sind somit beide auch mit allen anderen Arealen des präfrontalen Cortex verbunden, und sogar jeweils bis auf ein Areal sowohl afferent als auch efferent. Ein Vergleich mit den Ergebnissen von Passingham et al. zeigt, daß die beide Areale mit den niedrigsten Shapleywerten aus deren letzter Gruppe kommen. Area 24 (anteriorer zingulärer Cortex) ist mit vielen verschiedenen Gehirnsystemen verbunden und dafür bekannt, eine zentrale Bedeutung in Motivationsprozessen zu haben. Ähnlich bildet die sich dorsal befindende Area 9 ein Brücke zwischen präfrontalen Arealen und dem prämotorischen Cortex.

Wenn wir noch die beiden Areale mit dem nächstniedrigsten Shapleywert in die Betrachtung einbeziehen (Area 12 und 46), dann sind nicht nur alle drei Arealgruppen von Passingham et al. unter den niedrigsten Shapleywerten vertreten, sondern wir können auch feststellen, daß auf funktionaler Ebene mindestens zwei dieser Areale (Area 24 und 46) bekannterweise unter den am häufigsten aktivierten Regionen unter einer Reihe von Versuchsparadigmen bei bildgebenen Verfahren vertreten sind. Außerdem gelten sie als überaktiviert bei Attention Deficit Hyperactivity Disorder (siehe zum Beispiel [DBCM06]).

Auf der anderen Seite sind die drei Areale mit den höchsten Shapleywerten (8B, 45, 14) klar mit sehr speziellen Gehirnfunktionen assoziiert (Raum- und Objektwahrnehmung, autonome Kontrolle). Jedes dieser Areale gehört zu einer anderen der drei Arealgruppen, die von Passingham et al. definiert wurden, so daß jeder Cluster Areale mit sehr hohen und sehr niedrigen Shapleywerten enthält. Das nach den Shapleywerten am wenigsten wichtige Areal ist die Area 8A (Frontales Augenfeld), das sehr spezifische Verbindungen und Funktionen hat. Dagegen hatte dieses Areal in dem Netzwerk des visuellen Systems einen sehr niedrigen Shapleywert.

Dies paßt zu der Idee, daß seine Rolle in einem speziellen Zusammenhang wichtig ist, aber nicht wesentlich für das gesamte präfrontale System.

Insgesamt haben die Hälfte der Areale einen niedrigeren Shapleywert als $\frac{1}{12}$.

$\phi_{f_{W40-0}}(v)$		$\phi_{f_{W40-1}}(v)$		$\phi_{f_{W40-2}}(v)$	
9	-0.045093	9	0.031709	12	-0.004113
24	-0.031204	12	0.041630	9	0.013745
12	0.037049	24	0.042821	46	0.035570
46	0.044191	25	0.063455	25	0.062951
25	0.073520	8B	0.075360	24	0.075216
8A	0.073917	11	0.079329	11	0.078824
10	0.084668	10	0.096392	10	0.092280
11	0.112842	8A	0.102741	13	0.102633
13	0.127128	13	0.106349	8B	0.130736
14	0.131890	46	0.107900	14	0.135101
45	0.178282	45	0.120995	45	0.137157
8B	0.212806	14	0.131313	8A	0.139899

Tabelle 4.2: Shapleywerte für die drei Versionen der W40-Verbindungsmatrix.

Wie schon erwähnt, gibt es keine vollständigen Informationen über alle Kanten des Verbindungsgraphen, so daß einige Felder in der Verbindungsmatrix für Berechnungen ergänzt werden müssen. Kanten, über die keine Informationen vorhanden sind, wurden in der ersten Shapleywertberechnung als abwesend eingestuft. Dies ist eine verbreitete Vorgehensweise, die durch die Annahme gestützt ist, daß alle wirklich vorhandenen Verbindungen wahrscheinlich dokumentiert sind. Allerdings sind einige der efferenten Verbindungen von Areal 8B aufgrund einer Neudefinition von Areal 8B in späteren Tracing-Studien nicht verfügbar (siehe z.B. [KS03]), und diese machen einen Großteil (sieben von neun) der unbekanntenen Verbindungen aus. Deshalb wurden alle Shapleywerte noch einmal unter der Annahme berechnet, daß alle Verbindungen, über die keine Informationen vorliegen tatsächlich existieren (Verbindungsma-trix W40-1, siehe Anhang A.4).

Wie aufgrund der relativ kleinen Änderungen zu erwarten, ergibt diese Berechnung mit der alternativen Verbindungsma-trix für die meisten Areale ähnliche Bewertungen wie die erste Berechnung. Einige Unterschiede fallen dennoch auf: Zum einen hat kein Areal eine negative Bewertung, das heißt, daß die absolute Abhängigkeit von ein paar Arealen abnimmt. Da auch die hohen Shapleywerte niedriger sind als für die ersten Verbindungsma-trix W40-0, ist die Spanne der Shapleywerte in diesem Netzwerk also insgesamt geringer. Da der Graph auch schon vorher dicht vernetzt war, und die zehn neu eingefügten Kanten mit zehn verschiedenen Knoten verbunden sind, war dies auch zu erwarten - schließlich sind in einem vollständigen Graphen alle Knotenbewertungen gleich.

Allerdings hat das Einfügen der Kanten die anderen Knoten in unterschiedlichen Maße beeinflusst: Der Shapleywert von Areal 8B, das insgesamt sieben neue efferente und eine neue afferente Verbindung bekam, sank deutlich, ebenso der von Areal 45, welches zwei zusätzliche

afferente Verbindungen erhielt. Diese beiden Areale hatten vorher mit Abstand die höchsten Shapleywerte. Auf der anderen Seite stiegen die Shapleywerte der Areale, zu denen Areal 8B auch schon vor dem Einfügen der neuen Kanten efferente Verbindungen hatte (Areale 24, 46, 8A und 9), darunter also auch die von Areal 9 und Areal 24, die vorher die einzigen Knoten waren, die eine negative Bewertung hatten. Bei allen anderen Arealen zeigten sich aber nur minimale Auswirkungen durch die Modifikation.

Eine dritte Variante der präfrontalen Matrix wurde aus einer neueren Auswertung von Tracing-Studien gewonnen. Dazu wurde die Informationsdatenbasis aktualisiert und erweitert sowie die Kartierungsprozedur verbessert, um weitere Optionen zu berücksichtigen (Verbindungs-matrix W40-2, siehe Anhang A.5). Insgesamt hat diese Matrix nur einen undefinierten Eintrag: die Projektion von Areal 11 nach Areal 45, von der hier angenommen wird, daß sie nicht existiert.

Das Anordnen der Shapleywerte nach Größe führt zu einer sehr ähnlichen Folge von Beiträgen zur Zusammenhangsstruktur des Gesamtnetzwerkes wie bei den anderen beiden Verbindungsmatrizen, deswegen sind hier nur die deutlichsten Veränderungen aufgelistet. Während die Areale 12, 9, und 46 weiterhin die niedrigsten Bewertungen haben, nahm die Bedeutung von Areal 24 weiter ab. Im Gegensatz dazu nahm die Bedeutung von Areal 25 (in Walkers Karte der rostro-dorsale Teil des anterioren Gyrus cinguli, ähnlich Brodmanns Area 32) zu. Den größten Shapleywert hat jetzt Areal 8A, gefolgt von den Arealen 45, 14 und 8B.

4.6.3 Diskussion

Die Anwendung der Shapleywerttheorie auf neuronale Netzwerkmodelle ist ein neuer Ansatz der Verbindungsanalyse. Indem Shapleywerte den Knoten dieser Netzwerke zugeordnet werden, wird versucht, den Beitrag, den jeder einzelne Knoten zum Gesamtzusammenhang des Netzwerkes macht, adäquat zu quantifizieren.

Für Gehirnnetzwerke wurde schon das Maß der Kantenverwundbarkeit („edge vulnerability“) von Kaiser und Hilgetag definiert [KH04]. Dies ermöglichte die Identifikation von Kanten, deren Verlust den größten Effekt auf globale Strukturmaße hat. Diese Kanten sind oft „Brückenkanten“, die getrennte Kluster von Gehirnregion verbinden. Eine detailliertere Studie von Läsionseffekten in einem größeren Gehirnnetzwerk von Kaiser et al. [KMAY07] zeigte, daß sich dieses Netzwerk, nachdem Knoten entfernt wurden, in Bezug auf die durchschnittliche kürzeste Weglänge ähnlich wie ein vergleichbares „scale-free“ Netzwerk verhielt.

Hier wurde im Gegensatz dazu ein Maß betrachtet, in das Eigenschaften des gesamten Netzwerkes direkt eingehen. Aufgrund der Konstruktion der Knotenbewertung ist anzunehmen, daß das Entfernen eines Knotens mit einem niedrigen Shapleywert, der also einen großen Beitrag zum Zusammenhang leistet, den Netzwerkzusammenhang auf wichtige Weise beeinflussen würde, während das Entfernen eines Knotens mit hohem Shapleywert einen geringeren Einfluß haben sollte. Deshalb sollte die Bewertung potentiell nützlich sein, um diskrete Netzwerk-läsionen zu analysieren und vorherzusagen, in welchem Umfang eine spezielle Läsion den Zusammenhang des Netzwerkes unterbräche.

Shapleywerte könnten also ein neues Analysewerkzeug sein, um die funktionalen Auswirkungen eines Ausfalls von Netzwerkknoten vorherzusagen. Zum Beispiel deutet die Analyse des visuellen Systems nach Young darauf hin, daß eine Läsion von Areal V4 starke Auswirkungen

auf das visuelle Netzwerk haben könnte, was mit der Beobachtung von Felleman und Van Essen, daß V4 innerhalb des visuellen kortikalen Netzwerk eine extrem dichte Vernetzung aufweist, im Einklang steht. Eine Analyse einer ähnlichen Verbindungsmatrix, bestehend aus visuellen, somatosensorischen und motorischen kortikalen Arealen, zeigte, daß die Entfernung des Areals V4 die „small-world“ Eigenschaften des verbleibenden Netzwerkes stark vermindert [SHK07]. Als weitere mutmaßliche Hub-Regionen wurden in dieser Arbeit die Areale FEF, 46 und 7a identifiziert, die auch alle einen negativen Shapleywert innerhalb des visuellen Netzwerkes hatten. Somit deuten auch die Shapleywerte auf eine möglicherweise sehr signifikante Rolle der präfrontalen Areale 46 und FEF innerhalb des visuellen Systems hin, was bemerkenswert ist, da diese Areale kein grundlegender Bestandteil des visuellen kortikalen Systems sind. Es sollte untersucht werden, ob ihre Verbindungen mit dem posterioren Cortex mit anderen frontalen Areale vergleichbar sind. Im Gegensatz dazu legt die Analyse der Shapleywerte nahe, daß eine Läsion von inferior temporalen Arealen oder sogar von V1 zu nur sehr begrenzten Störungen innerhalb des cerebralen Kortex führen würde.

Die relative Beständigkeit der Arealbewertungen im präfrontalen Cortex bei allen drei Verbindungsmatrizen spricht für die Robustheit der Analysemethode, da sie, trotz unterschiedlicher Methoden, die sonst stark beschränkten Verbindungsmatrizen zu füllen, vergleichbare Ergebnisse liefert. Außerdem scheint die Liste der Areale mit extremen Bewertungen die Annahme zu bestätigen, daß ein niedriger Shapleywert mit einer breiten Funktionalität und häufiger Aktivierung des Areals in Zusammenhang steht, wohingegen ein hoher Shapleywert für eine spezialisierte Funktionalität und eine eher aufgabenabhängige Aktivierung spricht. Dabei wird auch Area 8A (FEF) im präfrontalen Kortex als ein (auf Augenbewegungen) spezialisiertes Areal angesehen. Die Bewertungen wecken zudem weiteres Interesse an den präfrontalen Arealen 12 und 9, deren Rollen schlecht verstanden sind, die aber anscheinend eine kritische Position an der Schnittstelle zwischen den Gruppen der präfrontalen Arealen einnehmen.

Die Ergebnisse legen also die Annahme nahe, daß eine Läsion eines Areals mit niedrigem Shapleywert große Auswirkungen auf die Funktion des Netzwerkes haben sollte. Außerdem kann die Berechnung von Shapleywerten in einem Netzwerk, das durch eine Läsion eingeschränkt ist, Anhaltspunkte geben, wie diese Läsion das restliche Netzwerk beeinflusst. Dies kann man zum Beispiel dadurch erreichen, daß man Shapleywerte während des schrittweisen Entfernens von Knoten aus dem Netzwerk berechnet. So könnte man eine Desintegration des Netzwerkes als zum Beispiel „kleine Degeneration“ im Vergleich zu „katastrophale Beeinträchtigung“ charakterisieren.

Es gibt allerdings einige praktische Einschränkungen, die der Berechnung von Shapleywerten anhaften. Wenn wir ein Netzwerk bestehend aus n Knoten haben, dann müssen wir, um die Bewertung eines Knotens nach der Formel 4.1 oder 4.2 zu erhalten, alle $n!$ Permutationen der Menge von n Knoten betrachten. Also ist die Anzahl der Rechenschritte zur Berechnung eines Shapleywertes im Allgemeinen proportional zu $n!$. Ein anderes Problem ist, daß die Zahlen, die in der Berechnung vorkommen, schnell zu groß für eine präzise Integerarithmetik werden. Deshalb war es sogar schon für die relativ kleinen Netzwerke von bis zu 30 Knoten, die hier verwendet wurden, notwendig, spezielle Datentypen zu implementieren, die lange Integer bewältigen können.

Es sollte auch nicht außer Acht gelassen werden, daß alle hier geführten Überlegungen die strukturelle Integrität eines Netzwerkes ansprechen. Dabei bleibt die Signifikanz dieser Maße innerhalb eines funktionalen Netzwerkes, insbesondere im Zusammenhang mit solchen Phäno-

menen wie Stimulus geleitete oder Spontanaktivität (und wie diese von Läsionen betroffen sind), noch ungeklärt (siehe Sporns et al. [SCKH04] für einige Gedanken zu dem Thema Gehirnnetzwerke und Kognition).

Während Shapleywerte schon auf funktioneller Ebene eingeführt wurden (siehe zum Beispiel [KSH⁺04]), ist nicht klar, in welchem Verhältnis sie zu strukturellen Shapleywerten stehen. Um die Struktur-Funktionsbeziehung herauszuarbeiten, bräuchte man eine große Datenbank von korrelierten Funktions- und Strukturdaten für eine Tierart oder noch besser für denselben Organismus. Dann werden Berechnungsmodelle gebraucht, die überprüfbare Vorhersagen von Zusammenhängen zwischen Struktur und Funktion bereit stellen können.

Kapitel 5

Hierarchien in funktionellen Systemen

5.1 Einführung

Um leichter aufklären zu können, wie die strukturelle und funktionelle Organisation des Gehirns die Informationsverarbeitung bewerkstelligt, wird der Cortex in funktionelle Systeme gegliedert (vergleiche Abschnitt 2.2). Innerhalb dieser Systeme nimmt man eine inhärente hierarchische Struktur des Informationsflusses an. Dabei geht man von einem funktionalen hierarchischen Modell aus, bei dem die in kortikalen Inputs enthaltene Information in niederen Arealen zunächst nach einfachen Merkmalen durchsucht werden, deren spezifische Kombination in höheren Arealen als Merkmale komplexerer Strukturen erkannt werden. So gelangt man von separaten Stücken sensorischer Information über eine zunehmend integrierten Repräsentation dieser Stücke zu einer semantischen Klassifikation. Zum Beispiel soll im visuellen System die Erkennung von Kanten verschiedener Orientierungen auf der nächsten Verarbeitungsstufe zur Erkennung von Ecken genutzt werden, deren Kombination wiederum zur Unterscheidung verschiedener Objektklassen führt, bis ein Objekt nach mehreren Verarbeitungsstufen identifiziert wird. Die Verarbeitung der sensorischen Signale erfolgt also in mehreren aufeinander aufbauenden Stufen.

Die Literatur, die sich mit hierarchischen Organisation der Informationsverarbeitung im Gehirn beschäftigt, ist sehr umfangreich. Ziel dieser Studien ist es, die hierarchische Ordnung der funktionellen Systeme zu finden, und dadurch die Informationsverarbeitung besser zu verstehen. Am besten ist dabei bisher das visuelle System untersucht. Einen Überblick über die Entwicklung der Forschung in diesem Bereich erhält man zum Beispiel in dem Übersichtsartikel von Vezoli et al [VFJ⁺04].

Die visuelle Verarbeitung in der Hirnrinde beginnt mit dem primären visuellen Cortex und läuft von dort weiter „nach oben“ zu zunehmend spezialisierten und assoziativen Kortikalregionen. Verbindungen die in Richtung dieses Informationsflusses verlaufen, werden auch als „Feedforward“-Projektionen beschrieben. Hingegen werden Verbindungen die entgegen dieser Laufrichtung „nach unten“ gehen, in der Regel als „Feedback“-Projektionen betrachtet. Wenn man den Verarbeitungsweg sensorischer Information durchläuft, trifft man auf zunehmende Ambiguität in Bezug auf die Zuordnung einer funktionellen Hierarchie zu den neuronalen Strukturen.

5.1.1 Die Arbeit von Felleman und Van Essen

Um diese Ambiguität aufzuklären, haben Felleman und Van Essen [FE91] ein einfaches anatomisches Kriterium verwendet, um Informationsverarbeitung in hierarchische Stufen zu gliedern. Sie nutzten dazu Verbindungsinformation aus Tracing-Studien im Makaken in Verbindung mit der Idee, daß verschiedene laminare Muster an Ausgangspunkt und Ende einer Verbindung zwischen aufsteigenden, absteigenden und lateralen Verbindungen unterscheiden kann (siehe Abschnitt 2.3 zu laminaren Mustern bei Tracing-Studien). Felleman und Van Essen charakterisieren die Verbindungen also basierend auf den Schichten, von denen sie im Startareal beginnt, und den Schichten, in denen sie im Zielareal endet, als „aufsteigend“, „lateral“ und „absteigend“, je nachdem ob sich das Ausgangsareal auf einer niedrigeren, derselben oder höheren Hierarchiestufe als das Zielareal befinden soll, siehe Abbildung 5.1. Damit haben Verbindungen nicht nur eine Richtung vom Start- zum Zielareal, sondern auch ein Richtung innerhalb der Hierarchie.

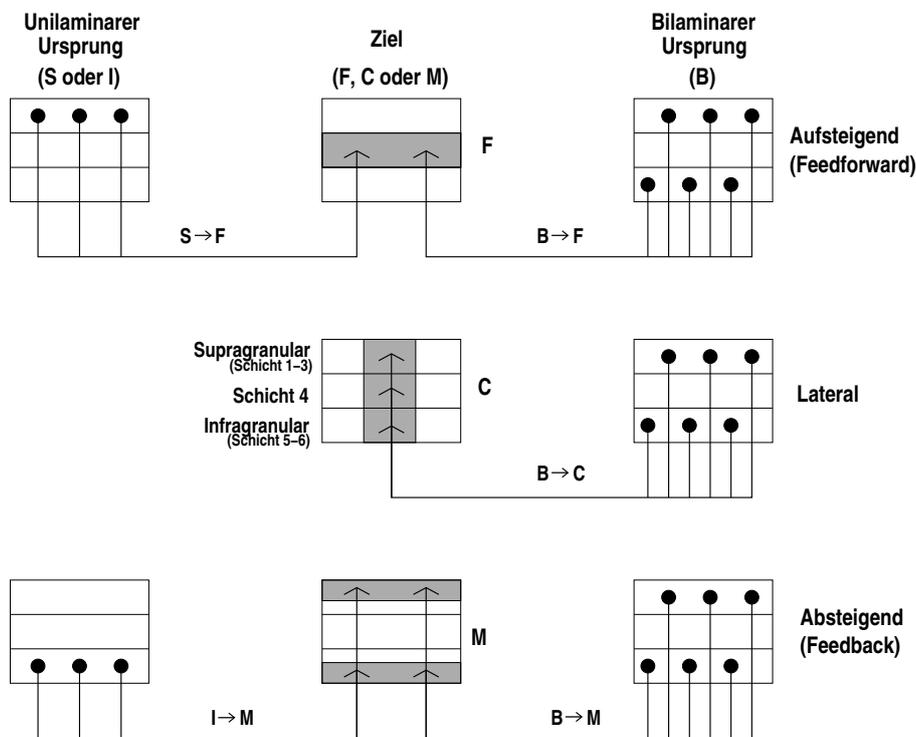


Abbildung 5.1: Klassifikation der Verbindungen nach Felleman und Van Essen [FE91]. Eine Verbindung ist demnach aufsteigend, wenn sie nur supragranularen (S) oder supra- und infragranularen (bilaminaren, B) Ursprung hat und in Schicht 4 (F) endet. Ist der Ursprung bilaminar oder infragranular (I) und endet sie multilaminar (M) aber nicht in Schicht 4, so handelt es sich um eine absteigende Verbindung. Bei supra- und infragranularen Ursprung und Endungen auf allen Schichten (kolumnar, C) wird eine Verbindung als lateral angesehen. Die Kriterien Supragranular und Infragranular wurden als erfüllt angesehen, wenn mindestens 70% der gefärbten Neuronen in diesem Bereich lagen. Wenn sich die gefärbten Neuronen einigermaßen gleichmäßig (bis zu 30%-70%) auf die infragranularen und supragranularen Schichten verteilen, so handelt es sich um ein bilaminares Muster.

Grundlage dieser Einstufung sind die Verbindungen der kortikalen Areale am unteren Ende der Verarbeitungshierarchie, über deren Beziehung zueinander große Einigkeit herrscht. Es wird also angenommen, daß Areale mit denselben laminaren Mustern auch auf dieselbe Art in der Hierarchie verbunden sind.

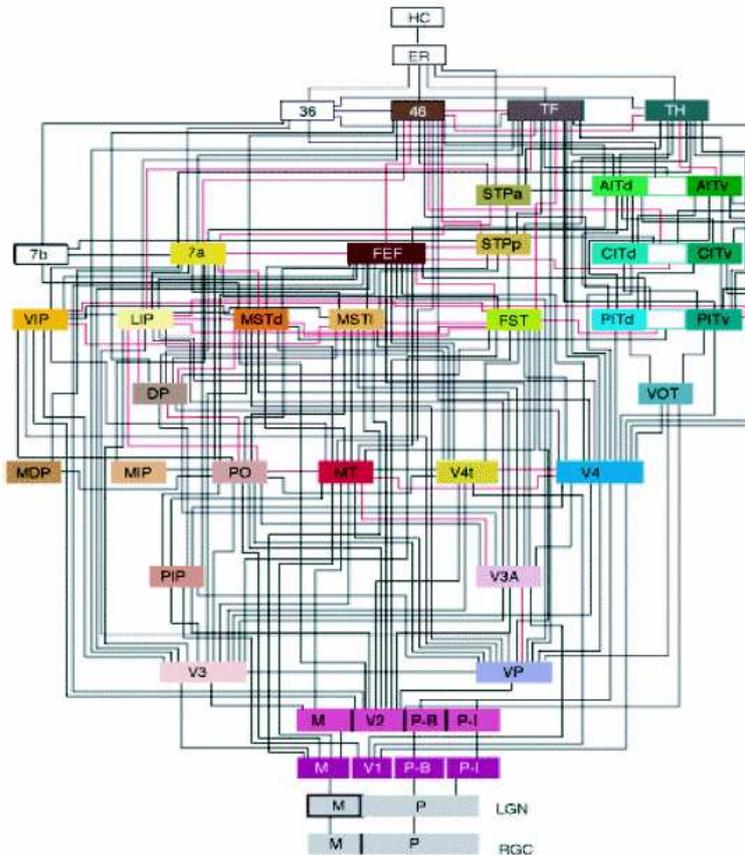


Abbildung 5.2: Hierarchie des visuellen System nach Felleman und Van Essen, entnommen aus [FE91].

In ihrer wegbereitenden Arbeit machten Felleman und Van Essen eine grundlegende hierarchische Analyse der der kortikalen Verbindungen und schufen eine erste hierarchische Karte des gesamten visuellen System [FE91]. Dazu haben sie aus bisherigen Veröffentlichungen Verbindungsdaten gesammelt und systematisiert. Sie klassifizierten Verbindungen zwischen Paaren von Arealen als „feedforward“ und „feedback“ basierend auf deren laminaren Eigenschaften. Die Autoren erstellten ein hierarchisches Modell des visuellen Systems mit 32 kortikalen Arealen und beschrieben 318 der 992 paarweise hierarchischen Beziehungen zwischen den Arealen. Mit Hilfe dieser Daten waren sie in der Lage ein hierarchisches Modell vom System zu entwerfen. Ausgehend vom Areal V1, das allgemein als Startareal für die visuelle Verarbeitung angesehen wird, bauten sie ihre Hierarchie von unten nach oben auf, so daß die Verbindungen so weit wie möglich der vorher festgelegten Klassifizierung entsprachen. Das Ergebnis ist in Abbildung 5.2 zu sehen. Dabei war es nicht in an allen Stellen möglich, die Areale entsprechend ihrer Verbindungen einzuordnen, da die Daten nicht widerspruchsfrei sind. Insgesamt

weist die von Felleman und Van Essen erstellte Hierarchie 3 Verstöße gegen die von ihnen mit den gesammelten Daten erstellten Kriterien auf. Dabei ist auch anzumerken, das die Autoren kein Optimierungsverfahren verwendeten, sondern die Areale so anordneten, wie es ihnen aufgrund ihrer Erfahrung am sinnvollsten erschien.

Dabei haben sie auch eine Verbindungsmatrix für das visuelle System geschaffen. Die von Felleman und Van Essen dazu gesammelten Daten werden weithin akzeptiert und auch heute noch als Grundlage für Untersuchungen und Vergleiche genommen, so auch in dieser Arbeit.

5.1.2 Kritik an der Arbeit von Felleman und Van Essen

Eine Kritik am Modell von Felleman und Van Essen ist, daß die Hierarchie relativ willkürlich ausgewählt wurde. Hilgetag et al. versuchten das Verfahren zu objektivieren, indem sie eine Kostenfunktion erstellten, die die Verstöße einer Hierarchie gegen die Klassifizierung der Verbindungen aufgrund der experimentell ermittelten anatomischen Gegebenheiten maß [HOY96]. Dann berechneten sie mit einem Optimierungsverfahren mögliche Hierarchien, die sich aus den Daten von Felleman und Van Essen ableiten ließen.¹ Sie fanden auf diese Art rund optimale 150000 Hierarchien, die jeweils sechs der anatomischen Bedingungen verletzen. Aufgrund von Beschränkungen in der Rechnerkapazität brachen sie dann die Berechnungen ab, so daß diese Zahl als untere Grenze für die tatsächliche Anzahl gewertet werden kann. Außerdem bleibt offen, ob es eine Hierarchie mit weniger Verletzungen der Bedingungen gibt. Das Modell von Felleman und Van Essen kommt nach der Kostenfunktion von Hilgetag et al. auf acht solcher Verletzungen.

Außerdem berichten die Autoren, daß selbst wenn das Modell von einer vollständigen Menge an Bedingungen eingeschränkt wird (wenn man zum Beispiel alle unbekanntes Verbindungen als vorhanden voraussetzt), die Anzahl der gleichermaßen optimalen Modelle weiterhin groß bleibt. Interessanterweise werden die primären kortikalen Areale (V1 und V2) in allen optimalen Modellen unten in der Hierarchie eingestuft, während die Einstufung vieler höherer Areale stark variiert. Eine Übersicht über die Ergebnisse von Hilgetag et al. findet sich in Abbildung 5.3.

Die Anzahl der Stufen der optimalen Hierarchien reicht von 13 (für 21 Lösungen) bis 24 (für 3 Lösungen) mit der höchsten Anzahl der Lösungen bei 18 und 19 Stufen. Felleman und Van Essen hingegen kamen mit zehn Hierarchiestufen aus. Sie haben aber auch ganz bewußt ein Hierarchie mit möglichst wenig Stufen gesucht, indem sie bei Verbindungen, die nicht eindeutig klassifiziert wurden, vorzugsweise eine laterale Verbindung annahmen. Die bei Hilgetag et al. auftretende Variabilität der Position der Areale ist nach Ansicht der Autoren darauf zurückzuführen, daß das Verhältnis vieler Areale zueinander nicht klar bestimmt werden kann.

Durch eine Verbindung zwischen zwei Arealen wird eine Ordnung zwischen diesen festgelegt. Man kann mit Hilfe der Verbindung also feststellen, ob sich eines auf einer höheren Hierarchiestufe befindet als das andere, oder ob beide auf derselben sind. Für die 32 Areale sind aber nur 318 der 992 möglichen Beziehungen definiert. Das führt sowohl zu unterschiedlich Positionen der Areale in Relation zueinander in verschiedenen optimalen Lösungen, als auch zu Lösungen mit einer unterschiedlichen Anzahl von Stufen. Hilgetag et al. argumentierten,

¹Hilgetag et al. schlossen die Areale MDP und MIP von der Analyse aus, da es keine Daten gab, die die hierarchische Beziehung dieser Areale zu anderen Arealen einschränkt. Dadurch können diese Areale beliebig plziert werden, ohne daß sie verletzte Bedingungen verursachen.

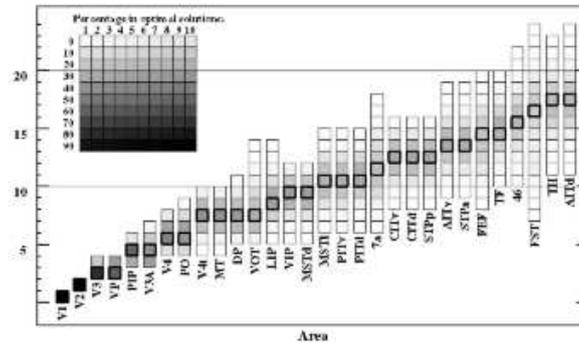


Abbildung 5.3: Ergebnisse von Hilgetag et al., entnommen aus [HOY96]. Die Schattierung der Kästchen drückt die relative Häufigkeit aus, mit der ein Areal auf diesem Level platziert wird, gemittelt über alle 150000 optimalen Hierarchien. Je dunkler die Schattierung desto häufiger ist eine Plazierung auf dieser Stufe. Die häufigste Plazierung für jedes Areal wird durch die dickeren Linien angezeigt. Die Ordnung dieser Schwerpunkte bildet auch eine optimale Hierarchie.

daß man aufgrund der vielen nachweisbar nicht vorhandenen Verbindungen keine eindeutige Hierarchie finden kann. Man sucht also eine totale Ordnung der Areale, hat aber nicht ausreichend viele Informationen um diese zu finden.

Das führt zu der Frage, ob eine Hierarchie durch die verwendete Methode (eindeutig) bestimmt werden kann. Ein Problem dabei ist, daß der Abstand der Levels und der Areale in Bezug auf die Hierarchie unklar ist. Insbesondere bei parallelen Wegen ist unklar, ob die Areale in den parallelen Wegen jeweils auf derselben oder auf verschiedenen Levels angeordnet werden sollen (siehe Abbildung 5.4 für ein Beispiel). Offensichtlich führen parallele Wege in einer Hierarchie ohne ein Abstandsmaß zwischen den Stationen zu verschiedenen möglichen Modellen.

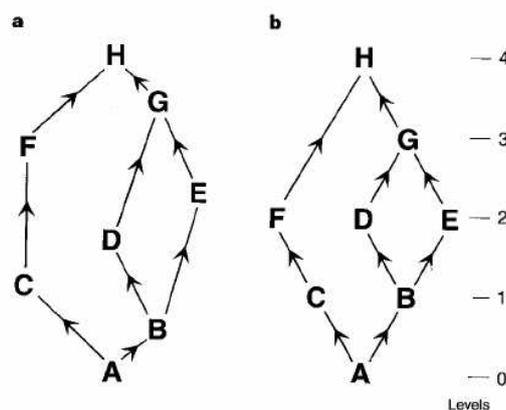


Abbildung 5.4: Zwei von vielen Möglichkeiten, Areale auf Hierarchiestufen zu verteilen, wenn nur bekannt ist, daß alle eingezeichneten Verbindungen aufsteigend sind. (Abbildung aus Crick und Koch [CK98])

5.1.3 Ein mögliches Abstandsmaß für Verbindungen

Während diese Ergebnisse den Eindruck einer dem System innewohnenden Unbestimmtheit in Bezug auf die hierarchische Ordnung bestärken, deuten weitere Untersuchungen der laminaren Verbindungsmuster, (die ja zur Bestimmung der Richtung von hierarchischen Projektionen verwendet werden), durch die Arbeitsgruppe um Kennedy darauf hin, daß es möglich ist, eine genauere Klassifizierung der Verbindung vorzunehmen [KB85, BBKK00, BBK⁺02]. Diese Autoren weisen den Verbindungen nicht nur eine Richtung, sondern auch einen „Abstand“ in Bezug auf die Hierarchie zu. Dazu verwenden sie den Anteil von retrograd gekennzeichneten Neuronen in den Supragranularen Schichten, den sie mit SLN% bezeichnen. Dieser korreliert nach ihren Untersuchungen stark mit den hierarchischen Rangunterschieden zwischen Start- und Zielregion einer Verbindung (in dem Modell von Felleman und Van Essen). Eine Übersicht zu diesem Ansatz und den verwendeten Methoden findet man bei [VFJ⁺04].

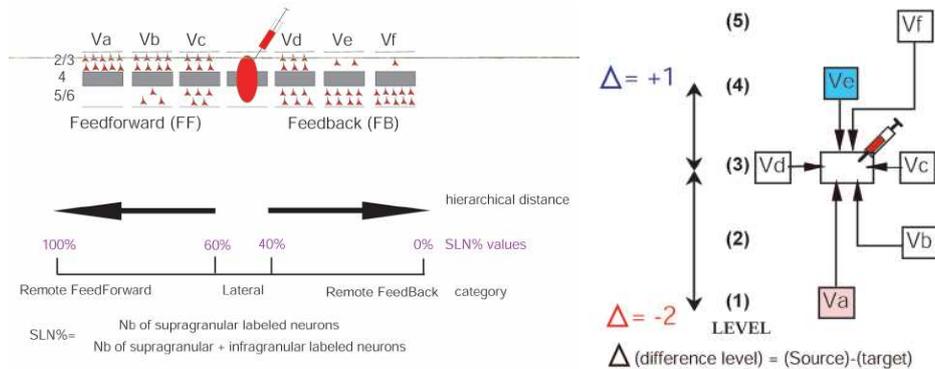


Abbildung 5.5: Klassifizierung von Verbindungen mit Hilfe der supra- und infragranular gefärbten Neuronen. Links die Verteilung der Tracing-Substanz auf die supra- und infragranularen Neuronen. Rechts die mögliche Umsetzung dieser Daten in ein Hierarchie. Abbildung entnommen aus [VFJ⁺04]

In zwei Tracing Studien wurde untersucht, ob sich die Art der Verbindung schon durch die laminare Ausprägung im Startareal zeigen läßt [BBKK00, BBK⁺02, VFJ⁺04]. Injiziert wurde dabei in die Areale V1 und V4, die sich am unteren Ende der Hierarchie befinden. In diesem Bereich herrscht weitgehende Einigkeit über die Anordnung der Areale auf den Hierarchiestufen.

Nach einer Injektion in V4 mit einem retrograden Tracer zeigten Areale, von denen man annimmt, daß sie sich auf einer niedrigeren Hierarchiestufe als V4 befinden, mehr eingefärbte Neuronen in supragranularen als in infragranularen Schichten. Bei V1 war der Anteil der gefärbten Neuronen, die sich in supragranularen Schichten befanden, sogar 100%. Umgekehrt wurden zunehmend mehr Neuronen in infragranularen Schichten markiert, je weiter sich das Areal in der Hierarchie oberhalb von V4 befand [BBKK00]. Die Autoren führen deshalb das Maß SLN% ein. Es gibt an, wieviel Prozent des Tracers in supragranularen Schichten zu finden ist. Die Autoren nehmen aufgrund der beschriebenen Verteilungen an, daß SLN% ein Maß für die hierarchische Distanz zwischen zwei Arealen ist. Mit Hilfe des SLN%-Wertes wurde ein diskretes Abstandsmaß für Areale innerhalb der Hierarchie definiert. So wurden zum Beispiel Verbindungen, bei denen das Startareal einen SLN%-Wert von 40%-60% aufwies, als lateral

klassifiziert. An der erstellten Einteilung soll man ablesen können, wie viele Hierarchiestufen zwischen zwei Arealen liegen. (Vergleiche die Abbildung 5.5 für eine graphische Darstellung.) Mit Hilfe dieser neuen Klassifikationen wurde ein Teil der Hierarchie von Felleman und Van Essen überarbeitet (siehe Abbildung 5.6).

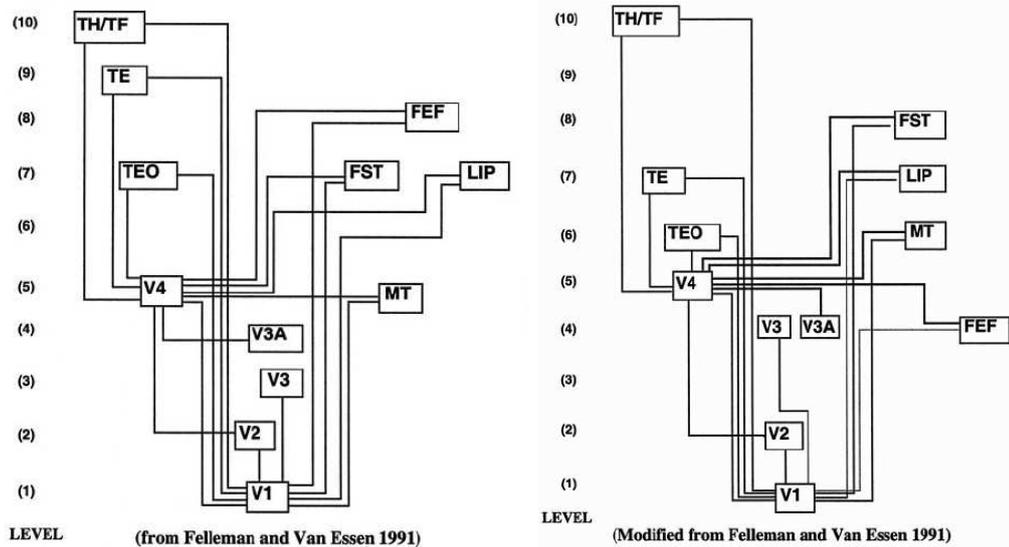


Abbildung 5.6: Hierarchie nach Felleman und Van Essen (links) sowie Barone et al. (rechts), Abbildung entnommen aus [BBKK00]

Diese Klassifizierung der Verbindungen über die SLN%-Werte hat Parallelen zu der von Felleman und Van Essen, basiert aber nur auf der laminaren Verteilung der Verbindung im Startareal, während die von Felleman und Van Essen zusätzlich auch die Schichten des Ziellareals verwendet. Trotzdem scheint sie mehr Information zu enthalten, denn die Autoren schließen, daß der SLN%-Wert als Maß für die hierarchische Distanz verwendet werden kann.

5.1.4 Ein neues Modell zur Bestimmung von Hierarchien

Durch ein Abstandsmaß ergibt sich eine Bewertung der Verbindungen, die verwendet werden kann, um ein optimales hierarchisches Modell zu bestimmen. Wenn man den Abstand der Areale bezüglich ihrer Stellung in der Hierarchie zur Verfügung hat, kann man auch bei parallelen Verarbeitungswegen eindeutig angeben, auf welcher Hierarchiestufe die Areale anzuordnen sind. Es ist auch nicht mehr von Bedeutung, wie viele der möglichen Verbindungen vorhanden sind, solange das Netzwerk zusammenhängend ist. Man kann für jedes Areal, das von V1 aus erreicht werden kann, ausrechnen, wie viele Hierarchiestufen es von V1 entfernt liegt. Dafür ist es allerdings Voraussetzung, daß sich die Abstände transitiv verhalten. Dies ist aber im Rahmen einer Hierarchie sinnvoll. Theoretisch legt eine Hierarchie eine totale Ordnung der Areale fest. Man kann also für je zwei Areale immer sagen, wie sich ihre Hierarchiestufen zueinander verhalten, also ob eine größer als die andere ist, oder ob sie beide gleich sind.

Eine weitere Einschränkung der bisher vorgestellten Modelle ist die Definition der Hierarchiestufen auf einer diskreten Skala. Dieser Ansatz trägt die implizite Annahme, daß der Abstand

zwischen den Hierarchiestufen auf allen Hierarchieebenen gleich ist. Dagegen läßt die große Varianz, die bei Hilgetag et al. beim Plazieren der Areale auf den höheren Hierarchiestufen aufgetreten ist, darauf schließen, daß sich die Abstände auf den höheren Hierarchieebenen anders als auf den unteren verhalten. Die Ausprägung der laminaren Muster von Verbindungen weist auch keinen sprunghaften sondern einen kontinuierlichen Übergang zwischen zwei Mustern auf, so daß jeder Grenzwert für eine Verbindungsklassifikation willkürlich sein muß. Und da die Ausprägung der laminaren Muster von Verbindungen, wie man am SLN%-Wert sieht, auf einer kontinuierlichen Skala liegen, ist es nur konsequent, auch die Hierarchie auf einer kontinuierlichen Skala anzusiedeln.

Hier wird ein neuer Ansatz zur Bestimmung der hierarchischen Organisation von kortikalen Netzwerken vorgestellt, der sowohl ein Distanzmaß als auch Hierarchiestufen auf einer kontinuierlichen Skala verwendet. Anschaulich sollen die Knoten dabei reellwertige Levels zugeordnet werden, so daß eine Hierarchie entsteht, die das Abstandsmaß respektiert.

5.2 Reellwertige Hierarchien

Wir betrachten also in diesem Kapitel gerichtete Graphen $G = (V, E)$ mit einer Kostenfunktion $f : E \rightarrow \mathbb{R}$, die jeder Kante einen reellen Kostenwert, oder in unserem Fall eine Distanz zuordnet. Ein solcher Graph G mit Kostenfunktion f heißt *kantenbewerteter Graph*. Die *Länge* eines Weges $p = (v_1, \dots, v_k)$ in einem kantenbewerteten Graphen ist die Summe der Kostenwerte (Längen) aller Kanten des Weges,

$$f(p) = \sum_{i=1}^{k-1} f((v_i, v_{i+1})).$$

Man möchte nun eine Ordnung auf Knoten in Graph G finden, die die Richtung und die Bewertung der Kanten berücksichtigt. Für eine Kante (u, v) mit reeller Kantenbewertung bedeutet dann $f((u, v)) > 0$, daß u in der Ordnung unter v steht, $f((u, v)) < 0$, daß u in der Ordnung über v steht und $f((u, v)) = 0$, daß u und v sich auf der gleichen Stufe der Ordnung befinden. Dabei wird der Betrag der Bewertung $|f((u, v))|$ einer Kante (u, v) als Abstandsangabe zwischen ihren inzidenten Knoten bezüglich der gesuchten Ordnung verstanden. Da die Abstände reellwertig sind, sollte auch die Ordnung reellwertig sein. Gesucht ist also eine Funktion $l : V \rightarrow \mathbb{R}$. Durch die Werte, die die Funktion l den Knoten zuordnet, entsteht dann die Ordnung bzw. Hierarchie.

Da Richtung und Bewertung der Kanten in dem Graphen die Ordnung auf den Knoten bestimmen sollen, können nur Knotenwerte innerhalb einer schwachen Zusammenhangskomponente einen Bezug zueinander haben. Somit ist es nicht sinnvoll Graphen mit mehr als einer schwachen Zusammenhangskomponente zu betrachten. Wir gehen in diesem Kapitel deshalb immer von einem schwach zusammenhängenden Graphen G aus.

Im folgenden werden verschiedene Ansätze vorgestellt, das Hierarchie-Problem zu formulieren sowie Algorithmen zur Lösung der verschiedenen Problemvarianten. Dabei werden neben reellwertigen Bewertungsfunktionen auch solche betrachtet, die der Kante ein reelles Intervall zuordnen.

5.2.1 Das einfache Hierarchie-Problem

Haben die Kanten in einem Graphen reellwertige (oder auch ganzzahlige) Kantenbewertungen, so ist ein erster intuitiver Ansatz, daß die Bewertung einer Kante den genauen Abstand zwischen ihren beiden inzidenten Knoten in Bezug auf die Hierarchie angibt. Formal läßt sich das folgendermaßen ausdrücken:

Definition 5.1 *Hierarchie-Problem*

Gegeben sei ein gerichteter Graph $G = (V, E)$, eine Kantenbewertung $f : E \rightarrow \mathbb{R}$ und ein Knoten $s \in V$.

Gesucht wird eine Funktion $l_G : V \rightarrow \mathbb{R}$, so daß $l_G(s) = 0$ und $l_G(v) - l_G(u) = f((u, v))$ für jede Kante $(u, v) \in E$ gilt.

Wenn es für einen schwach zusammenhängenden Graphen G eine solche Funktion l_G gibt, ist sie eindeutig bestimmt. Ob es eine solche Funktion gibt, hängt von der Kantenbewertung f ab. Um sie zu berechnen, ist die folgende Konstruktion hilfreich:

Betrachte zu einem Graphen $G = (V, E)$ mit Kantenbewertungsfunktion $f : E \rightarrow \mathbb{R}$ den erweiterten Graphen $G' = (V, E' = \{(u, v) | (u, v) \in E \wedge (v, u) \in E\})$ mit Kantenbewertungsfunktion $f' : E \rightarrow \mathbb{R}$ definiert durch

$$f'((u, v)) = \begin{cases} f((u, v)), & \text{falls } (u, v) \in E \\ -f((v, u)) & \text{sonst} \end{cases}$$

Siehe Abbildung 5.7 für ein Beispiel eines Graphen G mit Kantenbewertung f und den zugehörigen erweiterten Graphen G' mit Kantenbewertung f' .

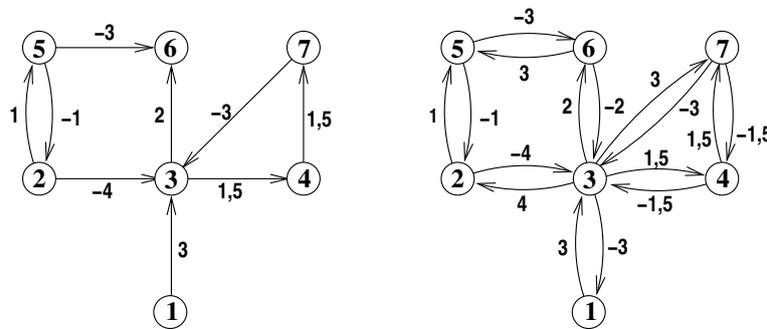


Abbildung 5.7: Links ein gerichteter Graph G mit Kantenbewertung f für einen gerichteten Graph G mit 7 Knoten und rechts der zugehörige erweiterte Graph G' mit Kantenbewertung f' . Die zugehörige Hierarchie l für $G, f, s = v_3$ (und damit auch für $G', f', s = v_3$) ist $l(v_1) = -3, l(v_2) = 4, l(v_3) = 0, l(v_4) = 1,5, l(v_5) = 5, l(v_6) = 2$ und $l(v_7) = 3$.

Lemma 5.2 Sei $G = (V, E)$ ein schwach zusammenhängender Graph mit Kantenbewertung $f : E \rightarrow \mathbb{R}$ und $s \in V$ eine Instanz für das Hierarchie-Problem und G', f' wie oben definiert.

1. Wenn es für diese Instanz eine Lösung l_G gibt, so ist diese eindeutig bestimmt.
2. Es gibt genau dann eine Lösung l_G für die Instanz G, f, s , wenn es eine Lösung $l_{G'}$ für die Instanz G', f', s gibt. Es gilt dann $l_G(v) = l_{G'}(v)$ für alle $v \in V$.

3. Die gesuchte Funktion $l_G = l_{G'}$ existiert genau dann, wenn es für je zwei Knoten $u, v \in V$ eine eindeutig bestimmte Konstante $c \in \mathbb{R}$ gibt, so daß für jeden Weg p von u nach v in G' gilt $f'(p) = c$, und für jeden Weg q von v nach u in G' gilt $f'(q) = -c$. (f' ist dann transitiv, es gilt also für alle $(u, v), (v, w), (u, w) \in E'$, daß $f'((u, v)) + f'((v, w)) = f'((u, w))$).

Beweis: 1.: Seien l_G und l_G^* zwei Lösungen für die Instanz. Betrachte den ungerichteten Graphen $G_{ung} = (V, E_{ung} = \{\{u, v\} \mid (u, v) \in E\})$. Da G schwach zusammenhängend ist, gibt es in G_{ung} für jeden Knoten $v \in V - s$ einen Weg $p = (s = w_1, \dots, w_k = v)$ von s nach v . In G gibt es entweder die Kante (w_i, w_{i+1}) oder die Kante (w_{i+1}, w_i) für $i = 1, \dots, k - 1$. Für jede Kante $(u, v) \in E$ gilt

$$l_G(v) - l_G(u) = f((u, v)) = l_G^*(v) - l_G^*(u).$$

Daraus folgt $l_G(v) - l_G^*(v) = l_G(u) - l_G^*(u)$. Wegen $(s, w_2) \in E$ oder $(w_2, s) \in E$, folgt aus $l_G(s) = l_G^*(s) = 0$ dann $l_G(w_2) = l_G^*(w_2)$ und daraus wiederum $l_G(w_3) = l_G^*(w_3)$ und so weiter. Damit ergibt sich $l_G(v) = l_G^*(v)$ für alle $v \in V$.

2.: Offensichtlich ist jede Lösung l_G für die Instanz G, f, s auch eine Lösung für die Instanz G', f', s und umgekehrt. Die Aussage folgt damit direkt aus 1.

3.: \Rightarrow : Es existiert eine Funktion $l_{G'} : V \rightarrow \mathbb{R}$, so daß $l(s) = 0$ und $l(v) - l(u) = f((u, v))$ für jede Kante $(u, v) \in E$ gilt. Dann gilt auch $l(v) - l(u) = f'((u, v))$ für jede Kante $(u, v) \in E$ und $l(v) - l(u) = -(l(u) - l(v)) = -f(v, u) = f'((u, v))$ für jede Kante $(u, v) \in E' - E$. Da G schwach zusammenhängend ist, ist G' stark zusammenhängend, und somit gibt es in G' für jedes Knotenpaar $u, v \in V$ $u \neq v$ einen Weg $p = (u = w_1, \dots, w_k = v)$ von u nach v . Dann muß gelten $f'(p) = l(v) - l(u)$, denn nach Voraussetzung ist

$$l(v) = l(u) + f'((w_1, w_2)) + \dots + f'((w_{k-1}, w_k)) = l(u) + f'(p).$$

Diese Bedingung gilt für jeden Weg p von u nach v . Durchläuft man den Weg p in umgekehrter Richtung erhält man $l(u) - l(v) = -f(p)$.

\Leftarrow : Sei $u \in V - \{s\}$ und p ein Weg von s nach u . Setze $l(u) = f'(p)$. □

Um die Hierarchie für eine Instanz G, f, s zu bestimmen, kann man also den Graphen $G' = (V, E')$ mit Kostenfunktion f' verwenden. Dazu beginnt man mit der Berechnung beim Startknoten $s \in V$ in G' , der den Wert $l_{G'}(s) = 0$ erhält. Von dort durchläuft man den Graphen wie in einer Breitensuche und berechnet den Hierarchiewert eines Knotens, indem man zu dem Hierarchiewert des Vorgängers das Kantengewicht der gerade betrachteten Kante addiert. Hatte der Knoten schon einen Hierarchiewert, der von dem neu berechneten abweicht, so gibt es keine Lösung. Da der Graph G' stark zusammenhängend ist, werden über eine Breitensuche alle Knoten erreicht.

Zuerst muß aber der Hilfsgraph $G' = (V, E')$ mit Kostenfunktion f' konstruiert werden. Dabei kann man gleich überprüfen, ob es zu einer Kante $(u, v) \in E$ eine Kante $(v, u) \in E$ mit $f(v, u) \neq -f(u, v)$ gibt. In diesem Fall kann es keine Lösung geben, da der Abstand zwischen u und v schon durch $f((u, v))$ festgelegt wird.

Der Algorithmus zur Berechnung von l_G für die Instanz G, f, s zerfällt also in zwei Teile:

1. Überprüfen vorhandener Kantengewichte für die Kanten in E und erstellen des Graphen $G' = (V, E')$ mit Kantengewichtung f' .

2. Berechnung $l_{G'}$ auf dem erweiterten Graphen $G' = (V, E')$ mit Kantenbewertung f'

1. Überprüfen der Kantengewichte und erstellen des Graphen $G' = (V, E')$ sowie f'

setze $Lösung = \text{wahr}$

für alle Knoten $v \in V$ setze Markierung $m(v) = \text{nicht definiert}$

für alle Knoten $v \in V$ {

 für alle in v einlaufenden Kanten (u, v)

 markiere den Knoten u mit $m(u) = f((u, v))$

 für alle aus v auslaufenden Kanten (v, w)

 wenn $m(w)$ nicht definiert dann füge (w, v) mit $f(w, v) = -f(v, w)$ in E ein

 sonst wenn $m(u) \neq -f(v, w)$ dann $Lösung = \text{falsch}$

 für alle in v einlaufenden Kanten (u, v)

 setze $m(u) = \text{nicht definiert}$

}

2. Berechnung $l_{G'}$ auf dem erweiterten Graphen $G' = (V, E')$ mit Kantenbewertung f'

setze $Lösung = \text{wahr}$

initialisiere alle Knoten $v \in V$ mit $l(v) = \text{nicht definiert}$

setze $l(s) = 0$

initialisiere eine Queue Q

füge s in Q ein

solange $Q \neq \emptyset$ und $Lösung$ {

 entnehme eine Knoten u aus Q

 für auslaufenden Kanten (u, v) {

 wenn $l(v)$ nicht definiert dann {

 setze $l(v) = l(u) + f((u, v))$

 füge v in Q ein

 }

 sonst wenn $l(u) + f((u, v)) \neq l(v)$ dann $Lösung = \text{falsch}$

 }

}

Wird zu irgendeinem Zeitpunkt $Lösung$ false, so bricht die Berechnung ab. Ist beim Terminieren des Algorithmus $Lösung$ true, so hat jedes $v \in V$ einen Wert $l(v)$ erhalten und diese Werte erfüllen alle die geforderte Bedingung. Die Laufzeit des ersten Teiles ist in $O(|V| + |E|)$, die des zweiten in $O(|V| + |E'|)$. Da $|E'| \leq 2|E|$ ergibt sich eine Gesamtlaufzeit in $O(|V| + |E|)$.

5.2.2 Das Intervall-Hierarchie-Problem

Wir wollen jetzt Bewertungsfunktionen betrachten, die einer Kante keinen festen Wert, sondern einen Wertebereich zuordnen. Für unser Problem bedeutet das anschaulich, daß ein Bereich angegeben wird, in dem der Abstand zwischen zwei Knoten liegen soll.

Dafür benötigen wir zunächst einige Definitionen. Sei $I(\mathbb{R}) = \{[a, b] \mid a, b \in \mathbb{R}, a \leq b\}$, also die Menge der Intervalle, die durch zwei reelle Zahlen begrenzt werden. Es gilt für $x \in \mathbb{R}$: $x \in [a, b] \Leftrightarrow a \leq x \leq b$. In diesen Zusammenhang betrachten wir auch $[a, a]$, $a \in \mathbb{R}$, als

Intervall, obwohl es nur einen Wert enthält.

Der Schnitt von zwei Intervallen $[a, b], [c, d] \in I(\mathbb{R})$ ist $[a, b] \cap [c, d] = [\max\{a, c\}, \min\{b, d\}]$, $a, b, c, d \in \mathbb{R}$. Ist $\max\{a, c\} > \min\{b, d\}$, so ist das Ergebnis die leere Menge. Ist $\max\{a, c\} = \min\{b, d\}$, so ist das Ergebnis das Intervall, das nur diesen einen Wert enthält.

Die Addition von zwei Intervallen $[a, b], [c, d] \in I(\mathbb{R})$ definieren wir als

$$[a, b] + [c, d] := [a + c, b + d].$$

Da $a \leq b$ und $c \leq d$ ist das Ergebnis wieder ein Intervall.

Um ein Inverses eines Intervalls zu erhalten, definieren wir die Funktion

$$\text{inv} : I(\mathbb{R}) \rightarrow I(\mathbb{R}), \quad \text{inv}([a, b]) := [-b, -a].$$

Da $a \leq b$ und somit $-a \geq -b$ ist das Ergebnis wieder ein Intervall. Man beachte, daß für ein Intervall $[a, b]$ mit $a \neq b$ gilt $[a, b] + \text{inv}([a, b]) = [a, b] + [-b, -a] = [a - b, b - a] = [-(b - a), b - a] \neq [0, 0]$. Man erhält also nicht das Intervall $[0, 0]$ (was bei der Addition von zwei Intervallen, die mehr als einen Punkt enthalten, auch nicht möglich ist), sondern ein um 0 symmetrisches Intervall.

Sei $G = (V, E)$ ein gerichteter Graph mit Kantenbewertungsfunktion $f : E \rightarrow I(\mathbb{R})$. Sei $p = (v_1, \dots, v_k)$ ein Weg in G . Die Länge eines Weges ist wieder

$$f(p) = \sum_{i=1}^{k-1} f((v_i, v_{i+1}))$$

mit der definierten Addition für Intervalle. Für $u, v \in V$ sei

$$P_{u \rightarrow v}^G := \{p = (u = w_1, \dots, w_k = v) \mid w_1, \dots, w_k \in V \wedge p \text{ ist Weg von } u \text{ nach } v \text{ in } G\}$$

die Menge aller Wege von u nach v in G .

Jetzt können wir das Hierarchieproblem für Graphen, deren Kantenbewertungen Intervalle sind, formulieren.

Definition 5.3 Intervall-Hierarchie-Problem

Gegeben sei ein gerichteter Graph $G = (V, E)$ mit Kantenbewertung $f : E \rightarrow I(\mathbb{R})$ und $s \in V$. Gesucht wird eine Funktion $l : V \rightarrow \mathbb{R}$, so daß $l(s) = 0$ und $l(v) - l(u) \in f((u, v))$ für jede Kante $(u, v) \in E$.

Ob es für eine Instanz G, f, s eine Lösung gibt, hängt wieder von der Kantenbewertung f ab. Gibt es eine Lösung, so muß diese nicht eindeutig sein.

Auch hier definieren wir zur Erleichterung der Berechnung einer Lösung wieder einen Hilfsgraphen. Für einen Graphen $G = (V, E)$ mit Kantenbewertungsfunktion $f : E \rightarrow \mathbb{R}$ sei der erweiterte Graph G' definiert durch $G' = (V, E' = \{(u, v) \mid (u, v) \in E \wedge (v, u) \in E\})$ mit Kantenbewertungsfunktion $f' : E \rightarrow \mathbb{R}$

$$f'((u, v)) = \begin{cases} f((u, v)) \cap \text{inv}(f((v, u))), & \text{falls } (u, v) \in E \text{ und } (v, u) \in E \\ \text{inv}(f((v, u))) & \text{sonst} \end{cases}$$

Insgesamt erhält man so einen Graphen G' mit Kantenbewertung f' , in dem für jede Kante $(u, v) \in G'$ gilt $f'((u, v)) = \text{inv}(f'((v, u)))$. In Abbildung 5.8 ist ein Beispiel eines Graphen G mit Kantenbewertung f und den zugehörigen erweiterten Graphen G' mit Kantenbewertung f' zu sehen.

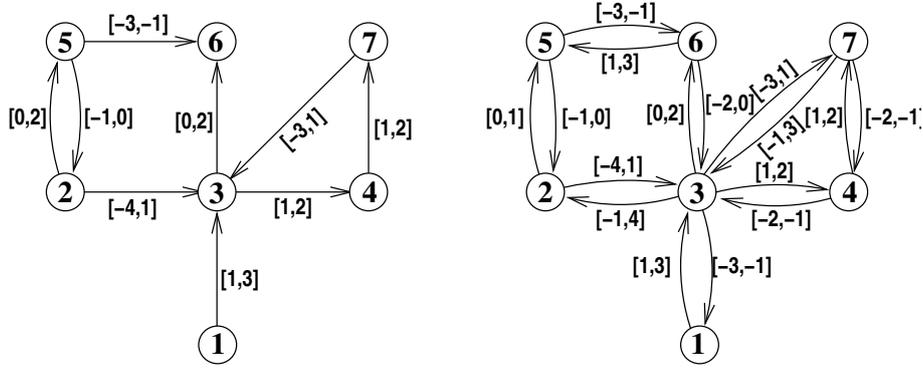


Abbildung 5.8: Links ein gerichteter Graph G mit Kantenbewertung f für einen gerichteten Graph G mit 7 Knoten und rechts der zugehörige erweiterte Graph G' mit Kantenbewertung f' . Eine mögliche Hierarchie l für $G, f, s = v_3$ (beziehungsweise $G', f', s = v_3$) ist $l(v_1) = -1, l(v_2) = 1, l(v_3) = 0, l(v_4) = 1, l(v_5) = 1, l(v_6) = 0$ und $l(v_7) = 2$.

Lemma 5.4 Sei $G = (V, E)$ ein schwach zusammenhängender Graph mit Kantenbewertung $f : E \rightarrow \mathbb{R}$ und $s \in V$ eine Instanz für das Intervall-Hierarchie-Problem.

1. Jede Lösung l_G für die Instanz G, f, s ist auch Lösung $l_{G'}$ für die Instanz G', f', s und umgekehrt.
2. Eine Lösung $l_G = l_{G'}$ existiert genau dann, wenn für je zwei Knoten $u, v \in V$ der Schnitt der Längen aller Wege von u nach v in G' nicht leer ist, also $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p) \neq \emptyset$.
3. Es existiert genau dann ein Lösung $l_G = l_{G'}$, wenn für jeden Kreis K in G' $0 \in f'(K)$ ist.

Beweis: Der Beweis von 1. und 2. kann analog zu dem von Lemma 5.2 geführt werden. Zu 1. ist dabei folgendes zu beachten: Gibt es zu einer Kante $(u, v) \in E$ mit Kantenbewertung $f((u, v)) = [a, b]$ die Kante $(v, u) \in E$ mit Kantenbewertung $f((v, u)) = [c, d]$, so muß für eine Lösung l_G sowohl $l_G(v) - l_G(u) \in [a, b]$ als auch $l_G(v) - l_G(u) \in [-d, -c]$ und somit $l_G(v) - l_G(u) \in [a, b] \cap [-d, -c]$ gelten.

3.: Gibt es eine Lösung $l_{G'}$, dann gilt für jeden Weg $p = (v_1, \dots, v_k)$ die Beziehung $l_{G'}(v_k) - l_{G'}(v_1) \in f'(p)$. Für einen Kreis $K = (v_1, \dots, v_k)$ gilt somit $l_{G'}(v_1) - l_{G'}(v_1) \in f'(K)$, also $0 \in f'(K)$. Gibt es andererseits einen Kreis $K = (v_1, \dots, v_k)$ mit $0 \notin f'(K)$, so kann es keine Funktion $l_{G'}$ geben, für die $0 = l_{G'}(v_1) - l_{G'}(v_1) \in f'(K)$ gilt, und somit keine Lösung. \square

Gibt es eine Lösung l_G für eine Instanz G, f, s , so ist diese im allgemeinen nicht eindeutig. Wenn für je zwei Knoten $u, v \in V$ der Schnitt der Längen aller Wege von u nach v in G' nicht leer ist, also $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p) \neq \emptyset$, so kann man ein c aus dem Schnitt wählen. Es gilt für jeden Weg p von u nach v in G' : $c \in f'(p)$. Es gibt dann für jeden Weg $p = (u = w_1, \dots, w_k = v)$

von u nach v in G' Konstanten $c_i \in f(w_i, w_{i+1})$, $i = 1, \dots, k-1$, so daß $\sum_{i=1}^{k-1} c_i = c$ ist. Daraus folgt unmittelbar, daß es immer eine triviale Lösung gibt, wenn jedes Intervall, das als Kantengewicht verwendet wird, die Null enthält. Dann enthält auch eine beliebige Summe von Intervallen immer die Null. In dem Fall kann man auch alle obigen Konstanten als Null wählen.

Damit es eine Lösung gibt, muß für alle $u, v \in V$ gelten, daß der Schnitt der Gewichte aller Wege von u nach v nicht leer ist, also $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(g) \neq \emptyset$. Daraus folgt, daß es insbesondere dann keine Lösung gibt, wenn für ein $e \in E'$ gilt: $f'(e) = \emptyset$.

Um eine Lösung zu ermitteln, muß man jedoch nicht zu jedem Knotenpaar $u, v \in V$ das Intervall berechnen, in dem der Abstand der beiden Knoten also $l_{G'}(v) - l_{G'}(u)$ liegen muß. Es reicht für alle Knoten die Abstandsintervalle zu einem festen Knoten $s \in V$ zu berechnen, wie daß folgende Lemma zeigt. Für einen Knoten $v \in V - \{s\}$ werden wir dieses Abstandsintervall im folgenden als $I_s(v)$ bezeichnen, also $I_s(v) := \bigcap_{p \in P_{s \rightarrow v}^{G'}} f'(p)$. Für s setzen wir $I_s(s) = [0, 0]$.

Lemma 5.5 *Sei $G = (V, E)$ ein schwach zusammenhängender Graph mit Kantenbewertung $f : E \rightarrow I(\mathbb{R})$, $G' = (V, E', f')$ der erweiterte Graph und $s \in V$. Gilt für ein Paar $u, v \in V$ mit $I_s(v) \neq \emptyset$ und $I_s(u) \neq \emptyset$, dann folgt*

$$\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p) \neq \emptyset.$$

Beweis: Durch Widerspruch: Wir zeigen aus $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p) = \emptyset$ folgt $I_s(u) = \emptyset$.

Angenommen $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p) = \emptyset$. Dann gibt es zwei Wege $p = u, w_1, \dots, w_m, v$ und $\hat{p} = u, x_1, \dots, x_n, v$ von u nach v mit $f'(p) \cap f'(\hat{p}) = \emptyset$. Sei $f'(p) = [a, b]$ und $f'(\hat{p}) = [c, d]$. Dann ist entweder $a > d$ oder $c > b$. Sei ohne Einschränkung $a > d$.

Wir definieren rekursiv Wege von s nach u . Sei

$$p_0 = s, y_1, \dots, y_l, u$$

ein Weg von s nach u mit $f'(p_0) = [e, f]$. Für $i > 0$ sei

$$p_i = p_{i-1} w_1, \dots, w_m, v, x_n, \dots, x_1, u.$$

Also beginnt der Weg $p_1 = s, y_1, \dots, y_l, u, w_1, \dots, w_m, v, x_n, \dots, x_1, u$ mit dem Weg p_0 , dann werden die Knoten von p durchlaufen, und zum Schluß folgen die Knoten von \hat{p} in umgekehrter Reihenfolge. Der Weg p_1 hat die Länge $f'(p_1) = [e + a - d, f + b - c]$. Allgemein gilt $f'(p_i) = [e + i(a - d), f + i(b - c)]$ für jedes $i \in \mathbb{N}_0$. Da $a > d$ ist $a - d > 0$. Damit gibt es ein $i \in \mathbb{N}$ mit $e + i(a - d) > f$. Daraus folgt dann

$$\bigcap_{i \in \mathbb{N}_0} f'(p_i) = \emptyset.$$

Da aber $I_s(u) \subseteq \bigcap_{i \in \mathbb{N}_0} f'(p_i)$ ist dann auch $I_s(u) = \emptyset$. □

Ist $I_s(v) = \emptyset$ oder $I_s(u) = \emptyset$, dann kann es keine Lösung geben. Also reicht es, für alle $v \in V$ das Intervall $I_s(v)$ zu berechnen, um ein gültige Lösung für das Intervall-Hierarchie-Problem zu finden. In Abbildung 5.9 sind für den Graphen G' aus Abbildung 5.8 die Intervalle I_s angegeben.

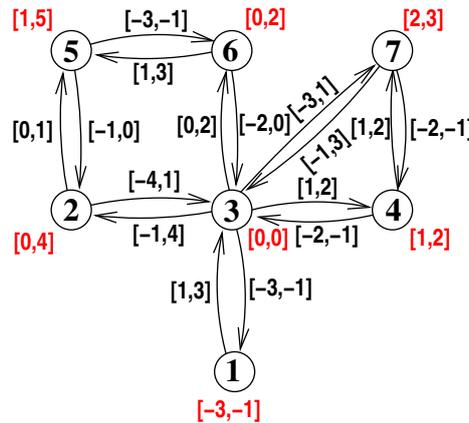


Abbildung 5.9: In rot sind die Intervalle $I_s(u)$, $u \in V$ und $s = v_3$ für den erweiterten Graphen G' aus Abbildung 5.8 angegeben

Für die Berechnung einer Lösung für eine Instanz G, f, s kann nach Lemma 5.4 der stark zusammenhängende Graph $G' = (V, E')$ mit Kostenfunktion f' verwendet werden. Für diesen werden dann die Intervalle $I_s(v)$ für alle $v \in V - \{s\}$ berechnet. Für den Knoten s ist das Intervall $I_s(s) = [0,0]$. Es gilt dann für jede Lösung $l_{G'}$ und jeden Knoten $v \in V$, daß $l_{G'}(v) \in I_s(v)$ ist.

Zur Berechnung der Intervalle $I_s(v)$, $v \in V$, kann ähnlich wie bei einem kürzeste Wege Algorithmus vorgegangen werden. Dazu wird für jede Kante $(u, v) \in E'$ überprüft, ob $(I_s(u) + f'((u, v))) \cap I_s(v) \subsetneq I_s(v)$ und gegebenenfalls das Intervall $I_s(v)$ entsprechend verkleinert. Die Reihenfolge der Vergleiche wird durch eine Queue bestimmt, in die Knoten v eingefügt werden, wenn das Intervall $I_s(v)$ verkleinert wird. Wird ein Knoten aus der Queue entnommen, werden alle seine auslaufenden Kanten betrachtet. Als Startknoten wird s in die Queue eingefügt, der das Intervall $I_s(s) = [0,0]$ erhält.

Der Algorithmus zur Berechnung der Hierarchie gliedert sich in zwei Teile:

1. Einfügen zusätzlicher Kanten und Anpassung der Kantengewichte
2. Berechnung der Intervalle $I_s(v)$ für die Knoten $v \in V - s$

1. Einfügen zusätzlicher Kanten und Anpassung der Kantengewichte

setze *Lösung*=wahr

für alle Knoten $v \in V$ setze Markierung $m(v) =$ nicht definiert

für alle Knoten $v \in V$ {

für alle in v einlaufenden Kanten (u, v)

markiere den Knoten u mit $m(u) = f((u, v))$

für alle aus v auslaufenden Kanten (v, w)

wenn $m(w)$ nicht definiert dann {

füge (w, v) in E ein, $f(w, v) = \text{inv}(f(v, w))$

}

sonst wenn $m(w) \cap \text{inv}(f(v, w)) = \emptyset$ dann *Lösung*=falsch

sonst wenn $m(w) \cap \text{inv}(f(v, w)) \neq m(w)$ dann {

$$f(v, w) = f(v, w) \cap \text{inv}(m(w))$$

}

für alle in v einlaufenden Kanten (u, v) {

entferne die Markierung $m(u)$ von u

}

}

2. Berechnung der Intervalle $I_s(v)$ für die Knoten $v \in V - s$

setze *Lösung*=wahr

initialisiere zu jeden Knoten $v \in V$ ein Intervall $I_s(v) = [-\infty, \infty]$

setze $I_s(s) = [0, 0]$

initialisiere eine Queue Q

füge s in Q ein

solange $Q \neq \emptyset$ und *Lösung*{ (*)

entnehme eine Knoten u aus Q

für alle aus u auslaufenden Kanten (u, v) {

wenn $(I_s(u) + f'((u, v))) \cap I_s(v) \subsetneq I_s(v)$ dann {

$I_s(v) = (I_s(u) + f'((u, v))) \cap I_s(v)$

wenn $I_s(v) \neq \emptyset$ und $v \notin Q$ dann füge v in Q ein

sonst *Lösung*=falsch

}

}

}

Der Algorithmus zur Berechnung der Intervalle für die Knoten terminiert, wenn es keine Lösung gibt oder wenn sich die Intervallgrenzen nicht mehr ändern. Im zweiten Fall wurde zu jedem Knoten das gesuchte Intervall berechnet. Dieses kann auch jeweils nur aus einem Punkt bestehen. Solange es aber diesen einen Wert enthält, gibt es auch eine Lösung.

Wenn mindestens eines der berechneten Intervalle mehr als einen Punkt enthält, gibt es mehr als eine (unendlich viele) Lösung. Eine gültige Lösung erhält man dann zum Beispiel, indem man für jeden Knoten $v \in V$ die Untergrenze von $I_s(v)$ als $l_{G'}(v)$ wählt. Alternativ kann man zu jedem Knoten $v \in V$ auch die Obergrenze von $I_s(v)$ als $l_{G'}(v)$ wählen. Da jeder Wert in $I_s(v)$ ein möglicher Wert für $l_{G'}(v)$ ist, kann man für einen Knoten v auch einen beliebigen Wert $c \in I_s(v)$ als $l_{G'}(v) = c$ wählen. Diese Wahl kann aber den Wertebereich der anderen Knoten weiter einschränken. Deshalb müssen dann die Intervalle der anderen Knoten neu berechnet werden. Dazu fügt man den Knoten v als neuen Startknoten in Q mit $I_s(v) = [c, c]$ ein und startet die Schleife (*) im 2. Algorithmus erneut, wobei für die anderen Knoten die zuvor berechneten Intervalle als Startwerte verwendet werden. Es findet also keine neue Initialisierung statt. Die mit dieser neuen Eingabe berechneten Intervalle geben die möglichen Werte der Knoten an, unter der Bedingung, daß $l_{G'}(s) = 0$ und $l_{G'}(v) = c$. Da c innerhalb des vorher berechneten Wertebereichs liegt, gibt es auch jetzt wieder eine Lösung. Man kann so sukzessive für jeden Knoten einen Wert wählen. Dabei werden allerdings die Intervalle für die noch nicht fest bewerteten Knoten mit jeden gewählten Wert weiter eingeschränkt.

Für die Laufzeitanalyse sind die folgenden Überlegungen hilfreich. Nach Lemma 5.4 gilt: Gibt es eine Lösung $l_{G'}$, so ist für jeden Kreis K $0 \in f'(K)$. Dann gilt aber $[a, b] \subseteq [a, b] + f'(K)$

für alle Intervalle $[a, b] \in I(\mathbb{R})$, da die Untergrenze von $f'(K)$ kleiner oder gleich Null und die Obergrenze größer oder gleich Null sein muß. Also wird in diesem Fall kein Intervall durch das Durchlaufen eines Kreises verkleinert und es sind für die Berechnung der Intervalle nur die einfachen Wege relevant. Daraus folgt, daß die Laufzeit des Algorithmus in $O(|V| \cdot |E|)$ ist, wenn es eine Lösung gibt.

Gibt es keine Lösung, terminiert der Algorithmus, wenn für einen Knoten v $I_s(v) = \emptyset$ wird. Es gibt dann einen Kreis mit K mit $0 \notin f'(K)$. Dann gilt $[a, b] \not\subseteq [a, b] + f'(K)$ für alle Intervalle $[a, b] \in I(\mathbb{R})$, da die Untergrenze und die Obergrenze von $f'(K)$ entweder beide kleiner oder beide größer als Null sein müssen. Die Laufzeit ist dann abhängig von der Länge der Kreise K mit $0 \notin f'(K)$ und der Größe von $I_s(v)$ für den ersten Knoten v , der in einem solchen Kreis betrachtet wird. Sie läßt sich also nicht mehr durch die Größe der Instanz abschätzen.

Man kann aber den zweiten Algorithmus auch nach $|V| \cdot |E|$ Schritten beenden, da jede Lösung in dieser Zeit gefunden wird. Der erste Teil der Berechnung hat wieder eine Laufzeit in $O(|V| + |E|)$, so daß sich dann auch insgesamt eine Laufzeit von $O(|V| \cdot |E|)$ ergibt.

Eine Alternative Berechnungsmöglichkeit für Teil 2 (Berechnung der Intervalle für die Knoten) erhält man durch folgende Überlegung. Für die Berechnung der Obergrenze eines Intervalls $I_s(v)$, $v \in V$, sind nur die Werte der Obergrenzen der Intervalle auf den Wegen von s nach v relevant. Aus der Definition des Schnittes zweier Intervalle $[a, b] \cap [c, d] = [\max\{a, c\}, \min\{b, d\}]$, folgt, daß die kleinere der beiden Obergrenzen die neue Obergrenze bildet. Man sucht also eigentlich die kleinste Obergrenze aller Wege von s nach v , um die obere Grenze von $I_s(v)$ zu bestimmen. Dies entspricht genau einer kürzesten-Wege-Suche, bei der als Kantenbewertung die Intervallgrenzen der Kantenbewertung f' verwendet werden.

Analoge Überlegungen zeigen, daß ist die Berechnung der Untergrenzen der Intervalle $I_s(v)$, $v \in V$ als eine längste-Wege-Suche auf den unteren Intervallgrenzen der Kantenbewertung f' betrachtet werden kann. Diese kann man auch mit einem kürzeste-Wege-Algorithmus umsetzen.

Man kann diese Version des Problems also auf ein kürzeste-Wege-Problem zurückführen. Dieses hat bekanntlich keine Lösung, wenn es in dem Graphen negative Kreise gibt. Tritt bei den Obergrenzen der Intervalle ein negativer Kreise auf, so müssen auch die entsprechenden Untergrenzen, die ja kleiner sind, einen negativen Kreis bilden. In dem Fall gibt es einen Kreis K , so daß $f'(K)$ negative Ober- und Untergrenzen hat, also $0 \notin f'(K)$. Dann hat aber wie oben gezeigt auch das Intervall-Hierarchie-Problem keine Lösung.

Zur Vereinfachung der Notation definieren wir $og : I(\mathbb{R}) \rightarrow \mathbb{R}$ durch $og([a, b]) := b$ und $ug : I(\mathbb{R}) \rightarrow \mathbb{R}$ durch $ug([a, b]) := a$. Für die Kürzeste-Wege-Suche auf den Obergrenzen der Kantenbewertungen f' definieren wir eine Kantenbewertung $og_{E'} : E' \rightarrow \mathbb{R}$ durch $og_{E'}(e) = og(f'(e))$ und entsprechend für die Längste-Wege-Suche auf den Untergrenzen eine Kantenbewertung $ug_{E'} : E' \rightarrow \mathbb{R}$ mit $ug_{E'}(e) = ug(f'(e))$. Die Kürzeste-Wege-Suche liefert als Ergebnis eine Funktion $kw : V \rightarrow \mathbb{R} \cup -\infty$ und die Längste-Wege-Suche entsprechend eine Funktion $lw : V \rightarrow \mathbb{R} \cup \infty$. Die Funktionen $og_V : V \rightarrow \mathbb{R}$ und $ug_V : V \rightarrow \mathbb{R}$ geben für jeden Knoten $v \in V$ die berechnete Ober- beziehungsweise Untergrenze von $I_s(v)$ an. Damit können wir die zweite Version der Berechnung der Knotenintervalle I_s wie folgt angeben.

Berechnung der Intervalle $I_s(v)$ für die Knoten $v \in V - s$ (Version 2)

setze *Lösung*=wahr

für alle Kanten $e \in E'$ setze $og_{E'}(e) = og(f'(e))$
 führe Kürzeste-Wege-Suche auf $G' = (V, E')$ mit Kantenbewertung $og_{E'}$ und Startknoten s durch

für alle Knoten $v \in V$

wenn ($kw(v) > -\infty$) dann $og_V(v) = kw(v) \setminus \setminus$ kürzester Weg von s nach v ist definiert
 sonst $Lösung = falsch$

für alle Kanten $e \in E'$ setze $ug_{E'}(e) = ug(f'(e))$

führe Längste-Wege-Suche auf $G' = (V, E')$ mit Kantenbewertung $ug_{E'}$ und Startknoten s durch

für alle Knoten $v \in V$

wenn ($-w(v) < \infty$) dann $ug_V(v) = lw(v) \setminus \setminus$ längster Weg von s nach v definiert
 sonst $Lösung = falsch$

wenn $Lösung$ dann

für alle Knoten $v \in V$

wenn $ug_V(v) \leq og_V(v)$ dann setze $I(v) = [ug_V(v), og_V(v)]$
 sonst $Lösung = falsch$

Ist nach dem Terminieren des Algorithmus $Lösung$ false, so gibt es keine Lösung. Ansonsten wurden die Intervalle berechnet. Dazu wird zweimal eine kürzeste-Wege-Suche durchgeführt. Diese ist mit Dijkstras Algorithmus (und einer Implementierung der dort verwendeten Priority-Queue mit Fibonacci-Heaps) mit einer Laufzeit in $O(|E| + |V| \cdot \log |V|)$ möglich. Damit ergibt sich auch zusammen mit der Vorverarbeitungsphase eine Laufzeit in $O(|E| + |V| \cdot \log |V|)$.

5.2.3 Das Δ -Hierarchie-Problem

Gibt es keine exakte Lösung für das Hierarchie-Problem, möchte man eine möglichst gute Näherungslösung bekommen. Diese sollte eine möglichst geringe Abweichung von den Kriterien für eine exakte Lösung haben. Eine Möglichkeit, das zu erreichen, ist, daß man die obere Grenze für die Abweichung von jedem einzelnen Kriterium minimiert. Damit erhält man folgende Problemdefinition:

Definition 5.6 Δ -Hierarchie-Problem

Gegeben sei ein gerichteter Graph $G = (V, E)$, eine Kantenbewertung $f : E \rightarrow \mathbb{R}$ und ein Knoten $s \in V$.

Gesucht wird ein minimales $\Delta_G \in \mathbb{R}_0^+$, so daß es eine Funktion $l : V \rightarrow \mathbb{R}$ gibt, mit $l(s) = 0$ und $f((u, v)) - \Delta_G \leq l(v) - l(u) \leq f((u, v)) + \Delta_G$ für alle $(u, v) \in E$.

Siehe Abbildung 5.10 für ein Beispiel eines Graphen G mit Kantenbewertung f .

Da es sich hier um ein Optimierungsproblem handelt, gibt es immer eine Lösung Δ_G und Δ_G ist eindeutig bestimmt. Es gilt:

Lemma 5.7 Sei $G = (V, E)$, f, s eine Instanz für das Δ -Hierarchie-Problem und Δ_G eine Lösung. Dann ist Δ_G maximal so groß, wie der größte Betrag der Kantengewichte, also $\Delta_G \leq \max\{ |f(e)| \mid e \in E \}$.

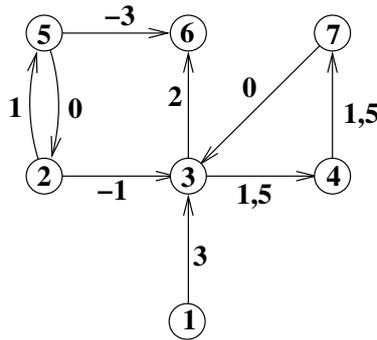


Abbildung 5.10: Ein gerichteter Graph $G = (V, E)$ mit 7 Knoten und Kantenbewertung f . Egal welchen Knoten $s \in V$ man als Startknoten wahlt, dieser Graph hat keine Losung fur das einfache Hierarchie-Problem.

Beweis: Fur $\Delta_G = \max\{ |f(e)| \mid e \in E \}$ gilt $f((u, v)) - \Delta \leq 0 \leq f((u, v)) + \Delta$ fur alle $(u, v) \in E$ und dann erfullt $l \equiv 0$ alle Ungleichungen $f((u, v)) - \Delta_G \leq l(v) - l(u) \leq f((u, v)) + \Delta_G$. \square

Wenn fur eine Instanz G, f, s das minimale Δ_G bestimmt ist, kann man damit eine Hierarchie l_G auf G berechnen, indem man die Kantengewichte zu Intervallen umwandelt. Man definiere dafur eine neue Kantenbewertung $f^I : E \rightarrow I(\mathbb{R})$ durch $f^I(e) = [f(e) - \Delta, f(e) + \Delta]$. Dann ist G, f^I, s eine Instanz fur das Intervall-Hierarchie-Problem. Allerdings ist in dem Fall klar, da eine Losung existiert. Sonst ware das Δ falsch gewahlt.

Wenn es in G Kanten $(u, v), (v, u) \in E$ mit $f((u, v)) \neq -f((v, u))$ gibt, so mussen beide Werte bei der Berechnung von Δ_G berucksichtigt werden. Es ist also nicht einfach moglich, wie beim einfachen Hierarchieproblem den erweiterten Graphen G' fur die Berechnung zu erstellen. Deshalb fuhren wir die folgenden Konventionen ein:

Ein Pfad $P = (u_1, \dots, u_k, E_P)$ in einem Graphen $G = (V, E)$ ist eine Knotenfolge u_1, \dots, u_k mit Kantenmenge $E_P = \{e_1, \dots, e_{k-1}\} \subseteq E$, so da fur jedes Knotenpaar $u_i, u_{i+1}, 1 \leq i \leq k-1$, entweder (u_i, u_{i+1}) oder (u_{i+1}, u_i) eine gerichtete Kante in E_P ist. $(u_i, u_{i+1}) \in E_P$ ist eine Vorwartskante und $(u_{i+1}, u_i) \in E_P$ eine Ruckwartskante in Pfad P . Sei $VK_P = \{(u_i, u_{i+1}) \in E_P \mid 1 \leq i \leq k-1\}$ die Menge aller Vorwartskanten und $RK_P = \{(u_{i+1}, u_i) \in E_P \mid 1 \leq i \leq k-1\}$ die Menge aller Ruckwartskanten in P . Die Lange eines Pfades $P = (u_1, \dots, u_k, E_P)$ in einem kantenbewerteten Graphen ist die Summe der Kostenwerte (Langen) aller Vorwartskanten minus die Summe der Kostenwerte (Langen) aller Ruckwartskanten, also

$$f(P) = \sum_{e \in VK_P} f(e) - \sum_{e \in RK_P} f(e)$$

$K = (u_1, \dots, u_k, E_K)$ mit Kantenmenge $E_K = \{e_1, \dots, e_k\} \subseteq E$ heit ein Kreispfad, wenn $K = u_1, \dots, u_k, E_K - \{e_k\}$ ein Pfad ist und $e_k = (u_k, u_1)$ oder $e_k = (u_1, u_k)$. Die Lange von K ist wiederum $f(K) = \sum_{e \in VK_K} f(e) - \sum_{e \in RK_K} f(e)$.

Lemma 5.8 Sei $G = (V, E), f, s$ eine Instanz fur das Δ -Hierarchie-Problem und Δ_G eine Losung. Dann gibt es fur je zwei Knoten $u, v \in V$ eine Konstante $c \in \mathbb{R}$ gibt, so da folgendes

gilt: Für jeden Pfad $P = (u = u_1, \dots, u_k = v, E_P)$ von u nach v gibt es $k - 1$ Konstanten $f(e) - \Delta_G \leq c_e \leq f(e) + \Delta_G$, $e \in E_P$, so daß $\sum_{e \in VK_P} c_e - \sum_{e \in RK_P} c_e = c$ ist.

Beweis: Die Aussage folgt mit derselben Argumentation wie im letzten Abschnitt direkt aus der Tatsache, daß Δ_G eine Lösung für das Δ -Hierarchie-Problem ist. \square

Wenn für alle Knotenpaare $u, v \in V$ die Länge aller Pfade von u nach v gleich ist, kommt man mit $\Delta_G = 0$ aus. Entscheidend für die Berechnung von Δ_G sind also die Pfade $P_1 = (u = u_1, \dots, u_k = v, E_{P_1}), P_2 = (u = v_1, \dots, v_l = v, E_{P_2})$ zwischen zwei Knoten $u, v \in V$, für die die Länge unterschiedlich ist, also $f(P_1) \neq f(P_2)$. Um eine optimale Lösung zu finden, möchte man diese Differenz möglichst gleichmäßig auf die beiden Pfade verteilen.

Dazu ist folgende Überlegung hilfreich: Da man jeden Pfad auch in umgekehrter Richtung durchlaufen kann, bilden die Kanten aus P_1 und die Kanten aus P_2 einen Kreispfad $K = (u_1, \dots, u_k = v_l, v_{l-1}, \dots, v_2, E_{P_1} \cup E_{P_2})$. Da in K die Kanten von E_{P_1} in derselben Richtung durchlaufen werden wie in P_1 aber die Kanten von E_{P_2} in K in umgekehrter Richtung durchlaufen werden wie in P_2 , ist die Länge von K $f(K) = f(P_1) - f(P_2)$. Da $f(P_1) \neq f(P_2)$ ist $f(K) = c_K \neq 0$. Dieses c_K soll gleichmäßig auf alle Kanten des Kreises verteilt werden, um ein minimales Δ zu erhalten.

Satz 5.9 Sei $G = (V, E)$, f, s eine Instanz für das Δ -Hierarchie-Problem und

$$\Delta_{max}^G = \max\left\{ \frac{|f(K)|}{|E_K|} \mid K = (u_1, \dots, u_k, E_K) \text{ ist Kreispfad in } G \right\}.$$

Dann ist Δ_{max}^G Lösung für das Δ -Hierarchie-Problem.

Beweis: Der Beweis gliedert sich in zwei Teile: Zum einen muß gezeigt werden, daß es eine Funktion $l : V \rightarrow \mathbb{R}$ gibt, mit $l(s) = 0$ und $f((u, v)) - \Delta_{max}^G \leq l(v) - l(u) \leq f((u, v)) + \Delta_{max}^G$ für alle $(u, v) \in E$. Zum anderen, daß es kein $\Delta < \Delta_{max}^G$ gibt, daß diese Bedingung erfüllt.

1.: Wir zeigen zuerst, daß es kein $\Delta < \Delta_{max}^G$ gibt, das eine Lösung für das Δ -Hierarchie-Problem ist.

Für eine Lösung Δ_G für das Δ -Hierarchie-Problem, gilt

$$f((u, v)) - \Delta_G \leq l(v) - l(u) \leq f((u, v)) + \Delta_G$$

und

$$-f((u, v)) - \Delta_G \leq l(u) - l(v) \leq -f((u, v)) + \Delta_G$$

für alle $(u, v) \in E$. Daraus folgt für $(u, v), (v, w) \in E$:

$$f((u, v)) - \Delta_G + f((v, w)) - \Delta_G \leq l(v) - l(u) + l(w) - l(v) \leq f((u, v)) + \Delta_G + f((v, w)) + \Delta_G$$

\Leftrightarrow

$$f((u, v)) + f((v, w)) - 2 \cdot \Delta_G \leq l(w) - l(u) \leq f((u, v)) + f((v, w)) + 2 \cdot \Delta_G$$

und für $(u, v), (w, v) \in E$:

$$f((u, v)) - \Delta_G - f((w, v)) - \Delta_G \leq l(v) - l(u) - (l(v) - l(w)) \leq f((u, v)) + \Delta_G - f((w, v)) + \Delta_G$$

$$\Leftrightarrow$$

$$f((u, v)) - f((w, v)) - 2 \cdot \Delta_G \leq l(w) - l(u) \leq f((u, v)) - f((w, v)) + 2 \cdot \Delta_G.$$

Damit ergibt sich induktiv für Kreispfade K in G , die in $v \in V$ beginnen und enden folgende Bedingung:

$$\sum_{e \in VK_K} (f(e) - \Delta_G) - \sum_{e \in RK_K} (f(e) - \Delta_G) \leq l(v) - l(v) \leq \sum_{e \in VK_K} (f(e) + \Delta_G) - \sum_{e \in RK_K} (f(e) + \Delta_G)$$

$$\Leftrightarrow$$

$$f(K) - |E_K| \cdot \Delta_G \leq 0 \leq f(K) + |E_K| \cdot \Delta_G$$

Sei $\Delta < \Delta_{max}^G$. Dann gibt es einen Kreispfad $K = (u_1, \dots, u_k, E_K)$ in G mit

$$\frac{|f(K)|}{|E_K|} > \Delta$$

Ist $f(K) \geq 0$, dann folgt

$$f(K) > \Delta \cdot |E_K| \quad \Leftrightarrow \quad f(K) - \Delta \cdot |E_K| > 0$$

Ist $f(K) \leq 0$, dann folgt

$$f(K) < \Delta \cdot |E_K| \quad \Leftrightarrow \quad f(K) + \Delta \cdot |E_K| < 0$$

Also ist in dem Kreis K eine der beiden Grenzen der Bedingung für Kreise verletzt und Δ somit keine Lösung für das Δ -Hierarchie-Problem.

2: Nun zeigen wir, daß Δ_{max}^G die Bedingungen für das Δ -Hierarchie-Problem erfüllt. Dazu wandeln wir mit Δ_{max}^G die Instanz in eine Eingabe für das Intervall-Hierarchie-Problem um und definieren eine neue Kantenbewertung $f^I : E \rightarrow I(\mathbb{R})$ wie folgt: $f^I(e) = [f(e) - \Delta_{max}^G, f(e) + \Delta_{max}^G]$. Dann erfüllt Δ_{max}^G die Bedingungen für das Δ -Hierarchie-Problem, wenn es eine Lösung für das Intervall-Hierarchie-Problem für die Instanz G, f^I, s gibt. Wir zeigen also, daß es eine Lösung für das Intervall-Hierarchie-Problem für diese Instanz G, f^I, s gibt. Das ist nach Lemma 5.4 genau dann der Fall, wenn für jedes Knotenpaar $u, v \in V$ der Schnitt der Längen aller Wege von u nach v in G' nicht leer ist, also $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f^{I'}(p) \neq \emptyset$, wobei G' der erweiterte Graph mit Kantenbewertung $f^{I'}$ ist. Da die Menge $P_{u \rightarrow v}^{G'}$ endlich ist, ist $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f^{I'}(p) = \emptyset$ genau dann, wenn es zwei Wege $p_1, p_2 \in P_{u \rightarrow v}^{G'}$ gibt mit $f^{I'}(p_1) \cap f^{I'}(p_2) = \emptyset$. Seien also $p_1, p_2 \in P_{u \rightarrow v}^{G'}$ Wege in G' . Zu jedem Weg $p = u_1, \dots, u_k$ in G' gibt es einen korrespondierenden Pfad $P = (u_1, \dots, u_k, E_P)$ in G , so daß für $1 \leq i < k$ gilt

$$f^{I'}((u_i, u_{i+1})) = [f((u_i, u_{i+1})) - \Delta_{max}^G, f((u_i, u_{i+1})) + \Delta_{max}^G]$$

oder

$$f^{I'}((u_i, u_{i+1})) = [-f((u_{i+1}, u_i)) - \Delta_{max}^G, -f((u_{i+1}, u_i)) + \Delta_{max}^G]$$

Sei also $P_1 = (p_1, E_{P_1})$ der zu p_1 und $P_2 = (p_2, E_{P_2})$ der zu p_2 korrespondierende Pfad in G . Es gilt

$$f^{I'}(p_1) = [f(P_1) - |E_{P_1}| \cdot \Delta_{max}^G, f(P_1) + |E_{P_1}| \cdot \Delta_{max}^G]$$

und

$$f^{I'}(p_2) = [f(P_2) - |E_{P_2}| \cdot \Delta_{max}^G, f(P_2) + |E_{P_2}| \cdot \Delta_{max}^G]$$

Also ist $f^{I'}(p_1) \cap f^{I'}(p_2) \neq \emptyset$, wenn

$$\max\{f(P_1) - |E_{P_1}| \cdot \Delta_{max}^G, f(P_2) - |E_{P_2}| \cdot \Delta_{max}^G\} \leq \min\{f(P_1) + |E_{P_1}| \cdot \Delta_{max}^G, f(P_2) + |E_{P_2}| \cdot \Delta_{max}^G\}$$

Da

$$f(P_1) - |E_{P_1}| \cdot \Delta_{max}^G \leq f(P_1) + |E_{P_1}| \cdot \Delta_{max}^G$$

und

$$f(P_2) - |E_{P_2}| \cdot \Delta_{max}^G \leq f(P_2) + |E_{P_2}| \cdot \Delta_{max}^G$$

bleibt zu zeigen, daß

$$f(P_1) - |E_{P_1}| \cdot \Delta_{max}^G \leq f(P_2) + |E_{P_2}| \cdot \Delta_{max}^G$$

und

$$f(P_2) - |E_{P_2}| \cdot \Delta_{max}^G \leq f(P_1) + |E_{P_1}| \cdot \Delta_{max}^G.$$

Wie oben kann man die beiden Pfade P_1 und P_2 zu einem Kreispfad K zusammensetzen, indem man die Kanten von P_2 in umgekehrter Richtung durchläuft. Für diesen Kreispfad K gilt nach Voraussetzung:

$$\frac{|f(K)|}{|E_{P_1} \cup E_{P_2}|} = \frac{|f(P_1) - f(P_2)|}{|E_{P_1} \cup E_{P_2}|} \leq \Delta_{max}^G$$

Angenommen $f(P_1) \geq f(P_2)$. Dann ist

$$f(P_1) - f(P_2) \leq |E_{P_1}| \cdot \Delta_{max}^G + |E_{P_2}| \cdot \Delta_{max}^G$$

also

$$f(P_1) - |E_{P_1}| \cdot \Delta_{max}^G \leq f(P_2) + |E_{P_2}| \cdot \Delta_{max}^G.$$

Und mit $f(P_1) \geq f(P_2)$ ist auch $f(P_2) - |E_{P_2}| \cdot \Delta_{max}^G \leq f(P_1) + |E_{P_1}| \cdot \Delta_{max}^G$, da $\Delta_{max}^G \geq 0$. Mit einer analogen Argumentation erhält man die beiden Beziehungen auch für $f(P_1) < f(P_2)$. Daraus folgt das jedes Knotenpaar $u, v \in V$ der Schnitt $\bigcap_{p \in P_{u \rightarrow v}^{G'}} f'(p)$ nicht leer ist. Damit hat das Intervall-Hierarchie-Problem für die Instanz G, f^I, s eine Lösung. Zusammen mit dem ersten Teil folgt daraus, das Δ_{max}^G Lösung für das Δ -Hierarchie-Problem ist. \square

Das Δ -Hierarchie-Problem läßt sich also in folgendes Problem umwandeln:

Gegeben ist ein gerichteter Graph $G = (V, E)$ mit Kantenbewertung $f : E \rightarrow \mathbb{R}$

Gesucht wird $\Delta_{max}^G = \max\{ \frac{|f(K)|}{|E_K|} \mid K = (u_1, \dots, u_k, E_K) \text{ ist Kreispfad in } G \}$.

Dies ist ähnlich zu Karps minimalen mittleren Kreislänge (minimum mean-weight cycle) [Kar78].

Definition 5.10 Problem: Minimale mittlere Kreislänge

Gegeben sei ein gerichteter Graph $G = (V, E)$ mit Kantenbewertung $f : E \rightarrow \mathbb{R}$

Gesucht wird $\Delta_{min}^G = \min\{ \frac{f((u_k, u_1)) + \sum_{i=1}^{k-1} f((u_i, u_{i+1}))}{k} \mid K = u_1, \dots, u_k \text{ ist Kreis in } G \}$.

Karp gibt einen Algorithmus an, der das minimale mittlere Kreisgewicht in $O(|V| \cdot |E|)$ berechnet [Kar78]. Dabei werden nur Kreise, aber keine Kreispfade betrachtet.

Da zu einem Kreispfad $K = (u_1, \dots, u_k, E_K)$ auch $K_u = (u_k, \dots, u_1, E_K)$ ein Kreispfad (mit umgekehrter Laufrichtung) ist und $f(K) = -f(K_u)$ gilt, folgt

$$\min\left\{ \frac{|f(K)|}{|E_K|} \mid K = (u_1, \dots, u_k, E_K) \text{ ist Kreispfad in } G \right\} = -\Delta_{max}^G.$$

Um das Δ -Hierarchie-Problem zu lösen, konstruieren wir zu einer Instanz G, f, s einen erweiterten Graphen G' mit Kantenbewertung f' , so daß $\Delta_{min}^{G'} = -\Delta_{max}^G$.

Da es zu jedem Kreispfad in $G = (V, E)$ einen Kreis in $G' = (V, E')$ geben muß, werden wieder die zusätzlichen Kanten in inverser Richtung eingefügt, also $E' = \{(u, v) \mid (u, v) \in E \wedge (v, u) \in E\}$. Da wir kürzeste Wege suchen, möchten wir für Kanten $(u, v), (v, u) \in E$ mit $f((u, v)) \neq -f((v, u))$ für jede Richtung die kleinere der beiden Kantenbewertungen für f' übernehmen. Ist zum Beispiel $f((u, v)) < -f((v, u))$, also $f((v, u)) < -f((u, v))$, so sollte $f'((u, v)) = f((u, v))$ und $f'((v, u)) = f((v, u))$ sein. Deshalb definieren wir die Kantenbewertungsfunktion $f' : E' \rightarrow \mathbb{R}$ als

$$f'((u, v)) = \begin{cases} \min\{f((u, v)), -f((v, u))\}, & \text{falls } (u, v), (v, u) \in E \\ f((u, v)) & \text{falls } (u, v) \in E, (v, u) \notin E \\ -f((v, u)) & \text{falls } (v, u) \in E, (u, v) \notin E \end{cases}$$

Das garantiert, daß in dem betrachteten Graphen G' alle Wege die nur aus einer Kante $(u, v) \in E'$ bestehen so lang sind, wie der kürzere Pfad $(u, v, \{e\})$ in G , der nur über eine Kante $e \in E$ läuft. Siehe Abbildung 5.11 für ein Beispiel eines Graphen G mit Kantenbewertung f und den zugehörigen erweiterten Graphen G' mit Kantenbewertung f' .

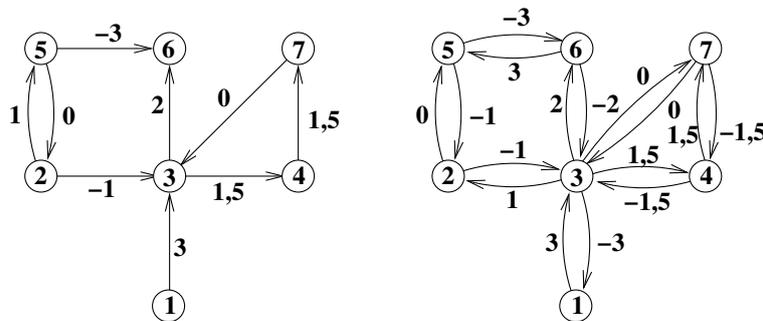


Abbildung 5.11: Links ein gerichteter Graph G mit Kantenbewertung f für einen gerichteten Graph G mit 7 Knoten und rechts der zugehörige erweiterte Graph G' mit Kantenbewertung f' . In diesem Fall ist $\Delta_G = 1$.

Der Algorithmus zur Berechnung von Δ_G für eine Instanz G, f, s gliedert sich in drei Teile:

1. Erstellen von G' und Bewertungsfunktion f'
2. Bestimmung von $\Delta_{min}^{G'}$ mit Karps Algorithmus [Kar78]
3. $\Delta_G = |\Delta_{min}^{G'}|$

1. Erstellen von G' und Bewertungsfunktion f'

für alle Knoten $v \in V$ setze Markierung $m(v) =$ nicht definiert

für alle Knoten $v \in V$ {

 für alle in v einlaufenden Kanten (u, v)

 markiere den Knoten u mit $m(u) = -f((u, v))$

 für alle aus v auslaufenden Kanten (v, w)

 wenn $m(w)$ nicht definiert dann füge (w, v) mit $f((w, v)) = -f((v, w))$ in E ein

 sonst wenn $m(u) < f((v, w))$ dann $f(v, w) = m(u)$

 für alle in v einlaufenden Kanten (u, v)

 setze $m(u) =$ nicht definiert

}

Die Laufzeit von Teil 1 ist in $O(|V| + |E|)$ und von Teil 2 (Karps Algorithmus) in $O(|V| \cdot |E|)$. Die Gesamtlaufzeit des Verfahrens ist somit in $O(|V| \cdot |E|)$.

5.2.4 Das Intervall- Δ -Hierarchie-Problem

Da es auch für das Intervall-Hierarchie-Problem nicht immer eine Lösung gibt, ist es sinnvoll, auch hier eine Optimierungsversion zu betrachten. Sei wieder $og : I(\mathbb{R}) \rightarrow \mathbb{R}$ definiert durch $og([a, b]) := b$ und $ug : I(\mathbb{R}) \rightarrow \mathbb{R}$ definiert durch $ug([a, b]) := a$. Analog zur Definition des Δ -Hierarchie-Problems erhalten wir:

Definition 5.11 *Intervall- Δ -Hierarchie-Problem*

Gegeben sei ein gerichteter Graph $G = (V, E)$ mit $f : E \rightarrow I(\mathbb{R})$ und ein Knoten $s \in V$.

Gesucht wird ein minimales $\Delta_G \in \mathbb{R}_0^+$, so daß es eine Funktion $l : V \rightarrow \mathbb{R}$ gibt, mit $l(s) = 0$ und $ug(f((u, v))) - \Delta_G \leq l(v) - l(u) \leq og(f((u, v))) + \Delta_G$, oder anders geschrieben $l(v) - l(u) \in f((u, v)) + [-\Delta_G, +\Delta_G]$ für alle Kanten $(u, v) \in E$.

Abbildung 5.12 zeigt ein Beispiel für eine Instanz für das Intervall- Δ -Hierarchie-Problem

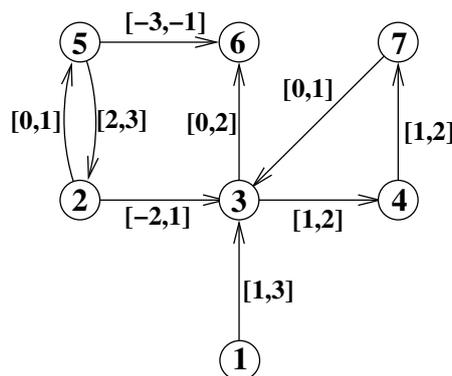


Abbildung 5.12: Ein gerichteter Graph G mit Kantenbewertung f für einen gerichteten Graph G mit 7 Knoten. Egal welchen Knoten $s \in V$ man als Startknoten wählt, dieser Graph hat keine Lösung für das Intervall-Hierarchie-Problem.

Auch dieses Problem hat immer eine Lösung. Es gilt:

Lemma 5.12 Sei G, f, s eine Instanz für das Intervall- Δ -Hierarchie-Problem und Δ_G eine Lösung. Dann ist Δ_G maximal so groß wie das Maximum über die Abstände zu 0 der betragsmäßig kleineren Intervallgrenzen. Es gilt also

$$\Delta_G \leq \max\{ \min\{|og(e)|, |ug(e)|\} \mid e \in E \}.$$

Beweis: Analog zu Lemma 5.7 □

Nach der Berechnung von Δ_G kann die Instanz G, f, s in eine Instanz für das Intervall-Hierarchie-Problem umgewandelt werden, um eine Hierarchie l zu berechnen. Dazu wird eine neue Kantenbewertung $f^I : E \rightarrow I(\mathbb{R})$ definiert durch $f^I(e) = f(e) + [-\Delta_G, +\Delta_G]$. Dann ist das Intervall-Hierarchie-Problem für die Instanz G, f^I, s lösbar. Ist die Lösung für das Intervall- Δ -Hierarchie-Problem $\Delta_G = \max\{ \min\{|og(e)|, |ug(e)|\} \mid e \in E \}$, so ist $l \equiv 0$ eine Lösung für das zugehörige Intervall-Hierarchie-Problem.

Um das Intervall- Δ -Hierarchie-Problem näher zu untersuchen, werden wir wieder Pfade verwenden. Auch hier ist die Länge eines Pfades $P = (u_1, \dots, u_k, E_P)$ in einem kantenbewerteten Graphen die Summe der Kostenwerte (Längen) aller Vorwärtskanten plus die Summe inversen der Kostenwerte (Längen) aller Rückwärtskanten, also

$$f(P) = \sum_{e \in VK_P} f(e) + \sum_{e \in RK_P} inv(f(e))$$

mit der Addition für Intervalle und wie zuvor $inv : I(\mathbb{R}) \rightarrow I(\mathbb{R})$, $inv([a, b]) := [-b, -a]$. Für einen Kreispfad $K = (u_1, \dots, u_k, E_K)$ mit Kantenmenge $E_K = \{e_1, \dots, e_k\} \subseteq E$ ist die Länge also $f(K) = \sum_{e \in VK_K} f(e) + \sum_{e \in RK_K} inv(f(e))$.

Nach Lemma 5.4 hat das Intervall-Hierarchie-Problem genau dann eine Lösung, wenn die Länge jedes Kreises die 0 enthält. Somit muß für eine gültige Lösung Δ_G des Intervall- Δ -Hierarchie-Problem jeder Kreispfad K in G mit der eben definierten Kantenbewertung f^I , eine Länge haben, die 0 enthält, also $0 \in f^I(K)$.

Die für die Berechnung von Δ_G interessanten Kreispfade sind also die Kreispfade K , deren Länge mit der Kantenbewertung f die 0 nicht enthält, also $0 \notin f(K)$. Das sind genau die Kreispfade K , bei denen die Untergrenzen der Länge größer als 0 ist, also $ug(f(K)) > 0$, (dann ist auch die Obergrenzen der Länge größer als 0) beziehungsweise die, bei denen die Obergrenzen der Länge (und damit auch die Untergrenzen der Länge) kleiner als 0 ist, also $og(f(k)) < 0$.

Zu einem Kreispfad $K = (u_1, \dots, u_k, E_K)$ sei $K_u = (u_k, \dots, u_1, E_K)$ der Kreispfad mit umgekehrter Laufrichtung. Dann gilt $f(K) = inv(f(K_u))$. Also ist $og(f(K)) = -ug(inv(f(K_u)))$. Analog zu 5.9 erhalten wir folgendes Theorem:

Satz 5.13 Sei $G = (V, E)$, f, s eine Instanz für das Intervall- Δ -Hierarchie-Problem und

$$\Delta_{max}^G = \max\{ \frac{|\gamma|}{|E_K|} \mid K = (u_1, \dots, u_k, E_K) \text{ ist Kreispfad in } G \text{ mit } \gamma = og(f(K)) < 0 \}$$

Dann ist Δ_{max}^G Lösung für das Δ -Hierarchie-Problem.

Beweis: Der Beweis ist analog zu dem Beweis von 5.9. □

Damit folgt direkt aus den Vorbemerkungen:

Korollar 5.14 Sei $G = (V, E)$, f, s eine Instanz für das Intervall- Δ -Hierarchie-Problem und Δ_{max}^G Lösung. Dann ist

$$\Delta_{max}^G = \max\left\{ \frac{|\gamma|}{|E_K|} \mid K = (u_1, \dots, u_k, E_K) \text{ ist Kreispfad in } G \text{ mit } \gamma = ug(g(K)) > 0 \right\}.$$

Aus Satz 5.13 kann man entnehmen, daß man auch das Intervall- Δ -Hierarchie-Problem auf das Problem der minimalen mittleren Kreislänge zurückführen kann. Um das Problem wieder in eine Instanz für das minimale mittlere Kreislänge umzuformen, ist es einfacher, mit den Obergrenzen als mit den Untergrenzen der Intervalle zu arbeiten. Um das Intervall- Δ -Hierarchie-Problem zu lösen, konstruieren wir also zu einer Instanz G, f, s einen Graphen G' mit Kantenbewertung f' , so daß $\Delta_{min}^{G'} = -\Delta_{max}^G$, wobei $\Delta_{min}^{G'}$ wieder die minimale mittlere Kreislänge für die Instanz G', f' ist.

Da es zu jeden Kreispfad in $G = (V, E)$ einen Kreis in $G' = (V, E')$ geben soll, werden wieder die zusätzlichen Kanten in inverser Richtung eingefügt, also $E' = \{(u, v) \mid (u, v) \in E \wedge (v, u) \in E\}$. Als Kantenbewertungen verwenden wir für $(u, v) \in E$ die Obergrenze des Intervalls der $f((u, v))$ und für $(u, v) \notin E$ die Obergrenze des Intervalls $inv(f((v, u)))$. Für Kanten $(u, v), (v, u) \in E$ mit $og(f((u, v))) \neq og(inv(f((v, u))))$ werden wir für jede Richtung die kleinere der beiden Obergrenzen der Kantenbewertungen f für f' übernehmen. Dieses Vorgehen beruht darauf, daß für eine Lösung Δ_G gelten muß

$$l(v) - l(u) \in f((u, v)) + [-\Delta_G, +\Delta_G]$$

sowie auch

$$l(v) - l(u) \in inv(f((v, u))) + [-\Delta_G, +\Delta_G]$$

und somit

$$l(v) - l(u) \in (f((u, v)) + [-\Delta_G, +\Delta_G]) \cap inv(f((v, u))) + [-\Delta_G, +\Delta_G].$$

Dieser Schnitt hat die obere Intervallgrenze $\min\{og(f((u, v))), og(inv(f((v, u))))\} + \Delta_G$. Ist zum Beispiel $og(f((u, v))) < og(inv(f((v, u))))$, also $og(f((v, u))) < og(inv(f((u, v))))$, so sollte $f'((u, v)) = og(f((u, v)))$ und $f'((v, u)) = og(f((v, u)))$ sein. Deshalb definieren wir die Kantenbewertungsfunktion $f' : E' \rightarrow \mathbb{R}$ als

$$f'((u, v)) = \begin{cases} \min\{og(f((u, v))), og(inv(f((v, u))))\}, & \text{falls } (u, v), (v, u) \in E \\ og(f((u, v))) & \text{falls } (u, v) \in E, (v, u) \notin E \\ og(inv(f((v, u)))) & \text{falls } (v, u) \in E, (u, v) \notin E \end{cases}$$

Siehe Abbildung 5.13 für ein Beispiel eines Graphen G mit Kantenbewertung f und den zugehörigen erweiterten Graphen G' mit Kantenbewertung f' .

Damit haben wir also eine Funktion f' mit reellen Kantenbewertungen und G', f' ist eine Instanz, für die minimale mittlere Kreislänge berechnet werden kann. Ergibt sich bei der Berechnung ein $\Delta_{min} > 0$, so haben alle kürzesten Kreise auf den Obergrenzen ein positives Gewicht. Daraus folgt dann, daß alle Kreise auf den Untergrenzen ein negatives Gewicht haben. Dann ist 0 in dem Längenintervall von jedem Kreis enthalten und somit $\Delta_G = 0$. Ist hingegen $\Delta_{min} \leq 0$, so gilt $\Delta_G = |\Delta_{min}|$

Der folgende Algorithmus erstellt aus dem Graphen G den Hilfsgraphen G' , und ermittelt eine Kantenbewertung $f' : E' \rightarrow \mathbb{R}$, so daß der resultierende Graph $G' = (V, E')$ mit f' eine

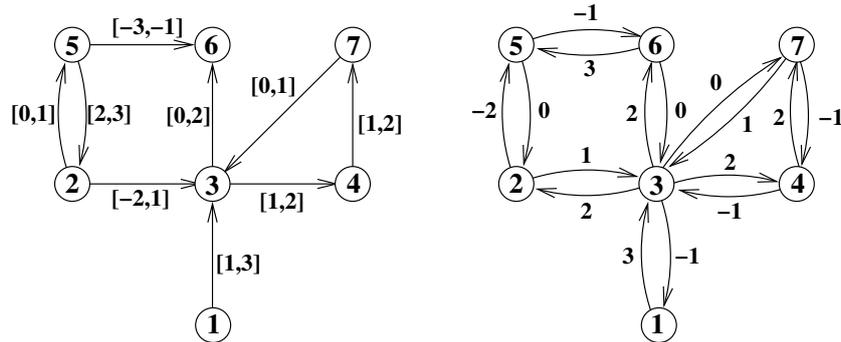


Abbildung 5.13: Links ein gerichteter Graph G mit 7 Knoten und Kantenbewertung f und rechts der zugehörige erweiterte Graph G' mit Kantenbewertung f' . In diesem Fall ist $\Delta_G = 1$.

Eingabe für minimale mittlere Kreisgewicht bildet. Anschließend kann auf dem Graph G' Karps Algorithmus zur Berechnung von Δ_{min} durchgeführt werden. Treten negative Kreise auf, ist also $\Delta_{min} < 0$, so wird $\Delta_G = |\Delta_{min}|$ gesetzt. Andernfalls ist $\Delta_G = 0$.

Der Algorithmus zur Berechnung von Δ gliedert sich also wieder in drei Teile:

1. Erstellen von G' mit Kantenbewertung f'
2. Bestimmung von Δ_{min} mit Karps Algorithmus [Kar78]
3. if $\Delta_{min} \leq 0$ then $\Delta_G = |\Delta_{min}|$ else $\Delta_G = 0$

Erstellen von G' mit Kantenbewertung f'

für alle Knoten $v \in V$ setze Markierung $m(v) =$ nicht definiert

für alle Knoten $v \in V$ {

für alle in v einlaufenden Kanten (u, v)

markiere den Knoten u mit $m(u) = -ug(f((u, v)))$

für alle aus v auslaufenden Kanten (v, w)

wenn $m(w)$ nicht definiert dann {

füge (w, v) in E ein

$f'((w, v)) = -ug(f((v, w)))$

}

sonst wenn $m(u) < og(f((v, w)))$ dann $f'(v, w) = m(u)$

sonst $f'((v, w)) = og(f((v, w)))$

für alle in v einlaufenden Kanten (u, v)

setze $m(u) =$ nicht definiert

}

Die Laufzeit von Teil 1 des Verfahrens ist in $O(|V| + |E|)$ und damit liegt die Gesamtlaufzeit wieder in $O(|V| \cdot |E|)$.

5.3 Lösen der Hierarchie-Probleme mit Hilfe linearer Optimierung

In diesem Abschnitt soll gezeigt werden, wie man die oben definierten Hierarchieprobleme mit Hilfe von linearer Optimierung lösen kann. Dabei handelt es sich um ein weit verbreitetes Ansatz, um eine optimale Lösung zu einem Problem zu finden, vorausgesetzt es läßt sich durch eine lineare Funktion sowie ebenfalls lineare Nebenbedingungen darstellen. Eine detaillierte Einführung in das Thema findet man zum Beispiel in dem Buch von Papadimitriou und Steiglitz [PS98].

5.3.1 Lineare Optimierung

Wie der Name schon sagt, möchte man eine „beste“ Lösung für ein gegebenes (lineares) Problem finden. Eine Instanz für das lineare Optimierungsproblem besteht aus einer linearen Funktion, die maximiert oder minimiert werden soll, Nebenbedingungen aus linearen Gleichungen oder Ungleichungen sowie Grenzen, die die Variablen der Zielfunktion einschränken. Konkret sieht ein solches Optimierungsproblem dann so aus:

Maximiere / Minimiere den Ausdruck

$$c_1x_1 + c_2x_2 + \dots + c_nx_n$$

unter den Nebenbedingungen

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &\leq b_m \end{aligned}$$

und mit den Grenzen

$$\begin{aligned} l_1 &\leq x_1 \leq u_1 \\ l_2 &\leq x_2 \leq u_2 \\ &\vdots \\ l_n &\leq x_n \leq u_n \end{aligned}$$

Dabei sind die x_1, x_2, \dots, x_n Variablen, für die Werte gesucht werden, und die anderen Elemente Konstanten. Diese können rationale Zahlen sein, wobei die l_1, l_2, \dots, l_n auch den Wert $-\infty$ und die u_1, u_2, \dots, u_n auch den Wert ∞ annehmen können.

Man kann ein Problem für die lineare Optimierung auch durch Vektoren und Matrizen beschreiben. Dazu sei

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$$

und

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} \quad l = \begin{pmatrix} l_1 \\ l_2 \\ \vdots \\ l_n \end{pmatrix} \quad u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

Dann kann ein lineares Problem formuliert werden durch:
Maximiere/Minimiere den Ausdruck

$$c^T x$$

unter den Nebenbedingungen

$$Ax \leq b$$

und mit den Grenzen

$$l \leq x \leq u,$$

wobei c^T der Transponierte Vektor zu c ist.

Es gibt verschiedene Verfahren, um diese Art von Problemen zu lösen, am ältesten und am weitesten verbreitet ist das Simplex-Verfahren, das 1947 von Dantzig entwickelt wurde (siehe zum Beispiel [Dan63]). Es hat zwar im schlimmsten Fall eine exponentielle Laufzeit, diese tritt aber in der Praxis zumeist nicht auf. Heute wird auch häufig das Innere-Punkte-Verfahren von Karmarkar [Kar84] verwendet, daß eine polynomielle Laufzeit besitzt.

5.3.2 Hierarchieberechnung mit linearer Optimierung

Wie aber bekommen wir aus einer Instanz für die Δ -Hierarchieprobleme eine Eingabe für ein lineares Optimierungsproblem? Betrachten wir dazu noch einmal die beiden Beispiele aus dem letzten Abschnitt, zuerst für einen Graphen mit einer reellen Kantenbewertung (siehe Abbildung 5.14).

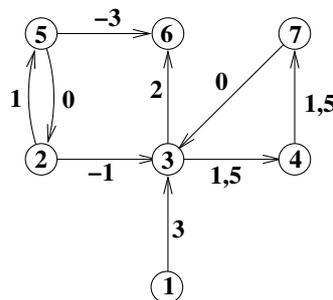


Abbildung 5.14: Ein gerichteter Graph G mit Kantenbewertung $f_1 : V \rightarrow \mathbb{R}$.

Wir haben also gegeben einen Graphen $G = (\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}, E)$ mit

$$E = \{(v_1, v_3), (v_2, v_3), (v_2, v_5), (v_3, v_4), (v_3, v_6), (v_4, v_7), (v_5, v_2), (v_5, v_6), (v_7, v_3)\}$$

und eine Kantenbewertung

$$\begin{aligned}
 f_1((v_1, v_3)) &= 3, \\
 f_1((v_2, v_3)) &= -1, \\
 f_1((v_2, v_5)) &= 1, \\
 f_1((v_3, v_4)) &= 1, 5, \\
 f_1((v_3, v_6)) &= 2, \\
 f_1((v_4, v_7)) &= 1, 5, \\
 f_1((v_5, v_2)) &= 0, \\
 f_1((v_5, v_6)) &= -3, \\
 f_1((v_7, v_3)) &= 0.
 \end{aligned}$$

Im Δ -Hierarchieproblem wird ein minimales $\Delta_G \in \mathbb{R}_0^+$ gesucht, so daß es eine Funktion $l : V \rightarrow \mathbb{R}$ gibt, mit $l(s) = 0$, für ein gegebenes $s \in v$ und $f((u, v)) - \Delta_G \leq l(v) - l(u) \leq f((u, v)) + \Delta_G$ für alle $(u, v) \in E$.

Damit erhalten wir für jede Kante $(u, v) \in E$ die beiden Bedingungen

$$l(v) - l(u) - \Delta_G \leq f((u, v)) \text{ und } l(v) - l(u) + \Delta_G \geq f((u, v)).$$

Dies sieht schon sehr wie die Form einer Nebenbedingung in einem linearen Programm aus. Um diese Bedingung in einen linearen Problem zu verwenden, müssen noch $l(v)$ und $l(u)$ als Variablen aufgefaßt werden. Wir verwenden deshalb im Programm für jeden Knoten $v \in V$ ein Variable v , die für den Funktionswert $l(v)$ steht. Damit sehen die Bedingungen folgendermaßen aus

$$v - u - \Delta_G \leq f((u, v)) \text{ und } v - u + \Delta_G \geq f((u, v)).$$

Im folgenden wird Δ_G mit Δ_{max} bezeichnet, weil es die maximale Abweichung darstellt, die eine Kantenbewertung $f((u, v))$ von der durch die Hierarchie bestimmten Kantenbewertungen $l(v) - l(u)$ hat. Das Programm um ein optimales Δ_{max} für den Beispielgraph zu bestimmen (und gleichzeitig eine Hierarchie zu berechnen) sehe also für $s = v_1$ (im für lineare Programme verbreiteten LP-Format) folgendermaßen aus

Problem

Beispiel

Minimize

obj: Δ_{max}

Subject

$$\begin{aligned}
 \text{c001: } v_3 - v_1 - \Delta_{max} &\leq 3 \\
 \text{c002: } v_3 - v_1 + \Delta_{max} &\geq 3 \\
 \text{c003: } v_3 - v_2 - \Delta_{max} &\leq -1 \\
 \text{c004: } v_3 - v_2 + \Delta_{max} &\geq -1 \\
 \text{c005: } v_5 - v_2 - \Delta_{max} &\leq 1 \\
 \text{c006: } v_5 - v_2 + \Delta_{max} &\geq 1 \\
 \text{c007: } v_4 - v_3 - \Delta_{max} &\leq 1, 5 \\
 \text{c008: } v_4 - v_3 + \Delta_{max} &\geq 1, 5 \\
 \text{c009: } v_6 - v_3 - \Delta_{max} &\leq 2 \\
 \text{c010: } v_6 - v_3 + \Delta_{max} &\geq 2 \\
 \text{c011: } v_7 - v_4 - \Delta_{max} &\leq 1, 5
 \end{aligned}$$

- c012: $v_7 - v_4 + \Delta_{max} \geq 1,5$
- c013: $v_2 - v_5 - \Delta_{max} \leq 0$
- c014: $v_2 - v_5 + \Delta_{max} \geq 0$
- c015: $v_6 - v_5 - \Delta_{max} \leq -3$
- c016: $v_6 - v_5 + \Delta_{max} \geq -3$
- c017: $v_3 - v_7 - \Delta_{max} \leq 0$
- c018: $v_3 - v_7 - \Delta_{max} \leq 0$
- c019: $v_1 = 0$

End

Die Variablen v_1 bis v_7 sind nicht in der Optimierungszeile enthalten. Man könnte sie aber jeweils mit dem Faktor 0 ergänzen. Wichtig ist, daß sie nicht optimiert werden, sondern abhängig vom optimalen Δ_{max} beliebig große Werte annehmen können. Ihr Wertebereich wird von der Optimierungsbedingung nicht direkt eingeschränkt.

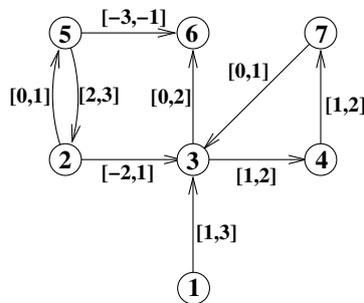


Abbildung 5.15: Ein gerichteter Graph G mit Kantenbewertung $f_2 : V \rightarrow I(\mathbb{R})$.

Ähnlich läßt sich auch für das Intervall- Δ -Hierarchieproblem ein lineares Programm erstellen. Wir betrachten dazu den Graphen G mit Intervall-Kantenbewertung f_2 in Abbildung 5.15. Wir haben wieder $G = (\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}, E)$ mit

$$E = \{(v_1, v_3), (v_2, v_3), (v_2, v_5), (v_3, v_4), (v_3, v_6), (v_4, v_7), (v_5, v_2), (v_5, v_6), (v_7, v_3)\}$$

und diesmal

$$\begin{aligned} f_2((v_1, v_3)) &= [1, 3], \\ f_2((v_2, v_3)) &= [-2, 1], \\ f_2((v_2, v_5)) &= [0, 1], \\ f_2((v_3, v_4)) &= [1, 2], \\ f_2((v_3, v_6)) &= [0, 2], \\ f_2((v_4, v_7)) &= [1, 2], \\ f_2((v_5, v_2)) &= [2, 3], \\ f_2((v_5, v_6)) &= [-3, -1], \\ f_2((v_7, v_3)) &= [0, 1]. \end{aligned}$$

Gesucht wird wieder ein minimales $\Delta_G \in \mathbb{R}_0^+$, so daß es eine Funktion $l : V \rightarrow \mathbb{R}$ gibt, mit $l(s) = 0$ für ein gegebenes $s \in V$ und $ug(f((u, v))) - \Delta_G \leq l(v) - l(u) \leq og(f((u, v))) + \Delta_G$ oder anders geschrieben $l(v) - l(u) \in f((u, v)) + [-\Delta_G, +\Delta_G]$.

Daraus erhalten wir die beiden Bedingungen

$$l(v) - l(u) - \Delta_G \leq og(f((u, v))) \quad \text{und} \quad l(v) - l(u) + \Delta_G \geq ug(f((u, v))).$$

Ersetzen wir $l(v)$ und $l(u)$ wieder durch Variablen v und u so ergibt das die Bedingungen

$$v - u - \Delta_G \leq og(f((u, v))) \quad \text{und} \quad v - u + \Delta_G \geq ug(f((u, v))).$$

Damit können wir wieder ein lineares Programm für das Beispiel erstellen, mit dem man ein optimales Δ_{max} für $s = v_1$ berechnen kann.

```

Problem
  Beispiel 2
Minimize
  obj:  $\Delta_{max}$ 
Subject
  c001:  $v_3 - v_1 - \Delta_{max} \leq 3$ 
  c002:  $v_3 - v_1 + \Delta_{max} \geq 1$ 
  c003:  $v_3 - v_2 - \Delta_{max} \leq 1$ 
  c004:  $v_3 - v_2 + \Delta_{max} \geq -2$ 
  c005:  $v_5 - v_2 - \Delta_{max} \leq 1$ 
  c006:  $v_5 - v_2 + \Delta_{max} \geq 0$ 
  c007:  $v_4 - v_3 - \Delta_{max} \leq 2$ 
  c008:  $v_4 - v_3 + \Delta_{max} \geq 1$ 
  c009:  $v_6 - v_3 - \Delta_{max} \leq 2$ 
  c010:  $v_6 - v_3 + \Delta_{max} \geq 0$ 
  c011:  $v_7 - v_4 - \Delta_{max} \leq 2$ 
  c012:  $v_7 - v_4 + \Delta_{max} \geq 1$ 
  c013:  $v_2 - v_5 - \Delta_{max} \leq 1$ 
  c014:  $v_2 - v_5 + \Delta_{max} \geq 0$ 
  c015:  $v_6 - v_5 - \Delta_{max} \leq -1$ 
  c016:  $v_6 - v_5 + \Delta_{max} \geq -3$ 
  c017:  $v_3 - v_7 - \Delta_{max} \leq 1$ 
  c018:  $v_3 - v_7 - \Delta_{max} \leq 0$ 
  c019:  $v_1 = 0$ 
End
    
```

Die lineare Optimierung bietet auch die Möglichkeit, die Summe aller Abweichung leicht zu minimieren. Dazu muß man in dem obigen Programm nur für jede Kante eine Variable einführen, die speziell die Abweichung der Hierarchie von dieser Kante mißt, sowie eine weitere Variable Summe, die der Summe aller einzelnen Abweichung entspricht. Konkret sieht das dann so aus.

```

Problem
  Beispiel 3
Minimize
  obj: summe
Subject
  c001:  $v_3 - v_1 - \Delta_1 \leq 3$ 
  c002:  $v_3 - v_1 + \Delta_1 \geq 1$ 
    
```

```

c003:  $v_3 - v_2 - \Delta_2 \leq 1$ 
c004:  $v_3 - v_2 + \Delta_2 \geq -2$ 
c005:  $v_5 - v_2 - \Delta_3 \leq 1$ 
c006:  $v_5 - v_2 + \Delta_3 \geq 0$ 
c007:  $v_4 - v_3 - \Delta_4 \leq 2$ 
c008:  $v_4 - v_3 + \Delta_4 \geq 1$ 
c009:  $v_6 - v_3 - \Delta_5 \leq 2$ 
c010:  $v_6 - v_3 + \Delta_5 \geq 0$ 
c011:  $v_7 - v_4 - \Delta_6 \leq 2$ 
c012:  $v_7 - v_4 + \Delta_6 \geq 1$ 
c013:  $v_2 - v_5 - \Delta_7 \leq 1$ 
c014:  $v_2 - v_5 + \Delta_7 \geq 0$ 
c015:  $v_6 - v_5 - \Delta_8 \leq -1$ 
c016:  $v_6 - v_5 + \Delta_8 \geq -3$ 
c017:  $v_3 - v_7 - \Delta_9 \leq 1$ 
c018:  $v_3 - v_7 - \Delta_9 \leq 0$ 
c019:  $v_1 = 0$ 
c020:  $-\text{summe} + \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4 + \Delta_5 + \Delta_6 + \Delta_7 + \Delta_8 + \Delta_9 = 0$ 
End

```

Mit diesem Programm wird also nicht die größte Abweichung, sondern die Summe aller Abweichungen minimiert. Dies ist auch unter dem Aspekt sinnvoll, daß bei allen Verbindungen, bei denen die Abweichung kleiner als Δ_{max} ist, die auftretende Abweichung nicht weiter minimiert wird. Bei der Summe hingegen werden alle Abweichungen gemeinsam minimiert. Allerdings schließt das nicht aus, daß die größte Abweichung sehr groß ist. Man kann auch beide Kriterien gleichzeitig optimieren. Dazu muß jede Bedingung zweimal angeführt werden, einmal für das indizierte Δ_i als Teil der Summe und einmal für das globale Δ_{max} . Für das obige Programm erhalten wir damit:

```

Problem
  Beispiel 4
Minimize
  obj:  $\text{summe} + \Delta_{max}$ 
Subject
  c001:  $v_3 - v_1 - \Delta_1 \leq 3$ 
  c002:  $v_3 - v_1 - \Delta_{max} \leq 3$ 
  c003:  $v_3 - v_1 + \Delta_1 \geq 1$ 
  c004:  $v_3 - v_1 + \Delta_{max} \geq 1$ 
  :
  c033:  $v_3 - v_7 - \Delta_9 \leq 1$ 
  c034:  $v_3 - v_7 - \Delta_{max} \leq 1$ 
  c035:  $v_3 - v_7 - \Delta_9 \leq 0$ 
  c036:  $v_3 - v_7 - \Delta_{max} \leq 0$ 
  c037:  $v_1 = 0$ 
  c038:  $-\text{summe} + \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4 + \Delta_5 + \Delta_6 + \Delta_7 + \Delta_8 + \Delta_9 = 0$ 
End

```

Diese Version führt dazu, daß die größte Abweichung doppelt gezählt wird. Man kann aber durch eine entsprechende Gewichtung einen noch höheren Wert auf die Minimierung der größten Abweichung legen, also zum Beispiel durch die Angabe

Minimize
obj: summe + 10000 · Δ_{max}

Aufgrund der auftretenden Größenordnungen bewirkt dieses, daß die Summe den Gesamtwert für die zu minimierende Funktion kaum ändert. Damit muß, um die Bedingung zu erfüllen, ein minimales Δ_{max} berechnet werden. Es wird also erreicht, das für dieses minimale Δ_{max} eine Lösung gewählt wird, für die auch die Summe minimal wird. Andersherum ist eine stärkere Gewichtung der Summe kaum sinnvoll, da ihr Wert sowieso schon deutlich größer ist, als der von Δ_{max} .

5.4 Berechnung von Hierarchien im visuellen System

Es sollen natürlich auch Hierarchien für das visuelle System berechnet werden. Hier werden in diesem Zusammenhang Berechnungen für zwei Datensätze unter verschiedenen Randbedingungen durchgeführt. Zum einen soll untersucht werden, was für Resultate die Methode auf dem kleinen Datensatz von Barone et al. [BBKK00], für den exakte SLN%-Werte und damit eine reelle Kantenbewertung vorliegen, erbringt. Zum anderen wird die vorgestellte Methode zum Berechnen von Hierarchien in Graphen mit Intervallbewertungen der Kanten auf dem Datensatz von Felleman und Van Essen [FE91] durchgeführt werden, da dieses nach wie vor der umfassendste Datensatz für das visuelle System ist. Dafür wird die dreistufige Skala von Felleman und Van Essen zur Klassifizierung der Verbindung auf eine fünfstufige Skala verfeinert. Zum Schluß wird auch noch eine Kantenbewertung analog zu der von Felleman und Van Essen verwendet.

Zur Berechnung der Hierarchien wurde das Programm für lineare Optimierung qsopt (siehe <http://www2.isye.gatech.edu/~wcook/qsopt/index.html>) verwendet.

5.4.1 Verwendung des SLN%-Wertes als Abstandsmaß

Der erste in diesem Kapitel vorgestellte Ansatz war, mit Hilfe einer reellen Kantenbewertung Hierarchiestufen zu bestimmen. Die SLN%-Werte der Arbeitsgruppe um Kennedy ([BBKK00], [BBK⁺02], [VFJ⁺04]) liefern eine solche Kantenbewertung (siehe dazu auch Abschnitt 5.1.3). Der SLN%-Wert ist eine reelle Zahl zwischen 0 und 100, die angibt, wie groß der supragranuläre Anteil der durch den retrograden Tracer gefärbten Neuronen ist. Dabei entsprechen laut den Autoren Werte unter 40% absteigenden, Werte zwischen 40% und 60% lateralen, Werte über 60% aufsteigenden Verbindungen. Wir wollen aus den SLN% Werten jetzt eine reelle Kantenbewertung konstruieren, bei der Werte kleiner 0 für absteigende Verbindungen, Werte gleich 0 für laterale Verbindungen und Werte größer 0 für aufsteigende Verbindungen stehen. Dazu legen wir den SLN%-Wert, der für eine laterale Beziehung steht, auf genau 50% fest.

Damit berechnen wir die Kantengewichte f für die von Barone et al. [BBKK00] bestimmten SLN%-Werte nach der folgenden Formel:

$$f((A_1, A_2)) = \frac{\text{SLN}\%((A_1, A_2)) - 50}{100} \cdot 2$$

wobei (A_1, A_2) die Verbindung von Areal A_1 nach Areal A_2 und $\text{SLN}\%((A_1, A_2))$ der dieser Verbindung zugeordnete SLN%-Wert ist. So erhalten wir normalisierte Kantengewichte in dem Intervall $[-1, 1] \in \mathbb{R}$. Verbindungen, die vorher einen SLN% Wert von mehr als 50% hatten, erhalten so einen Wert im Intervall $(0, 1]$; Verbindungen hingegen, die vorher einen SLN% von weniger als 50% hatten, erhalten Werte im Intervall $[-1, 0)$. Verbindungen mit einem SLN%-Wert von 50% bekommen die Bewertung 0. Die Ergebnisse dieser Berechnung sind in Tabelle 5.1 zu sehen.

Injektion in V1				Injektion in V4			
Projektion		SLN%	$f = \frac{(\text{SLN}\% - 50) \cdot 2}{100}$	Projektion		SLN%	$f = \frac{(\text{SLN}\% - 50) \cdot 2}{100}$
von	nach			von	nach		
V1	V1	(60,82)	(0,2164)	V1	V4	100	1
V2	V1	47,53	-0,0494	V2	V4	93,39	0,8678
V3	V1	8,54	-0,8292	V3	V4	59,75	0,1950
V4	V1	10,84	-0,7832	V4	V4	(64,27)	(0,2854)
MT	V1	5,24	-0,8952	MT	V4	47,15	-0,0570
FST	V1	0	-1	FST	V4	13,9	-0,7220
LIP	V1	2,1	-0,9580	LIP	V4	26,91	-0,4618
TEO	V1	0,44	-0,9912	TEO	V4	39,78	-0,2044
TE	V1	0,06	-0,9988	TE	V4	25,75	-0,4850
THTF	V1	0	-1	THTF	V4	0,82	-0,9836
FEF	V1	0	-1	FEF	V4	72,56	0,4512

Tabelle 5.1: Die von Barone et al. [BBKK00] durch retrograde Injektion in V1 und V4 ermittelten SLN%-Werte normalisiert auf einen Bereich von -1 bis 1 .

Barone et al. haben auch SLN%-Werte für die Projektionen V1 nach V1 und V4 nach V4 angegeben. Diese sollen hier aber unberücksichtigt bleiben, weil sie für die Berechnung der Hierarchie keine Relevanz haben, denn in einer Hierarchie hat ein Knoten zu sich immer den Abstand 0. Würden sie verwendet werden, so hätte die maximale Abweichung Δ_{max} immer mindestens den Wert 0.2854, was dem Wert der Projektion von V4 nach V4 entspricht.

Die Verbindung von V1 nach V4 hat mit 1 den größten Wert, den eine Verbindung haben kann. Dagegen ist der Abstand von V4 nach V1 mit -0.7832 deutlich kleiner. Dieses ist die einzige reziproke Verbindung, für die SLN%-Werte vorliegen, so daß man nicht sagen kann, ob die Abweichungen zwischen zwei entgegengesetzte Kanten immer so groß sind. Eine Abbildung der Areale mit den Verbindungen, für die Kantenbewertungen existieren ist in Abbildung 5.16 zu sehen.

Dieser Graph gibt nur einen kleinen Ausschnitt des visuellen Systems wieder. Es sind nur elf Areale vertreten und für die auch nur die Verbindungen, die zu V1 oder V4 hinführen. Insgesamt hat der Graph 11 Knoten und 20 Kanten. Drei der Knoten haben nur eine auslaufende Kante, die dann alleine über die Position des Knotens in der Hierarchie entscheidet. Für eine Hierarchieabbildung l muß für alle Areale A in dem Graphen die Gleichung

$$l(V4) - l(V1) = (l(V4) - l(A)) + (l(A) - l(V1))$$

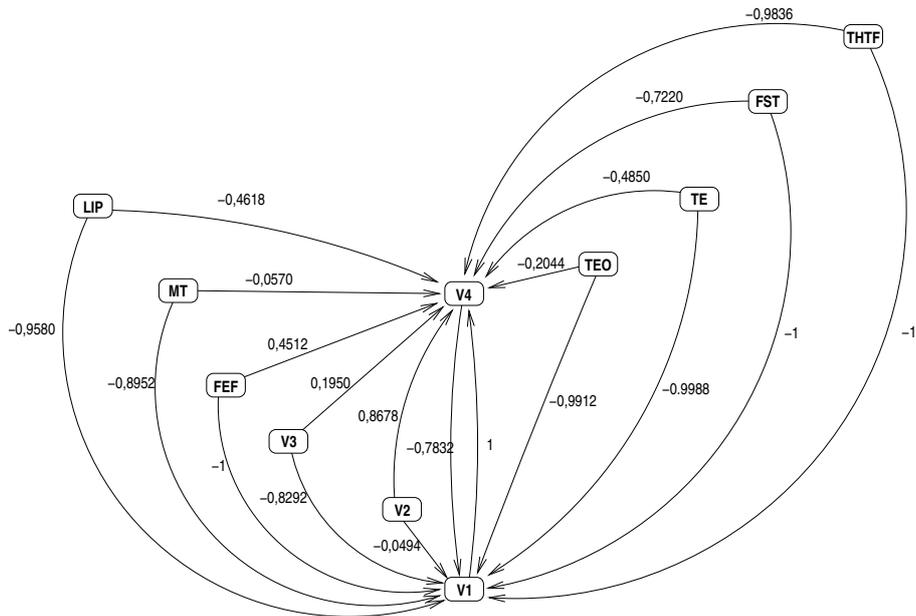


Abbildung 5.16: Der Graph aus den Daten von Barone et. al. [BBKK00] mit den zugehörigen normalisierten Kantenbewertungen.

gelten. Aber für je zwei Areale A_1, A_2 ist die Differenz der Kantenbewertungen

$$f((A_1, V4)) - f((A_1, V1)) \neq f((A_2, V4)) - f((A_2, V1)),$$

wie man aus der Wertetabelle 5.1 entnehmen kann. Demnach kann maximal für ein Areal A

$$l(V4 - l(A)) = f((A, V4)) \text{ und } l(A) - l(V1) = f((A, V1))$$

gelten, für alle anderen Verbindungen wird mindestens eine Bedingungen verletzt. Somit wird jede Hierarchie auf diesem Graphen mindestens 10 der 20 Kantenbedingungen verletzt. Insgesamt ist anzunehmen, daß aufgrund der reellen Kantenbewertungen, die sich fast alle in der Größe unterscheiden, bei einer Optimierung jedes Areal seine eigene Hierarchiestufe einnimmt.

Es wurde nach dem im Abschnitt 5.3.2 besprochenem Verfahren für den Graphen ein lineares Programm zur Berechnung einer Hierarchie erstellt. Dabei wurde zum einen die maximale Abweichung Δ_{max} und die Summe aller Abweichungen Δ_i minimiert und zum anderen nur die Summe aller Abweichungen Δ_i . Für V1 wurde die Hierarchiestufe 0 vorgegeben und für alle anderen Areale die Hierarchiestufen auf nicht negative Werte eingeschränkt. Die Ergebnisse sind in Tabelle 5.2 zusammengefaßt. Eine grafische Darstellung der Ergebnisse ist in Abbildung 5.17 zu sehen.

Beide Optimierungskriterien liefern beinahe die gleiche minimale Summe aller Abweichungen $\sum \Delta_i$. Die maximale Abweichung Δ_{max} verdoppelt sich aber fast, wenn sie nicht minimiert wird. Allerdings ist in dieser Bedingung die Gesamtzahl der Abweichungen, also die Anzahl der $\Delta_i > 0$, kleiner als in der kombinierten Bedingung. Wenn man allerdings die maximale Abweichung minimieren möchte, so ist es von Vorteil die notwendigen Abweichungen auf viele Kanten zu verteilen, um die einzelnen Abweichungen kleiner zu halten.

Minimierung von $10000\Delta_{max} + \sum \Delta_i$		Minimierung von $\sum \Delta_i$							
$\sum \Delta_i$	3,2950			3,1432					
Δ_{max}	0,3587			(0,6644)					
Anzahl Bedingungen mit $\Delta_i > 0$	15			12					
Hierarchie- stufen		Areale		Hierarchie- stufen		Hierarchie- Areale		stufen	
				THTF	1,6644	THTF			10
1,3587	FST	THTF		FST	1,5088				9
1,2188				TE	1,2718	FST			8
1,1956				LIP	1,2486	TE LIP			7
0,9912				TEO	0,9912	TEO MT			6
0,8952				MT	0,8952	V4			5
0,7338				V4	0,7868	V3 FEF			4
0,6413		FEF		V3	0,5918				3
0,5388		V3		FEF	0,3356	V2			2
0		V1	V2		0	V1			1

Tabelle 5.2: Die mit QS-Opt aus den Daten von Barone et al. [BBKK00] berechneten Hierarchiestufen für die Bedingung, daß V1 auf Stufe 0 liegt und es keine negativen Hierarchiestufen gibt. Es sind auch die Werte für Δ_{max} , $\sum \Delta_i$, sowie die Anzahl der Bedingungen, die ein $\Delta_i > 0$ erforderten, angegeben. Rechts zum Vergleich die Hierarchie aus [BBKK00].

Es ergeben sich in der einen Bedingung neun und in der anderen zehn verschiedene Hierarchiestufen, wobei zwischen den meisten Stufen ein deutlicher Abstand zu erkennen ist. Lediglich die Hierarchiewerte von LIP und TE liegen in beiden Bedingungen dicht beieinander, aber diese beiden Knoten haben auch sehr ähnliche Kantengewichte. Der Bereich in dem die Hierarchiestufen liegen reicht von 0 für V1 bis 1,3587 beziehungsweise 1,6644 für THTF. Die Reihenfolge der Areale, wenn sie aufsteigend bezüglich der Hierarchie angeordnet werden, unterscheidet sich für die beiden Bedingungen kaum und weist auch große Ähnlichkeit zu der Hierarchie von Barone et al. aus [BBKK00] auf. Diese Hierarchie hat auch zehn Stufen, allerdings sind 2 davon nicht besetzt, dafür 3 doppelt. Die Areale, die zu zweit auf einer Stufe liegen, befinden sich in den berechneten Hierarchien jeweils auf zwei aufeinanderfolgenden Stufen. Der Hauptunterschied ist, daß in der Hierarchie von Barone et al. THTF mit Abstand an der Spitze der Hierarchie ist. Dagegen liegt bei der Optimierung von Δ_{max} in Kombination mit der Summe der Δ_i das Areal THTF sogar mit FST auf einer Stufe. Bei der Optimierung nur von der Summe der Δ_i liegen die beiden Areale zwar auf verschiedenen Stufen, der Abstand zwischen diesen Stufen ist aber nicht auffallend groß. Dagegen ist der Abstand zwischen den Hierarchiestufen von V2 und der von V3 beziehungsweise FEF mit 0,5388 beziehungsweise 0,3356 in beiden Fällen der größte in der ganzen Hierarchie. Dies ist in Einklang mit der übersprungenen Stufe zwischen diesen Arealen bei Barone et al.. Alle anderen Abstände zwischen Hierarchiestufen sind kleiner als 0,2 bei der kombinierten Optimierung und kleiner als 0,26 bei der Minimierung der Summe der Abweichungen.

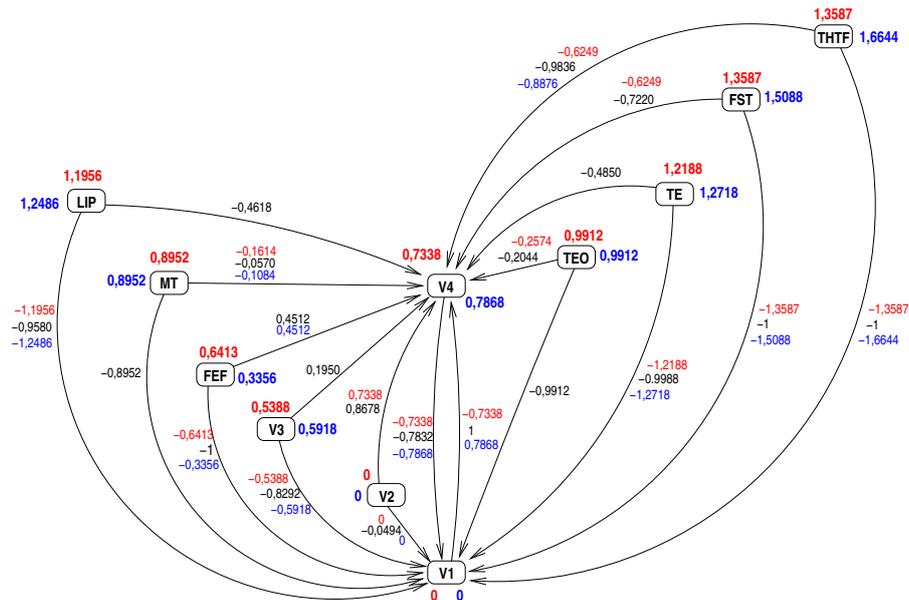


Abbildung 5.17: Der Graph aus den Daten von Barone et. al. [BBKK00] mit den normalisierten Kantenbewertungen und den Hierarchiestufen aus Tabelle 5.2. Dabei geben die roten Zahlen die Hierarchiestufen bei Minimierung von $10000\Delta_{max} + \sum \Delta_i$ und die blauen Zahlen die Hierarchiestufen bei Minimierung von $\sum \Delta_i$ an. Die Differenz der Hierarchiestufen von Ziel- und Startknoten einer Kante ist in der entsprechenden Farbe angegeben, sofern sie nicht mit dem Kantengewicht dieser Kante übereinstimmt.

Auffällig ist in den berechneten Hierarchien zum einen, daß auch V2 auf der Stufe 0 angesiedelt ist und zum anderen, daß FEF jeweils einen sehr kleinen Hierarchiewert hat, in einem Fall sogar direkt über V2 und unter allen anderen Arealen plaziert wurde. Die erste Tatsache ist leicht dadurch zu erklären, daß der Abstand von V2 zu V1 nur -0.0494 beträgt, was in beiden Bedingungen deutlich kleiner ist als die maximale Abweichung Δ_{max} . Es wäre allerdings auch möglich, eine Hierarchie zu bestimmen, in der V2 bei denselben minimierten Werten einen größeren Wert einnimmt. So würde ein Wert von 0.0494 für V2, also dem Abstand von V2 zu V1, in beiden Fällen die Qualität der Lösung nicht ändern, da die Entfernung von V1 über V2 nach V4 in der Summe $0,9172$ beträgt, der Abstand von V1 und V4 laut Hierarchie aber nur $0,7338$ beziehungsweise $0,7868$ ist. Die Differenz zu $0,9172$ ist also in beiden Fällen deutlich größer als 0.0494 . QS-Opt bevorzugt aber unter den optimalen Lösungen die Lösungen, bei denen alle Variablen möglichst kleine Werte erhalten. Der niedrige Wert von FEF ergibt sich daraus, daß FEF über eine deutlich aufsteigende Kante mit V4 verbunden ist. Folglich muß FEF klar unter V4 liegen.

Entgegen unserer Erwartungen haben wir aber keine Hierarchie erhalten, die aus elf verschiedenen Hierarchiestufen besteht. Wenn wir aber bei dem linearen Problem die Einschränkung des Wertbereichs auf nicht negative Werte weglassen, so ergibt sich für die kombinierte Optimierungsbedingung eine Hierarchie, die elf Stufen besitzt, aber dieselben Werte für die Minimierungsvariablen wie oben aufweist. Die Ergebnisse dieser Berechnung werden in Tabelle 5.3 gezeigt. Eine grafische Darstellung ist in Abbildung 5.18 zu sehen.

In diesen Berechnungen erhalten wir in beiden Fällen auch eine Verringerung der verletzten

Minimierung von $10000\Delta_{max} + \sum \Delta_i$		Minimierung von $\sum \Delta_i$			
$\sum \Delta_i$	3,2950		3,1432		
Δ_{max}	0,3587		(0,7704)	Hierarchie aus [BBKK00]	
Anzahl Bedingungen mit $\Delta_i > 0$	14		11		
Hierarchie- stufen	Areale	Hierarchie- stufen	Areale	Hierarchie- stufen	Areale
0,5295	THTF				
0,2679	FST	FST	THTF	0,1708	THTF
0,1696	TE			0,1696	
0,1620	TEO			0,1620	FST
0,1288	LIP			0,1288	TE LIP
0	V3				TEO MT
-0,0384	MT			0,0146	V4
-0,0954	V4	FEF	V3	0	V3 FEF
-0,1879	FEF	V4		-0,0424	
-0,7798	V2			-0,7798	V2
-0,8292	V1			-0,8292	V1

Tabelle 5.3: Die mit QS-Opt aus den Daten von Barone et al. [BBKK00] berechneten Hierarchiestufen für die Bedingung, daß alle Werte auch negativ sein können. Rechts zum Vergleich die Hierarchie aus [BBKK00].

Bedingungen, also der Anzahl der Verbindungen mit $\Delta_i > 0$. Damit wird bei der Minimierung von der Summe mit 11 fast die minimal mögliche Zahl an Abweichungen erreicht. Dafür ist Δ_{max} , das ja in dieser Bedingung nicht minimiert wird, größer geworden. Der Bereich in dem die Werte liegen reicht von jeweils -0,8292 für V1 bis zu 0,5295 beziehungsweise 0,1708 für THTF. Die Hierarchie für die kombinierte Optimierung hat somit wie zuvor eine Spanne von 1,3587, während die andere Bedingung diesmal nur eine Spanne von genau 1 hat, was dem Betrag des maximalen Kantengewichtes entspricht. Allerdings liegen in dieser Hierarchie auch auf zwei Stufen zwei Areale. In beiden Hierarchien gibt es eine große Differenz zwischen dem Wert von V2 und der darüberliegenden Hierarchiestufe. In der kombinierten Hierarchiebedingung gibt es diesmal einen mit 0,2616 einen relativ großen Abstand zwischen der Hierarchiestufe von THTF und der von FST. Die Sprünge zwischen den Stufen stimmen für diese Hierarchie also mit der Einteilung von Barone et al. überein. Alle anderen Abstände zwischen Hierarchiestufen sind für beide berechneten Hierarchien kleiner als 0,13. Interessanterweise haben die Areale V1, V2, LIP, TEO und TE in beiden Hierarchien dieselben Werte.

Auch die Hierarchie hat sich verändert, so liegt V3 jetzt in beiden Fällen über V4, bei der kombinierten Optimierungsbedingung sogar über MT. In der anderen Bedingung liegt V3 auf einer Stufe mit FEF, welches damit auch im Gegensatz zu den anderen Hierarchien über V4 liegt. Außerdem haben auch die Areale TEO und LIP ihre Plätze bezüglich der Ordnung durch die Hierarchie getauscht. Damit unterscheiden sich diese Hierarchien in ihrer Anordnung schon deutlich von der von Barone et al..

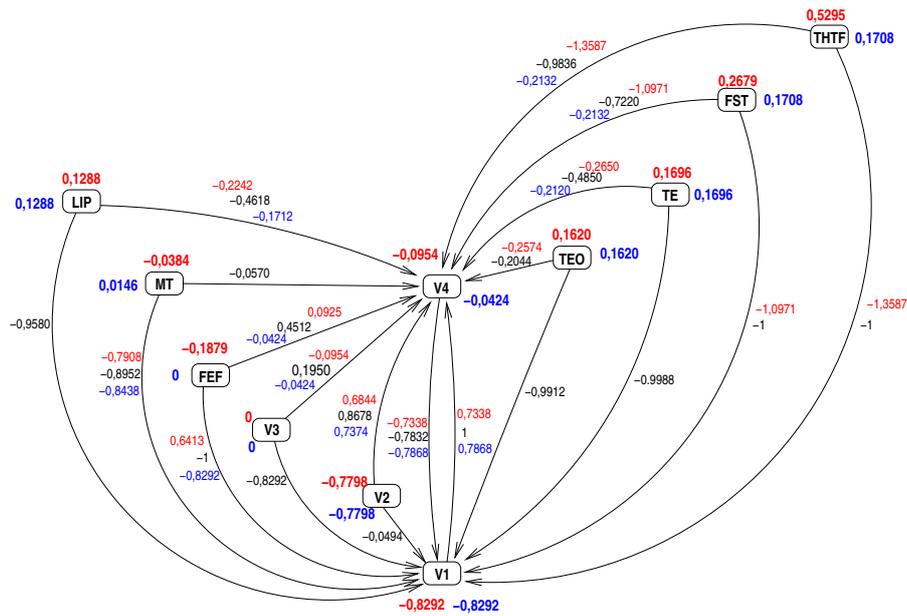


Abbildung 5.18: Der Graph aus den Daten von Barone et. al. [BBKK00] mit den normalisierten Kantenbewertungen und den Hierarchiestufen aus Tabelle 5.3. Dabei geben die roten Zahlen die Hierarchiestufen bei Minimierung von $10000\Delta_{max} + \sum \Delta_i$ und die blauen Zahlen die Hierarchiestufen bei Minimierung von $\sum \Delta_i$ an. Die Differenz der Hierarchiestufen von Ziel- und Startknoten einer Kante ist in der entsprechenden Farbe angegeben, sofern sie nicht mit dem Kantengewicht dieser Kante übereinstimmt.

Diese beiden Hierarchien lassen sich in gleichwertige Hierarchien mit V1 auf der Stufe 0 umformen, indem man von allen Hierarchiewerten den Wert von V1 abzieht. Die maximale Abweichung Δ_{max} und die Summe der Abweichungen Δ_i ändert das nicht. Wenn man aber bei einer gleich guten Lösung möglichst wenige Bedingungen verletzen möchte, dann ist also diese zweite Lösung vorzuziehen.

Mit den vorliegenden SLN%-Werten läßt sich also nach unserer Methode keine eindeutige Hierarchie und noch nicht mal eine eindeutige Rangfolge bestimmen. Allerdings stützt sich diese Berechnung nur auf wenige Daten, die in sich sehr widersprüchlich sind. Ein weiteres Problem ist es, daß bei großen hierarchischen Entfernungen ein Deckelungseffekt für den SLN%-Wertes auftritt. Die Werte können nicht größer als 100% oder kleiner als 0% werden, beziehungsweise nach unserer Umrechnung kleiner als -1 oder größer als 1. Die Abstände können also nicht beliebig groß werden, und, wie die vorhandenen Verbindungen zeigen, sind die Werte in vielen Fällen nahe am Maximum. Man könnte allerdings für Verbindungen die bei 100% beziehungsweise 0% liegen, Abstände zulassen, die größer sind als 1. Wenn weitere SLN%-Werte vorliegen, wird sich das Verhältnis der Areale zueinander wahrscheinlich besser bestimmen lassen und damit vielleicht auch eine eindeutige Hierarchie entstehen.

5.4.2 Berechnungen mit den Daten von Felleman und Van Essen

Während es natürlich wünschenswert ist, ein möglichst exaktes Abstandsmaß als Kantenbewertung zu haben, ist dieses für die Berechnung einer Hierarchie nicht unbedingt erforderlich. Es reicht auch schon aus, wenn man die Verbindungen in Bezug auf die Hierarchie in verschiedene Klassen einteilen kann, wie es zum Beispiel Felleman und Van Essen [FE91] getan haben. Diese Klassen können dann mit reellen Werten oder Wertebereichen assoziiert werden. Felleman und Van Essen haben die Verbindungen im visuellen System aufgrund von anatomischen Merkmalen, den laminaren Muster am Ursprung und Ziel, in drei Klassen eingeteilt (siehe Tabelle 5.4 und Abschnitt 5.1.1).

laminare Muster der Verbindung		
Ursprung	Ziel	Verbindungstyp
supragranular oder supra- und infragranular	Schicht 4	aufsteigend
supra- und infragranular	alle Schichten	lateral
supra- und infragranular oder infragranular	multilaminar ohne Schicht 4	absteigend

Tabelle 5.4: Klassifikation von Verbindungen bezüglich einer Hierarchie aufgrund ihrer laminaren Muster am Ursprung und Ziel nach Felleman und Van Essen [FE91]

Wenn man diese Einteilung mit der Klassifizierung von der Gruppe um Kennedy vergleicht, die nur die laminaren Muster am Ursprung betrachtet haben, so kann man die Klassen weiter verfeinern. Nach dieser Einteilung ist eine Verbindung umso stärker aufsteigend, je mehr sie von supragranularen Neuronen ausgeht. Andersherum sollen Verbindungen umso stärker absteigend sein, je mehr sie von infragranularen Neuronen ausgeht ([VFJ⁺04], vergleiche Abschnitt 5.1.3). Entsprechend kann man die Verbindungen, die nur supragranularen beziehungsweise nur infragranularen Ursprung haben, als stark aufsteigend beziehungsweise stark absteigend einstufen. Eine solche Aufteilung ist konsistent mit den Einstufungen nach dem SLN%-Wert. Man erhält damit also fünf Verbindungstypen, die jeder mit einem eindeutigen laminaren Muster gekennzeichnet sind (siehe Tabelle 5.5).

Eine Einteilung der Daten von Felleman und Van Essen nach diesen neuen Verbindungstypen findet sich im Anhang in Tabelle A.6. Es ergibt sich daraus ein Graph mit 32 Knoten und 386 Kanten. Es ist allerdings zu berücksichtigen, daß die Datenlage nicht für alle Verbindungen eine eindeutige Klassifizierung zuläßt. So ist nicht immer für Ursprung und Ziel das laminare Muster bekannt, in einigen Fällen liegen überhaupt keine laminaren Informationen vor. Verbindungen, die nicht eindeutig klassifiziert werden konnten, wurden allen in Frage kommenden Verbindungstypen zugeordnet. Damit hat man jetzt eine feinere Unterteilung der Verbindungsdaten als Felleman und Van Essen. Die Frage bleibt aber, welche Abstände man diesen Verbindungsarten zuteilt. Da die Grenzen zwischen den Kriterien fließend sind, ist es nicht sinnvoll, diskrete Werte für die fünf Verbindungsarten zu wählen. Natürlicher scheint

laminares Muster der Verbindung		Verbindungsart
Ursprung	Ziel	
supragranular	Schicht 4	stark aufsteigend
supra- und infragranular	Schicht 4	aufsteigend
supra- und infragranular	alle Schichten	lateral
supra- und infragranular	multilaminar ohne Schicht 4	absteigend
infragranular	multilaminar ohne Schicht 4	stark absteigend

Tabelle 5.5: Erweiterte Klassifikation von Verbindungen aufgrund ihrer laminaren Muster an Ursprung und Ziel

es, aneinandergrenzende Wertebereiche zu wählen, die ganz \mathbb{R} abdecken. Denn theoretisch spricht nichts dagegen, daß die hierarchischen Abstände zwischen den Arealen beliebig groß werden. Hier soll beispielhaft für verschiedene Wertebereiche, die den fünf Verbindungstypen zugeordnet werden, eine Hierarchieberechnung durchgeführt werden.

Zuerst nehmen wir an, daß eine stark aufsteigende Verbindung mindestens um 1.5 Hierarchiestufen ansteigt, eine aufsteigende Verbindung dagegen nur um 0.5 bis 1.5. Entsprechende negative Werte werden für absteigende und stark absteigende Verbindungen gewählt. Eine laterale Verbindung soll immer noch bis zu 0.5 Hierarchiestufen überbrücken. Eine genaue Übersicht ist in Tabelle 5.6 zu finde.

Verbindungsart	Wertebereich
stark aufsteigend	$[1.5, \infty)$
aufsteigend	$[0.5, 1.5]$
lateral	$[-0.5, 0.5]$
absteigend	$[-1.5, -0.5]$
stark absteigend	$(-\infty, -1.5]$

Tabelle 5.6: Zuordnung von reellen Werten zu den fünf Verbindungsarten ohne Überschneidung der Wertebereiche

Problematisch wird es, wenn für eine Verbindung nicht klar ist, welcher der drei Klassen sie angehört. Man kann die Wertebereiche nicht einfach miteinander kombinieren, weil sich dadurch nicht unbedingt ein zusammenhängendes Intervall ergibt, welches für die Optimierung benötigt wird. Um dieses zu vermeiden werden bei den hier durchgeführten Berechnungen die Lücken „aufgefüllt“. Bei Verbindungen die nicht eindeutig klassifiziert sind, wurde jeweils die größte obere und kleinste untere Grenze aller in Frage kommenden Wertebereiche verwendet. Dadurch werden eventuell auch Bereiche abgedeckt, die in der Klassifikation nicht vorgesehen sind. Für verwendete lineare Optimierung ist es aber notwendig, daß alle Verbindungen einen zusammenhängenden Wertebereich haben. Eine Verbindung, die entweder aufsteigend oder absteigend sein soll, aber nicht lateral, erhält somit trotzdem einen Wertebereich von $[-1.5, 1, 5]$ zugeordnet.

Für eine konkrete Zuweisung läßt sich natürlich auch „unendlich“ nicht als Intervallgrenze verwenden. Aber es reicht für die Berechnung aus, wenn man stattdessen einen Wert nimmt, der deutlich größer ist, als die anderen Grenzen. Hier wurde der Wert 99 für ∞ und -99 für $-\infty$ verwendet.

Mit diesen Wertebereichen wurde ein lineares Programm für das Δ -Intervall-Hierarchieproblem erstellt. Dabei wurde wieder nach zwei verschiedenen Bedingungen optimiert. Zum einen wurde $10000 \cdot \Delta_{max} + \sum \Delta_i$ minimiert, wobei Δ_{max} die maximale Abweichung und $\sum \Delta_i$ die Summe aller Abweichungen ist. Zum anderen wurde nur $\sum \Delta_i$ minimiert. Das Ergebnis ist in Tabelle 5.7 zu sehen. Zum Vergleich ist in Tabelle 5.8 noch einmal die Hierarchie von Felleman und Van Essen [FE91] angegeben.

	Minimierung von $10000 \cdot \Delta_{max} + \sum \Delta_i$	Minimierung von $\sum \Delta_i$
$\sum \Delta_i$:	29	20
Δ_{max} :	0.75	(1.5)
Anzahl Bedingungen mit $\Delta_i > 0$:	48	27
Hierarchiestufen		
	7.50	AITd
	7.25	
TH AITd	7.00	TH TF
	6.75	
TF 46	6.50	46
	6.25	
	6.00	FEF
	5.75	
STPa	5.50	AITv STPa STPp
	5.25	
FEF	5.00	7a CITd CITv
7a STPp AITv	4.75	
	4.50	MSTl MSTd FST PITd VIP PITv
MSTl CITv CITd	4.25	
FST LIP MSTd	4.00	LIP
PITv PITd	3.75	
VIP	3.50	VOT MDP DP MIP
DP PO MDP MIP	3.25	
MT	3.00	V4 V4t MT PO
VOT	2.75	
	2.50	V3A
PIP V4t V3A V4	2.25	
	2.00	PIP
	1.75	
VP V3	1.50	VP V3
	1.25	
	1.00	V2
V2	0.75	
	0.50	
	0.25	
V1	0	V1

Tabelle 5.7: Berechnete Hierarchien bei einer Einteilung der Wertebereiche laut Tabelle 5.6

Es werden bei der einen Optimierung 48 und bei der anderen 27 von 386 Bedingungen verletzt. Diese Zahl scheint im Vergleich zu bisherigen Hierarchieberechnungen, wie zum Beispiel von Hilgetag et al., hoch ([HOY96] vergleiche Abschnitt 5.1.2). Allerdings wurden bisher immer die nur Anzahl der Verletzungen unabhängig von ihrer Größe minimiert, so daß ein direkter Vergleich nicht angebracht ist. Die Summe aller Verletzungen $\sum \Delta_i$ ist für das kombinierte

Hierarchiestufen	Areale
9	46 TF TH
8	STPa AITd AITv
7	7a FEF STPp CITd CITv
6	VIP LIP MSTd MSTl FST PITd PITv
5	DP VOT
4	MDP MIP PO MT V4t V4
3	PIP V3A
2	V3 VP
1	V2
0	V1

Tabelle 5.8: Hierarchie von Felleman und Van Essen [FE91]

Optimierungskriterium 29 und für die Optimierung nur dieser Summe 20. Dafür ist die größte Abweichung Δ_{max} im ersten Fall mit 0.75 nur halb so groß wie im zweiten, wo sie 1.5 beträgt. Die Areale verteilen sich bei der kombinierten Minimierung der maximalen Abweichung und der Summe der Abweichungen auf 16 Hierarchiestufen, die Werte zwischen 0 für V1 und 7 für TH und AITd haben. Dagegen gibt es bei der einfachen Minimierung der Summe nur 15 Hierarchiestufen. Diese reichen aber von 0 für V1 bis zu 7,5 für AITd, umfassen also einen größeren Wertebereich. In beiden Fällen sind es deutlich mehr als die zehn Hierarchiestufen von Felleman und Van Essen.

Bei der kombinierten Optimierung nach der maximalen Abweichung und der Summe der Abweichungen sind die Abstände zwischen den Hierarchiestufen ungleichmäßig. Während es jeweils zwischen den untersten und den obersten Hierarchiestufen deutliche Abstände gibt, liegen die mittleren Hierarchiestufen nahe beieinander. Dieses Ergebnis überrascht, da man eigentlich annehmen würde, daß die Hierarchie nach oben flacher wird, also die Abstände zwischen den oberen Stufen kleiner werden. Der minimale Abstand zwischen zwei aufeinanderfolgenden Hierarchiestufen beträgt 0.25, der größte 0.75. Bei dieser Optimierungsbedingung befinden sich maximal vier Areale auf einer Ebene.

Bei der Optimierung nur nach der Summe der Abweichungen sind die Abstände zwischen den Hierarchie hingegen gleichmäßig. Nur zwischen den untersten beiden Hierarchiestufen ist ein Abstand von 1, alle anderen aufeinanderfolgenden Hierarchiestufen haben eine Abstand von 0.5. Auf den oberen und unteren Hierarchiestufen befinden sich jeweils maximal zwei Areale, auf den mittleren Stufen bis zu sechs.

Durch die höheren Anzahl von Hierarchiestufen verteilen sich die Areale mehr als in der Hierarchie von Felleman und Van Essen. So gibt es für beiden Optimierungsbedingungen deutlich mehr Areale, die alleine auf einer Hierarchieebene liegen. Ansonsten sind die Hierarchien denen von Felleman und Van Essen relativ ähnlich. Teilt man die Hierarchien grob in einen oberen, mittleren und unteren Bereich ein, so findet man in diesen Gruppen in allen drei Hierarchien dieselben Areale. Auffällig ist jedoch, daß AITd in beiden Berechnungen auf der obersten Hierarchieebene liegt.

Als nächstes soll untersucht werden, was passiert, wenn sich die Wertebereiche der verschiedenen Verbindungstypen stärker überschneiden. Eine solche Bedingung ist durchaus sinnvoll, da die unterschiedlichen Verbindungsmuster auch nicht unbedingt klar voneinander zu unterscheiden sind, sondern ein fließender Übergang stattfindet. Es ist dabei allerdings zu vermeiden, daß alle Bereiche die 0 enthalten, da es sonst auch eine triviale Lösung gibt. Konkret wurden die Wertebereiche in Tabelle 5.9 gewählt.

Verbindungsart	Wertebereich
stark aufsteigend	$[1.1, \infty)$
aufsteigend	$[0.2, 1.5]$
lateral	$[-0.5, 0.5]$
absteigend	$[-1.5, -0.2]$
stark absteigend	$(-\infty, -1.1]$

Tabelle 5.9: Zuordnung von reellen Werten zu den fünf Verbindungsarten mit Überschneidung der Wertebereiche

Bei nicht eindeutig klassifizierten Verbindungen wurden die Wertebereiche wieder nach dem oben beschriebenen Prinzip ermittelt. Es wurde auch in diesem Fall sowohl einmal die maximale Abweichung kombiniert mit der Summe aller Abweichungen, sowie nur die Summe aller Abweichungen minimiert. Das Ergebnis ist in Tabelle 5.10 zu sehen.

	Minimierung von $10000\Delta_{max} + \sum \Delta_i$	Minimierung von $\sum \Delta_i$
$\sum \Delta_i$:	14.90	9.8
Δ_{max} :	0.45	(1.1)
Anzahl Bedingungen mit $\Delta_i > 0$:	43	20

Hierarchiestufen		Hierarchiestufen	
6.05	AITd	AITd	6.0
5.85	TH	TH	5.6
5.40	TF 46	TF	5.4
4.75	STPa	46	5.1
4.35	FEF	FEF	4.9
4.10	STPp 7a	STPa	4.5
3.90	VIP	AITv STPp	4.3
3.80	AITv	7a	4.2
3.60	CITd CITv	MSTl CITv CITd	3.8
3.45	LIP FST MSTl MSTd	VIP MSTd	3.7
3.25	PITv PITd	PITv	3.6
2.80	PO	FST PITd	3.3
2.60	MIP DP MDP	LIP	3.2
2.40	MT	MDP DP PO MIP	2.7
2.35	V4t	VOT	2.4
2.15	VOT PIP	V4t V4 MT	2.2
1.95	V3A V4	V3A	2.0
1.30	VP V3	PIP	1.6
0.65	V2	VP V3	1.1
0	V1	V2	0.7
		V1	0

Tabelle 5.10: Berechnete Hierarchien bei einer Einteilung der Wertebereiche laut Tabelle 5.9

Die Summe der Δ_i hat sich im Vergleich zu den beiden zuvor besprochenen Hierarchien jeweils halbiert. Auch die Größe von Δ_{max} sowie die Anzahl der $\Delta_i > 0$, also der verletzen Bedingungen, ist zurück gegangen. Dies ist dadurch zu erklären, das durch die Überschneidungen bei den Bereichen mehr Bedingungen passen.

Ein weiterer Effekt von den sich überschneidenden Wertebereichen scheint zu sein, daß die Hierarchiestufen deutlich näher beieinanderliegen. Es treten auch mehr Hierarchiestufen auf, obwohl der Bereich, über den sie sich erstrecken, kleiner ist. Hier gibt es bei der kombinierten

Optimierung 20 Hierarchiestufen mit einem Wertebereich von 0 bis 6,05 und für die einfache Optimierung der Summe 21 Hierarchiestufen mit einem Wertebereich von 0 bis 6. Im ersten Fall liegen 11 Areale und im zweiten Fall 14 Areale alleine auf einer Hierarchiestufe. Das sind in beiden Fällen deutlich mehr als in der ersten Berechnung, vergleiche Tabelle 5.7. Auch hier liegt AITd in beiden Berechnungen auf der obersten Hierarchiestufe.

Die Abstände zwischen den Hierarchiestufen reichen bei der kombinierten Minimierung der größten Abweichung und der Summe aller Abweichungen von 0,05 bis 0,65. Dabei sind die größten Abstände zwischen den vier untersten Hierarchiestufen und am oberen Ende der Hierarchie zu finden, wobei der Abstand zwischen den obersten beiden Stufen mit 0,2 relativ klein ist. Einen größeren Abstand von 0,45 gibt es noch zwischen der Stufe mit den Areal PO und der mit den Arealen PITv, PITd.

Wenn man nur die Summe minimiert reichen die Abstände zwischen den Hierarchiestufen von 0,1 bis 0,7, wobei dieser größte Abstand nur zwischen V1 und V2 zu finden ist. Auch zwischen den drei darüberliegenden Stufen gibt es mit 0,4 und 0,5 große Abstände. Weitere Abstände von derselben Größenordnung gibt es noch zwischen den Stufen mit den Arealen MDP, DP, PO, MIP und LIP, den Stufen mit den Arealen MSTl, CITv, CITd und 7a, zwischen STPa und FEF sowie zwischen ganz oben zwischen AITd und TH.

Diese beiden Hierarchien unterscheiden sich schon allein durch ihre hohe Zahl an Hierarchiestufen deutlich der Hierarchie von Felleman und Van Essen. Ein weiterer Unterschied ist das Areal AITv, das hier im mittleren und nicht im oberen Bereich angesiedelt ist.

Man kann natürlich auch Wertebereiche für die Kantengewichte wählen, die sich nicht berühren. Implizit trifft man damit die Entscheidung, daß sich die verschiedenen Verbindungstypen voneinander abgrenzen lassen, man also eine deutliche Unterscheidung der Verbindungsarten hat. Der nächsten Berechnung soll die Annahme zu Grunde liegen, daß sich die fünf Verbindungstypen deutlich unterscheiden. Hier wurden beispielhaft die in Tabelle 5.11 angegebenen Kantengewichte gewählt.

Verbindungsart	Wertebereich
stark aufsteigend	$[1.7, \infty)$
aufsteigend	$[0.7, 1.3]$
lateral	$[-0.3, 0.3]$
absteigend	$[-1.3, -0.7]$
stark absteigend	$(-\infty, -1.7]$

Tabelle 5.11: Zuordnung von reellen Werten zu den fünf Verbindungsarten ohne Überschneidung der Wertebereiche

Bei nicht eindeutig klassifizierten Verbindungen wurden die Wertebereiche wieder nach dem oben beschriebenen Prinzip ermittelt. Dadurch können Kanten auch Gewichte erhalten, die nicht in einem der fünf Intervalle liegen. Es wurde auch für diese Kantengewichte sowohl einmal die maximale Abweichung, kombiniert mit der Summe aller Abweichungen, sowie nur die Summe aller Abweichungen minimiert. Das Ergebnis ist in Tabelle 5.12 zu sehen.

Mit diesen Kantengewichten erzielt man in Bezug auf die Optimierungswerte einen gegenteiligen Effekt im Vergleich zu der letzten Berechnung, wo sich die Wertebereiche überlappten. Hier sind die Summe der Δ_i , Δ_{max} und die Anzahl der $\Delta_i > 0$ für beide Bedingungen deutlich

5.4 Berechnung von Hierarchien im visuellen System

	Minimierung von $10000\Delta_{max} + \sum \Delta_i$	Minimierung von $\sum \Delta_i$
$\sum \Delta_i$:	43.80	32.9
Δ_{max} :	0.95	(1.9)
Anzahl Bedingungen mit $\Delta_i > 0$:	74	52

	Hierarchiestufen		
7.00	AITd		
6.95	TH	TH AITd	7.6
6.25	TF 46	TF	7.3
5.50	STPa	46	6.9
5.00	FEF	FEF	6.5
4.75	STPp 7a	STPa	5.9
4.55	AITv	7a STPp AITv	5.6
4.30	VIP	CITd CITv	5.2
4.25	CITv CITd	MSTl	4.9
4.20	MSTl	FST PITd	4.8
4.00	LIP MSTd FST	VIP MSTd	4.7
3.55	PITd PITv	PITv	4.5
3.45	MDP DP MIP	LIP DP	4.4
3.25	PO	MDP MIP	4.3
3.00	MT	VOT	3.8
2.85	VOT	PO	3.4
2.70	V4t	V4t	3.2
2.50	PIP	MT V4	3.1
2.25	V3A V4	PIP V3A	2.4
1.50	VP V3	VP V3	1.4
0.75	V2	V2	0.7
0	V1	V1	0

Tabelle 5.12: Berechnete Hierarchien bei einer Einteilung der Wertebereiche laut Tabelle 5.11

höher als bei den beiden vorherigen Berechnungen.

Die Anzahl der Hierarchiestufen ist mit 22 für die kombinierte Optimierung im Vergleich zur letzten Berechnung noch einmal leicht angestiegen. Dabei wird ein Wertebereich von 0 bis 7 abgedeckt. Die größten Abstände zwischen Hierarchiestufen sind wieder jeweils am oberen und unteren Ende der Hierarchie zu finden. Es treten dabei Werte bis zu einer Größe von 0,75 auf. Ein etwas größerer Sprung von 0,45 findet sich auch zwischen der Stufe mit PITd, PITv und der Stufe mit LIP, MSTd, FST. Alle anderen Abstände haben Werte zwischen 0,05 und 0,25.

Im Fall der Optimierung der Summe sind es auch diesmal 21 Hierarchiestufen, die sich über einen Wertebereich von 0 bis 7,6 erstrecken. Hier befinden am unteren Ende der Hierarchie die größten Abstände mit Werten von 0,7 und 1,0. Dann gibt es noch größere Abstände von der Stufe, auf der sich das Areal VOT befindet, sowohl nach oben mit 0,4 als auch nach unten mit 0,6. Ab der Hierarchiestufe mit dem Areal MSTl haben nach oben hin alle aufeinanderfolgenden Stufen einen Abstand zwischen 0,3 und 0,6. Alle verbleibenden Übergänge haben die Werte 0,1 oder 0,2.

Diese beiden Hierarchien sind mit maximal drei Arealen auf einer Hierarchiestufe schmaler, als die bisher betrachteten.

Zum Abschluß wollen wir noch untersuchen, was für Ergebnisse erzielt werden, wenn wir den Verbindungen dieselben Wertebereiche zuordnen, wie Fellman und Van Essen es getan ha-

ben. Dazu soll die Verfeinerung von drei auf fünf Verbindungsklassen ignoriert werden. Es muß dann natürlich die Frage geklärt werden, wie man diese drei Klassen von Verbindungen in reelle Wertebereiche übersetzen kann. Bei Felleman und Van Essen scheint die Annahme zu Grunde zu liegen, daß die aufsteigenden und die absteigenden Verbindungen jeweils mindestens eine aber ansonsten unbeschränkt viele Hierarchiestufen überbrücken könne, die laterale hingegen keine. Mit reellen Wertebereichen läßt sich dieser Sachverhalt wie in Tabelle 5.13 ausdrücken.

Verbindungsart	Wertebereich
stark aufsteigend	$[1, \infty)$
aufsteigend	$[1, \infty)$
lateral	0
absteigend	$(-\infty, -1]$
stark absteigend	$(-\infty, -1]$

Tabelle 5.13: Zuordnung von reellen Werten zu den Verbindungsarten von Felleman und Van Essen [FE91]

Auch in diesem Fall ergibt keine Kombination der drei verbleibenden Verbindungstypen ein zusammenhängendes Intervall, da keine Werte zwischen -1 und 0 sowie zwischen 0 und 1 vorgesehen sind. Die Wertebereiche sind in diesem Fall disjunkt. Wir werden aber trotzdem bei nicht eindeutigen Klassifikationen wieder wie zuvor immer die größte in Frage kommende Obergrenze und die kleinste Untergrenze wählen. Damit ergibt sich immer ein zusammenhängendes Intervall und gegebenenfalls als Folge auch zulässige Werte für Kanten im Bereich zwischen -1 und 0 sowie 0 und 1 .

Es wurde auch hier zum einen die Summe der Abweichungen und zum anderen die maximale Abweichung kombiniert mit der Summe der Abweichungen minimiert. Die Ergebnisse der Hierarchieberechnung sind in Tabelle 5.14 zu sehen.

Als erstes fällt auf, daß diesmal in beiden Optimierungsbedingungen Hierarchien mit ganzzahligen Werten für die Hierarchiestufen vorliegen. Damit können die Abweichungen Δ_i auch nur ganzzahlige Werte haben. Bei der kombinierten Optimierungsbedingung war $\Delta_{max} = 1$ und es gab 8 Verbindungen, für die das jeweilige Δ_i 1 war. Für die andere Optimierungsbedingung hatte Δ_{max} einen Wert von 2 und es gab insgesamt sechs Verbindungen, für die das jeweilige Δ_i entweder den Wert 1 oder 2 hatte.

Die beiden Optimierungsbedingungen erzeugten fast dieselbe Hierarchie, lediglich das Areal FEF wurde in einem Fall auf Stufe 7 und im anderen Fall auf Stufe 8 angesiedelt. Die Areale MDP und MIP wurden zwar auf Level 0 plaziert, aber sie könnten auch auf jeder anderen Stufe liegen, da ihre Verbindungen sie nicht einschränken. Alle Verbindungen die MDP und MIP haben, können sowohl aufsteigend, als auch lateral wie auch absteigend sein.

Beide Hierarchien haben 11 Hierarchiestufen und somit eine mehr als die Hierarchie von Felleman und Van Essen. Ansonsten sind sie dieser aber recht ähnlich. Der größte Unterschied ist, daß Areal AITd, hier wieder an der Spitze der Hierarchie steht, also über TF, TH und 46 und nicht darunter. Ansonsten befinden sich alle Areale entweder auf derselben Hierarchiestufe wie bei Felleman und Van Essen, oder auf der um eins höher liegenden beziehungsweise der um eins tiefer liegende Hierarchiestufe. Damit wurden die Areale um maximal eine Stufe verschoben. (Die Areale MDP und MIP können wie oben erklärt auf einer beliebigen Stufe

5.4 Berechnung von Hierarchien im visuellen System

	Minimierung von $10000\Delta_{max} + \sum \Delta_i$	Minimierung von $\sum \Delta_i$
$\sum \Delta_i$:	8	7
Δ_{max} :	1	(2)
Anzahl Bedingungen mit $\Delta_i > 0$:	8	6
	Hierarchiestufen	
	AITd 46 TF TH STPa AITv 7a FEF STPp CITd CITv DP VIP LIP MSTd MSTl FST PITd PITv VOT PO MT V4t V4 V3A PIP VP V3 V2 MDP MIP V1	10 9 8 7 6 5 4 3 2 1 0
		AITd 46 TF TH STPa AITv FEF 7a STPp CITd CITv DP VIP LIP MSTd MSTl FST PITd PITv VOT PO MT V4t V4 V3A PIP VP V3 V2 MDP MIP V1
	Hierarchiestufen	
	9 8 7 6 5 4 3 2 1 0	46 TF TH STPa AITd AITv 7a FEF STPp CITd CITv VIP LIP MSTd MSTl FST PITd PITv DP VOT MDP MIP PO MT V4t V4 PIP V3A V3 VP V2 V1
		$\sum \Delta_i$: 13 Δ_{max} : 2 Anzahl Bedingungen mit $\Delta_i > 0$: 10

Tabelle 5.14: Oben: Berechnete Hierarchien zu den Verbindungsarten von Felleman und Van Essen [FE91]. Unten zum Vergleich die Hierarchie von Felleman und Van Essen, sowie die Werte, die diese Hierarchie erzielt.

eingefügt werden.

Daß es bei dieser Berechnung im Vergleich zu den anderen berechneten Hierarchien so wenig verletzte Bedingungen gibt, liegt an der Beschränkung auf drei Verbindungstypen. Offensichtlich entstanden bei den anderen Berechnungen einige Verletzungen durch die Differenzierung von absteigend und stark absteigend sowie aufsteigend und stark aufsteigend.

In Tabelle 5.15 sind alle Verbindungen angegeben, die von einer der drei Hierarchien in Tabelle 5.14 verletzt wurden.

Auffällig ist, daß alle Verbindungen in Tabelle 5.15 Areale in höheren Hierarchiestufen verbinden. Dies könnte ein Hinweis darauf sein, daß der Datenbestand insbesondere für diese Areale inkonsistent ist. Für vier der Verbindungen (LIP nach PO, MSTd nach 7a, TH nach 46 sowie STPp nach FEF) wird die Verbindungsart sogar in allen drei Hierarchien nicht berücksichtigt.

Bei alle Hierarchien die mit dem modifizierten Datensatz von Felleman und Van Essen berechnet wurden, liegen AITd, TH, TF und Area 46 auf den obersten Hierarchiestufen. Dagegen sind Vp, V3, V2 und natürlich V1 immer auf den untersten zu finden. Das spricht dafür, daß die Hierarchie nicht nur am unteren sondern auch am oberen Ende deutlich ausgeprägt ist. Dafür spricht auch, daß sowohl am unteren als auch am oberen Ende der Hierarchie die Abstände zwischen den Hierarchiestufen in der Regel deutlich größer waren als zwischen Stu-

Von	Nach	laminare Ursprung	Muster Ziel	Klassifizierung der Verbindung	bei Minimierung von $\Delta_{max} + \sum \Delta_i$		Hierarchie aus [FE91]
LIP	PO	B	C/F	L/A?	D	D	D
MSTd	7a	B	C	L	A	A	A
MSTd	DP	B	C	L	L*	L*	D
DP	MSTl	S		A?	L	L	A*
TH	46	I/B		D?	L	L	L
AITd	46		M(s)	D?	D*	D*	A
46	AITd	S		A	A*	A*	D
FEF	STPp	B	M(s)	D?	L	D*	L
STPp	FEF	B	M(s)	D?	L	A	L
7a	FEF	B	C/M?	L/D?	L*	A	L*
CITd	FEF	S		A	L	A*	L
CITv	FEF	S		A	L	A*	L

Tabelle 5.15: Die zwölf Verbindungen, deren Klassifikation durch eine der Hierarchien in Tabelle 5.14 verletzt wird. Die Richtung ist jeweils angegeben mit D für absteigend, L für lateral und A für aufsteigend. Eine Übereinstimmung mit der Klassifikation ist durch * gekennzeichnet.

fen im mittleren Bereich der Hierarchie. Es scheint also einen klaren hierarchischen Aufbau sowohl der unteren als auch der ober Hierarchiestufen zu geben, während die Abstufung in der Mitte nicht so deutlich ist.

5.4.3 Diskussion

Die hierarchische Struktur des visuellen Systems läßt sich auch mit diesen neuen Ansatz deutlich zeigen. Eine wesentliche Neuerung war dabei die Einführung einer kontinuierlichen Skala, um die Hierarchie zu erstellen und die Güte zu messen. Diese orientiert sie sich mit ihren beliebig kleinen Übergängen an der Natur, die ja auch fließende Übergänge bietet. So scheint die Einteilung von Verbindungen in diskrete Klassen künstlich, auch wenn sie weit verbreitet ist. In diesem Zusammenhang scheint auch das bisher verwendete Kriterium der verletzten Beziehungen nicht angebracht, da diese nicht berücksichtigen, wie die Beziehung aussieht und wie sehr die stattdessen verwendete von ihr abweicht. In dem hier vorgestellten Ansatz wurde die Idee verwirklicht, zu erfassen, wie die Verletzung genau aussieht.

Die flexiblere Einteilung, die eine reelle Hierarchieskala im Vergleich zu einer diskreten bietet, ist offensichtlich von Vorteil, weil sie es ermöglicht, die Hierarchien feiner zu erfassen. Die berechneten Hierarchien haben auch alle mehr Stufen und dafür weniger Areale auf den einzelnen Stufen. Sie sind sozusagen „schmäler“. Die erhoffte feinere Ausdifferenzierung der Hierarchiestruktur ist also tatsächlich eingetreten.

Interessant ist auch, daß die Stufen im Vergleich zu den Verbindungswerten in fast allen Fällen klein waren. So lag der Wert der obersten Stufe einer Hierarchie in der Regel deutlich unter der Anzahl der Hierarchiestufen. Somit scheint der Wert 1 nicht der natürliche Abstand für diese Hierarchiestufen zu sein.

Abgesehen von der Verwendung einer reellen Skala, wurden auch Verfahren vorgestellt, um Hierarchien effizient zu berechnen. Dabei stellen die hier präsentierten Berechnungen nur eine

erste Annäherung an die tatsächliche Hierarchie des visuellen Systems dar. Diese kann aber jederzeit mit neuen Daten verbessert werden.

Das verwendete Verfahren liefert immer eine Hierarchie, die optimal zu den vorhandenen Werten paßt. Auch wenn die Hierarchie nicht exakt ist, so ist sie doch eine, die den vorhandenen Daten am ehesten entspricht. Aufgrund der Datenlage ist auch hier natürlich keine exakte Lösung möglich, da die Daten nicht widerspruchsfrei sind.

Man sollte natürlich auch berücksichtigen, daß die hier berechneten Hierarchien nur eine von unendlich vielen ist, die die Kriterien genauso gut erfüllen. Es handelt sich also jeweils um eine optimale Hierarchie, aber nicht um die einzige. Diese ist jedoch keine Konsequenz des Verfahrens, sondern der Datenlage.

Außerdem ist die Hierarchie davon abhängig, welche Wertebereiche man verwendet und welche Bedingungen man optimiert. Dabei scheinen beide hier vorgestellten Optimierungskriterien sinnvoll, auch wenn sie entgegengesetzte Effekte haben. Das eine Verfahren minimiert in erster Linie die maximale Abweichung, was häufig zur Folge hat, daß mehr Bedingungen verletzt werden, aber die einzelnen weniger stark. Dagegen bewirkt eine Minimierung allein der Summe der Abweichungen in der Regel eine kleinere absolute Zahl an verletzten Bedingungen. Der Fehler wird also nicht verteilt, sondern konzentriert. Allerdings gibt es keine Garantie daß dieser Effekt auftritt.

Kapitel 6

Streams im visuellem System

6.1 Einführung

Eine der zentralen Fragen der Neurologie ist, wie sich die Informationsweiterleitung und -verarbeitung im Gehirn organisiert. Im letzten Kapitel wurde der hierarchische Aufbau der Areal-Netzwerke am Beispiel des visuellen Systems besprochen. In diesem Kapitel werden wir uns mit einem weiteren Aspekt der Organisation des visuellen Systems beschäftigen. Es wird nicht nur angenommen, daß das visuelle System hierarchisch organisiert ist, sondern auch, daß es entlang dieser Hierarchie parallele Informationsverarbeitungswege für die verschiedenen Qualitäten der visuellen Information gibt. Verschiedene Aspekte eines sensorischen Eindrucks sollen zunächst getrennt verarbeitet werden und dann auf den höheren Hierarchieebenen des visuellen Systems zu einem ganzheitlichen Bild zusammengefügt werden.

Mishkin, Ungerleider und Macko kamen aufgrund von Läsionsstudien bei Primaten zu dem Schluß, daß es zwei getrennte Verarbeitungswege für Objektwahrnehmung und räumliches Sehen gibt [UM82, MUM83]. Dem liegt die Beobachtung zugrunde, daß Areale, die auf derselben Hierarchieebene liegen, zum einen eine unterschiedliche funktionelle Spezifität und zum anderen bevorzugt Verbindungen zu Arealen mit derselben Spezifität auf anderen hierarchischen Ebenen besitzen. Die separaten Verarbeitungsströme beschreiben also eine vertikale Teilung der Hierarchie. Diese getrennten Verarbeitungswege werden auch der dorsale und der ventrale Stream beziehungsweise aufgrund ihrer Funktion der „where is it“ und „what is it“ Stream genannt. Da die Streams entlang der Hierarchiestruktur verlaufen, sollte also auch innerhalb der Streams jeweils eine hierarchische Struktur bestehen.

Die meisten Studien zum Thema Streams basieren auf Beobachtung von funktionellen Aspekten der visuellen Verarbeitung. Eine Einführung in diesen Bereich der Forschung bietet zum Beispiel das Buch von Milner und Goodale [MG95]. Im Rahmen dieser Arbeit stellt sich aber die Frage, ob es auch ein strukturelles Korrelat der parallelen Informationsverarbeitungsströme des visuellen Systems gibt. Wir wollen uns also damit beschäftigen, ob und wie sich Streams anhand von Grapheigenschaften beschreiben lassen. Ziel dieses Kapitel soll es sein, einen Stream in Bezug auf seine Verbindungseigenschaften zu definieren. Mit Hilfe einer solchen Definition sollte man die parallelen Verarbeitungswege auch anhand der Verbindungsmatrix des visuellen Systems identifizieren können.

Als erstes gehe ich in diesem Zusammenhang auf die Arbeit von Young [You92] ein und

versuche zu klären, was für eine topologische Definition von Streams der Autor verwendet.

6.2 Die Untersuchung von Young

Eine der wenigen Untersuchungen, die versucht, sich den Streams über strukturelle Eigenschaften des Verbindungsnetzwerkes im visuellen System zu nähern, ist die Arbeit von Young [You92]. Young untersuchte die Verbindungsmatrix von Primaten und kam zu dem Schluß, daß es auch topologische Anzeichen für diese getrennten Verarbeitungswege gibt. Seine Analyse soll in diesem Abschnitt beschrieben werden.

Youngs Hypothese war, daß sich die Dichotomie der Informationsverarbeitung auch in den Verbindungsstrukturen finden lassen sollte. Dem lag die Annahme zugrunde, daß es zu mehr Informationsaustausch innerhalb eines Streams als zwischen den Streams kommen sollte, da die Areale eines Streams ja mit einer gemeinsamen Aufgabe beschäftigt sind, und sich diese Eigenschaft auch in der Verbindungsstruktur des visuellen Systems widerspiegeln müße. Es soll also im allgemeinen mehr Verbindungen zwischen den Arealen eines Streams geben als zwischen Arealen, die sich nicht in demselben Stream befinden.

Um seine Hypothese zu testen, benutzte Young Verbindungsdaten vom visuellen System des Makaken, wobei er sich an den Daten von Felleman und Van Essen [FE91] orientierte. Er verwendete insgesamt 29 Areale und vergab für jedes Arealpaar einen Verbindungswert entsprechend einer dreistufigen Skala: 0, wenn die Areale keine Verbindung haben, 1, wenn es eine Verbindung in eine Richtung gibt und 2, wenn es Verbindungen in beide Richtungen gibt. Young setzte ein Analyseverfahren ein, das „Non-Metric Multidimensional Scaling“ genannt wird. Dabei werden die Kantengewichte (0, 1, 2) als Bewertungen der Qualität der Verbindung auf einer Ordinalskala betrachtet. Die Ausgabe ist eine Punktmenge in der Ebene, wobei jeder Punkt ein Areal repräsentiert. Die Abstände zwischen den Punkten entsprechen „so genau wie möglich“ (as close as possible) der umgekehrten Rangordnung der „Nähe“ der Areale in Bezug auf die definierte dreistufige Skala. In der entstandenen Struktur haben Areale, die verbunden sind, eine minimale Entfernung unter der Zusatzbedingung, daß Areale, die nicht verbunden sind, eine maximale Entfernung erhalten. Dabei sollte eine Verbindung, die mit 2 bewertet wurde, eine geringere Entfernung haben als eine, die mit 1 bewertet wurde. Die Abbildung 6.1 zeigt das beste Ergebnis in einer zweidimensionalen Darstellung.

Auffällig ist, daß die Areale auf einem Oval angeordnet sind. Man erkennt eine obere und eine untere Gruppe von Arealen. Diese ordnete Young dem dorsalen beziehungsweise dem ventralen Stream zu. Für das Konzept von Streams braucht man aber nicht nur separate Verarbeitungswege, sondern auch einen Start und ein Ziel, die durch diese Wege verbunden werden. Der Ausgangspunkt für die Streams ist als unterstes Areal in der Verarbeitungshierarchie der primäre visuelle Cortex V1, als Zielareale betrachtet Young die Areale 46 und STPa. Dem oberen, dorsalen Stream ordnet er also 17, dem unteren, ventralen Stream 10 Areale zu. Die genaue Einteilung ist in Tabelle 6.1 zu finden. Es fällt auf, daß V1 im ventralen Stream nur zu dem Areal V4 projiziert. Dieses könnte man also auch als Startareal des ventralen Streams auffassen. Im dorsalen Stream gibt es keinen entsprechenden Anfangspunkt, da mehrere Areale von V1 Projektionen erhalten.

Der Graph wurde von Young visuell (also durch Anschauen) ausgewertet. Er faßte die Areale, die sich am nächsten sind, zu Gruppen zusammen. Dies tat er entlang der Ränder des Ovals.

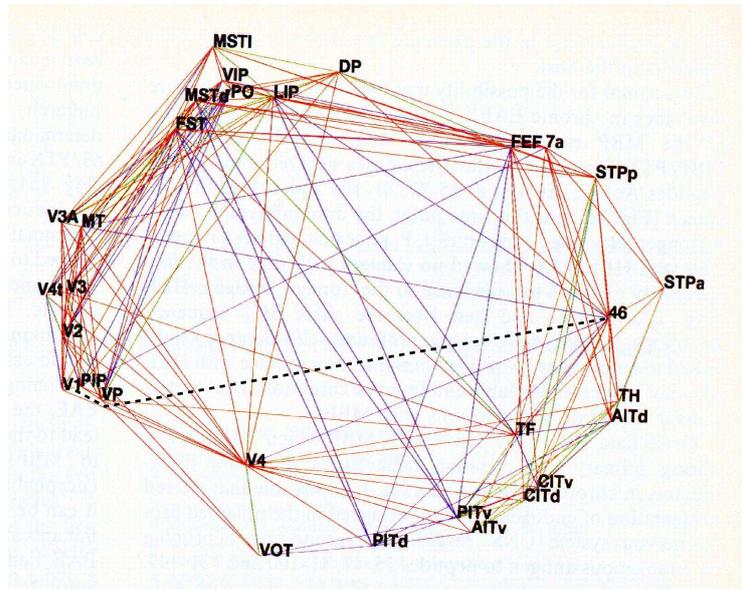


Abbildung 6.1: Einteilung der Areale im visuellen System in den dorsalen und den ventralen Stream nach Young [You92].

Startareal	V1
dorsaler Stream	V2, V3, V3a, MT, PO, PIP, VP, V4t, MSTd, MSTI, FST, VIP, LIP, DP, FEF, A7a, STPp
ventraler Stream	V4, VOT, PITd, PITv, CITd, CITv, AITd, AITv, TH, TF
Zielareale	A46, STPa

Tabelle 6.1: Einteilung der Areale im visuellen System in den dorsalen und den ventralen Stream nach Young [You92].

Da ja Areale mit ähnlichen Verbindungsstrukturen sich in der Abbildung näher sein sollen als solche mit stark verschiedenen Verbindungsstrukturen, folgert Young, daß die Verbindungen zwischen den Streams viel weniger dicht sind als innerhalb der Streams, es also zwischen den Arealen der zwei Streams tatsächlich weniger Verbindungen gibt als zwischen den Arealen innerhalb eines Streams.

Young kommt zu dem Schluß, daß die Aufteilung sowohl die von Felleman und Van Essen beschriebene Hierarchie widerspiegelt [FE91] als auch die funktionelle Dichotomie zwischen den dorsalen und ventralen Arealen. Er postulierte vier Organisationsprinzipien für das visuelle System:

1. Es dichotomisiert in zwei Streams.
2. Beide Streams bilden Hierarchien.
3. Die Streams vereinen sich wieder in Area 46 und STPa.
4. Benachbarte Areale haben die Tendenz, sich gegenseitig zu innervieren.

Kritik an Youngs Analyse gab es von Goodhill, Simmen und Willshaw [GSW95, SGW94]. Diese wandten ein, daß bei dem verwendeten multidimensionalen Skalierungsverfahren immer eine solche ovale Anordnung zu erwarten sei. Diese sei kein Hinweis auf Streams, sondern eine Konsequenz aus der Tatsache, daß die Daten nur drei Abstufungen aufweisen. Insgesamt zweifelten sie deshalb an der Aussagekraft von Youngs Ergebnissen.

6.3 Kriterien für Streams

In diesem Kapitel soll der Versuch unternommen werden, Streams über Grapheigenschaften zu definieren. Dabei sollen gerichtete Graphen verwendet werden, damit die genaue Verbindungsstruktur in die Definition mit eingehen kann. Wir nehmen als Grundlage unserer Berechnungen die von Young verwendeten Verbindungsinformationen, allerdings so modifiziert, daß sie die Verbindungsmatrix für einen gerichteten Graphen bilden (siehe Anhang, Tabelle A.1).

Implizit wurden von Young die folgenden Grundannahmen über die Struktur von Streams verwendet:

1. Die Streams haben gemeinsame Start- und Zielareale. Damit muß es auch in jedem Stream mindestens einen Weg vom Startareal zu einem Zielareal geben.
2. Areale mit ähnlicher Verbindungsstruktur (die ja in Youngs Graphen nahe beieinander abgebildet werden) gehören zu demselben Stream.
3. Die beiden Streams haben ungefähr dieselbe Anzahl von Arealen.

Zusätzlich nimmt Young noch explizit an, daß beide Streams Hierarchien bilden. Wir wollen im folgenden versuchen, diese Annahmen in konkrete Kriterien umzusetzen. Dazu wollen wir zuerst die Start- und Zielareale identifizieren, um dann verschiedene Möglichkeiten der Einteilung der übrigen Areale in Streams zu betrachten.

6.4 Finden von Start- und Zielarealen

Man geht also davon aus, daß es ein Startgebiet gibt, von dem die Informationsverarbeitung ausgeht, dann folgen parallele Verarbeitungswege, und am Ende werden die Informationen in ein gemeinsames Zielgebiet geleitet und dort integriert. Diese Ausgangs- und Zielareale sollen jetzt bestimmt werden. Sie sind an den Verarbeitungsprozessen in beiden Streams beteiligt und gehören somit beiden Streams an. Grundsätzlich kann es mehrere Start- und Zielareale geben.

Da da allgemein angenommen wird, daß die kortikale visuelle Verarbeitung im Areal V1 beginnt (siehe z.B. [FE91]), wählen wir den primären visuellen Cortex V1 als Startareal für den Ursprung der Streams. In der Regel wird auch des Areal V2 und zum Teil auch das Areal V3, die nacheinander in der Verarbeitungshierarchie auf V1 folgen, als zu beiden Streams gehörend angesehen. Wir wollen uns aber hier an der Einteilung von Young orientieren, und beschränken uns deshalb auf V1 als Startareal. Davon ausgehend wollen wir Zielareale finden.

Wir nehmen an, daß Zielareale diejenigen sind, die am weitesten von unserem Startareal V_1 entfernt sind. Die visuelle Information muß also auf ihren Weg von V_1 ausgehend über die meisten Zwischenstationen laufen, um diese Areale zu erreichen. Als Abstandsmaß nehmen wir also die Anzahl der Verbindungen in einem kürzesten Weg von V_1 zum Zielareal. Dabei sind Areale, die nicht von V_1 aus erreichbar sind, für die Bestimmung der Zielareale nicht relevant.

Formal läßt sich das dann folgendermaßen ausdrücken: Gegeben sei ein gerichteter stark zusammenhängender Graph $G = (V, E)$ und ein Knoten $s \in V$. Gesucht wird die Menge

$$\{v \in V \mid f_{s,G}(v) \geq f_{s,G}(u) \text{ für alle } u \in V\},$$

wobei $f : V \rightarrow \mathbb{N}_0$, so daß für jeden Knoten $v \in V$ gilt

$$f_{s,G}(v) = \min\{k \mid s = u_1, \dots, u_k = v \text{ ist ein Weg in } G, u_i \in V\}.$$

Die Beschränkung auf stark zusammenhängende Graphen bewirkt, daß es von s aus zu allen anderen Knoten in G einen Weg gibt und somit alle Werte von $f_{s,G}$ definiert und kleiner oder gleich $|V|$ sind. Diese minimale Weglänge $f_{s,G}$ von einem Startknoten s läßt sich mit Hilfe einer Breitensuche, beginnend bei dem Startknoten s , in Zeit $O(|V| + |E|)$ bestimmen. Dazu wird der Wert $f_{s,G}(s) = 1$ gesetzt, und jeder Knoten v erhält, wenn er über die Kante $(u, v) \in E$ zum ersten mal gefunden wird, den Wert $f_{s,G}(v) = f_{s,G}(u) + 1$. Die Knoten mit den maximalen Werten können dann in Zeit $O(|V|)$ bestimmt werden.

Ein gerichteter Graph $G = (V, E)$ mit einer solchen Knotenbewertung $f_{s,G}$ wird im folgenden auch *Levelgraph* genannt. Zwei Knoten $u, v \in V$ liegen auf demselben Level, wenn $f_{s,G}(u) = f_{s,G}(v)$ ist.

Ein Beispiel für eine Einteilung eines Graphen in Level durch eine Breitensuche ist in Abbildung 6.2 zu sehen.

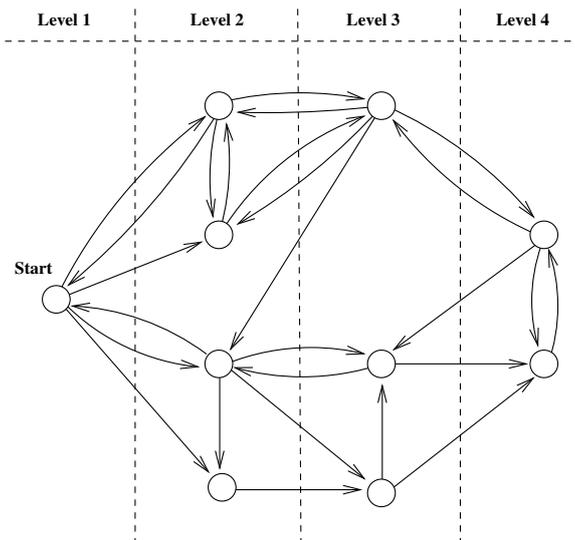


Abbildung 6.2: Beispiel für einen Graphen und die aus einer Breitensuche resultierenden Leveln.

Zuerst bestimmen wir also in dem durch die Verbindungsmatrix von Young gegebenen Graphen mit Hilfe einer Breitensuche ausgehend vom Areal V1 den Abstand der anderen Areale von V1. Die Areale mit demselben Abstand bilden einen Level. V1 liegt auf Level 1. Es ergeben sich dabei vier Leveln (siehe Tabelle 6.2). Das heißt, daß jedes Areal über maximal drei Kanten von Areal V1 aus erreicht werden kann.

Level 1	V1
Level 2	V2, V3, V3a, V4, MT, PO, PIP
Level 3	Vp, VOT, V4t, MSTd, MSTl, FST, VIP, TF, LIP, DP, PITd, PITv, CITd, CITv, AITv, TH, A46, FEF, A7a
Level 4	STPp, AITd, STPa

Tabelle 6.2: Die Leveln der Areale im visuellen System abhängig von ihrer Entfernung zu V1.

Auf Level 2 befinden sich alle Areale, zu denen V1 projiziert. Wie schon erwähnt liegt in Youngs ventralem Stream nur eines dieser Areale liegt, nämlich V4, während die anderen alle im dorsalen Stream liegen. Allerdings werden alle anderen Areale auf Level 2 auch in funktionellen Aufteilungen nie (ausschließlich) dem ventralen Stream zugeordnet.

Die Areale auf Level 4 werden im folgenden als Endbereich der Verarbeitung betrachtet, da sie den größten Abstand zu V1 haben. Es handelt sich dabei um die Areale STPp, STPa und AITd. Diese Festlegung unterscheidet sich von der Auswahl von Young, der Area 46 und STPa als Zielareale ausmachte [You92]. Allerdings wurden auch die Areale STPp und AITd, die bei Young keine Zielareale sind, von Felleman und Van Essen auf eine der oberen Stufen ihrer Hierarchie angesiedelt [FE91]. Insbesondere das Areal AITd wurde bei jeder Hierarchieberechnung im visuellen System im Kapitel 5 auf der obersten Hierarchieebene angesiedelt. Damit scheint die Wahl dieser Areale als Zielareale sinnvoll, da die Streams ja auch wieder Hierarchien bilden sollen. Das Ziel der Streams sollte also im oberen Hierarchiebereich liegen, genau wie der Anfang am unteren Ende der Hierarchie zu finden ist.

Somit haben wir Start- und Endareale der möglichen Streams festgelegt. Wir suchen jetzt nach einer disjunkten Aufteilung der übrigen 26 Areale, also der auf Level 2 und 3, in zwei disjunkte Streams. Da Young nur zwei Zielareale definiert, bleiben ihm insgesamt 27 Areale für die Zuordnung zu den beiden Streams.

6.5 Verschiedene Ansätze zur Definition von Streams

Wie soll die Aufteilung aufgrund der Verbindungsstruktur erfolgen? Die Annahme ist ja, daß Areale mit ähnlicher Verbindungsstruktur zu demselben Stream gehören (vergleiche Kriterium 2 in Abschnitt 6.3). Wie aber läßt sich dieses operationalisieren? Im folgenden wollen wir uns überlegen, nach welchen Kriterien wir eine Aufteilung der Areale, die nicht zu Start- und Zielarealen gehören, in zwei Gruppen vornehmen können.

6.5.1 Definition über die Verbindungen

Wie kann also die größere Ähnlichkeit der Verbindungsstrukturen innerhalb der Streams aussehen? Da die Streams der gezielten Informationsverarbeitung dienen sollen, ist anzunehmen,

daß es innerhalb der Streams viele Verbindungen zwischen Arealen gibt, aber zwischen den Streams nur wenige. Daraus ergibt sich die Idee, zu überprüfen, ob es Gruppen von Arealen im visuellen System gibt, zwischen denen es möglichst wenige Verbindungen bestehen. Da die Gesamtanzahl der Verbindungen in dem System konstant ist, bedeutet das gleichzeitig, daß die Anzahl der Verbindungen, die innerhalb der Gruppen liegen, möglichst groß ist.

Damit ergibt sich ein Optimierungsproblem. Wir suchen eine Aufteilung der Areale, die nicht als Start- oder Zielareal definiert wurden, in zwei disjunkte Mengen, so daß die Anzahl der Verbindungen zwischen den Mengen minimal wird. Formal sieht das Problem dann folgendermaßen aus:

Gegeben: Ein gerichteter Graph $G = (V, E)$, zwei Mengen $V_S, V_Z \subseteq V$ und eine Zahl $k \leq |V - (V_S \cup V_Z)|$.

Gesucht: Eine Aufteilung von $V - (V_S \cup V_Z)$ in zwei Mengen V_1, V_2 , $V_1 \cap V_2 = \emptyset$, $|V_1| = k$, mit $|\{(u, v) | u \in V_1, v \in V_2 \text{ oder } v \in V_1, u \in V_2\}|$ minimal.

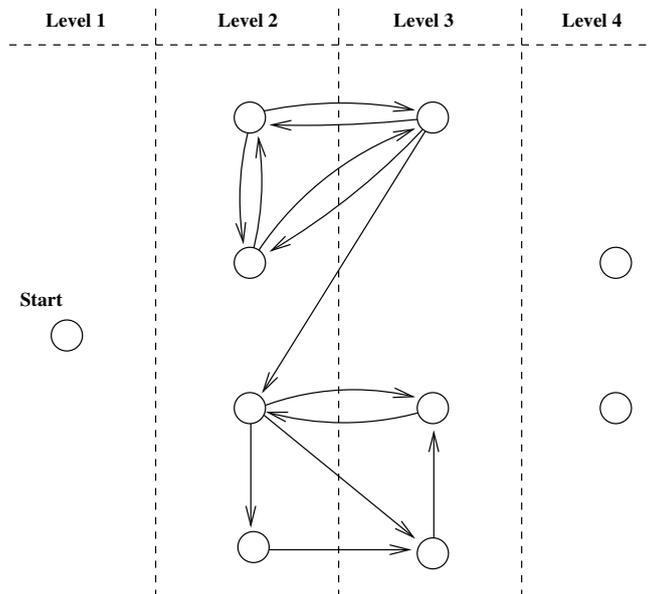


Abbildung 6.3: Der Beispielgraph aus Abbildung 6.2 nur mit den Verbindungen, die für die Suche nach Streams interessant sind. Verbindungen von und zu den Start- und Zielknoten wurden entfernt. Werden die drei oberen Knoten aus Level 2 und 3 zu einer Gruppe zusammengefaßt, dann gibt es nur eine Verbindung von dieser Gruppe zu den anderen vier Knoten auf Level 2 und 3. Bei allen anderen Aufteilungen dieser sieben Knoten gibt es mehr Verbindungen zwischen den Gruppen.

Diese Definition orientiert sich insofern an der Bewertung von Young, daß jede vorhandene Verbindung gezählt wird. Damit gehen auch hier reziproke Verbindungen doppelt so stark ein wie einfache. Die minimale Aufteilung muß für jede Gruppengröße k getrennt ermittelt werden, da sonst das Minimum aller Verbindungen zwischen den Gruppen immer bei $k = 0$ liegt. Für alle größeren k kann das Minimum größer als 0 sein. Die interessante Frage ist jedoch, ob sich das Minimum wieder verkleinert, wenn sich die Größen der beiden Gruppen annähern. Das wäre ein Indiz dafür, daß man Streams gefunden hat. Gibt es zwei Gruppen

von Knoten mit in sich geschlossener Verbindungsstruktur, für die es also kaum Verbindungen zwischen den Gruppen aber viele innerhalb der Gruppen gibt, so ist diese Gruppeneinteilung optimal. Erweitert man aber eine der Gruppen um einen Knoten aus der anderen Gruppe, so wird die Anzahl der Verbindungen zwischen den Gruppen steigen. Siehe dazu auch das Beispiel in Abbildung 6.3.

Es müssen aber nicht für alle $k \leq |V - (V_S \cup V_Z)| = n$ Aufteilungen bestimmt werden. Da sich die Areale frei auf die Gruppen verteilen können, reicht es, $k \leq \lceil \frac{n}{2} \rceil$ zu betrachten. Die Aufteilungen für größere k können daraus abgeleitet werden: Sei V_1^k eine optimale Gruppe V_1 für $k \leq n$. Dann ist

$$V_1^k = V - (V_S \cup V_Z \cup V_1^{n-k}).$$

Für jedes $k, k = 0, \dots, 13$, müssen jetzt die 26 Areale auf den oben definierten Leveln 2 und 3 (siehe Tabelle 6.2) in zwei Gruppen, eine mit k und die andere mit $26 - k$ Elementen, aufgeteilt werden. Um 26 Elemente in zwei Gruppen der Größe k aufzuteilen, gibt es $\binom{26}{k} = \frac{26!}{k!(26-k)!}$ Möglichkeiten. Für alle diese Einteilungen wird dann die Anzahl der Verbindungen zwischen den beiden Gruppen berechnet, um das Minimum dieser Werte (und damit eine optimale Aufteilung) zu bestimmen.

Im Rahmen dieser Untersuchung wurde für jedes dieser k und jede mögliche Gruppeneinteilung $V_1, V_2, V_1 \cap V_2 = \emptyset, |V_1| = k, |V_2| = 26 - k$, die Anzahl der Verbindungen zwischen V_1 und V_2 bestimmt und so für jedes k ein Minimum ermittelt. Das Ergebnis dieser Berechnung für die Verbindungsmatrix im visuellen System zeigt Tabelle 6.3. Dort ist die jeweils minimale Anzahl von Verbindungen abhängig von der Größe der beiden Mengen, also der Anzahl an Arealen, die jeder Stream enthält, aufgelistet. Zum Vergleich sind auch die maximale sowie die durchschnittliche Anzahl der Verbindungen bei diesen Gruppengrößen angegeben.

Gruppengrößen	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimale Anzahl Verbindungen	0	7	14	23	31	34	39
maximale Anzahl Verbindungen	0	36	64	84	101	117	128
durchschnittliche Anzahl Verbindungen	0	18.46	35.45	50.95	64.98	77.53	88.62

Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Anzahl Verbindungen	42	45	50	55	58	62	67
maximale Anzahl Verbindungen	136	144	151	155	156	157	156
durchschnittliche Anzahl Verbindungen	98.22	106.34	112.98	118.15	121.85	124.06	124.80

Tabelle 6.3: Anzahl von Verbindungen zwischen zwei Gruppen, abhängig von der Anzahl der Elemente in der Gruppe.

Wie man sieht, ergibt sich kein Minimum für zwei ungefähr gleich große Gruppen, sondern die minimale Zahl der Verbindungen nimmt beständig zu, wenn sich die Gruppengrößen annähern. Dies ist insbesondere bemerkenswert, weil die durchschnittliche Anzahl der Verbindungen bei ungefähr gleichgroßen Gruppen kaum noch zunimmt und das Maximum der Verbindungen zwischen den Gruppen sogar ab einer Gruppengröße von 10/16 stagniert. Es läßt sich also aus dieser Tabelle keine optimale Aufteilung oder Gruppengröße ableiten.

Deshalb wollen wir sehen, ob die genauen Gruppeneinteilungen, bei denen das Minimum erreicht wurden, einen Hinweis auf Streams liefern. In Tabelle 6.4 werden die optimalen Aufteilungen für ungefähr gleich große Gruppen aufgelistet. Diese sind am interessantesten, weil

wir ja davon ausgehen, daß die Streams jeweils aus einer vergleichbaren Anzahl von Arealen bestehen.

Größe der Gruppen	Anzahl Verbindungen	Aufteilung der Areale in Gruppen	
		V_1	V_2
10/16	55	V4, VOT, PITv, AITv, PITd, CITd, CITv, TF, TH, A46	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO, DP, FEF, A7a
		PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, V4, VOT, PO, DP
11/15	58	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a, VOT	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, V4, PO, DP
12/14	62	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a, V4, VOT	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO, DP
13/13	67	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a, V4, VOT, DP	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO

Tabelle 6.4: Gruppenaufteilungen für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl Verbindungen zwischen den Gruppen verläuft.

Für die Gruppengrößen 13/13, 12/14 und 11/15 wurde das Minimum nur von einer Einteilung angenommen. Für 10/16 gab es zwei Einteilungen mit minimaler Verbindungszahl zwischen den Gruppen. In Abbildung 6.4 sind diese Aufteilungen sowie das Ergebnis von Young eingezeichnet. Es gibt eine relativ große Übereinstimmung zwischen den Einteilungen, insbesondere verändern sich die Gruppen, die das Minimum bilden, für die verschiedenen Gruppengrößen kaum. Sie setzen sich nicht auf unterschiedlich Weise aus den 26 Arealen zusammen, sondern in der Regel unterscheiden sie sich die optimalen Aufteilungen mit einer um 1 unterschiedlichen Gruppengröße, nur durch ein Areal. Und das Verschieben von einem Areal ist natürlich schon zur Veränderung der Gruppengröße notwendig. Die Minima sind also mit einer bestimmten Art der Gruppeneinteilung assoziiert. Es fällt allerdings auf, daß die Einteilung eher zwischen Arealen auf unteren Hierarchiestufen und solchen auf oberen Hierarchiestufen trennt, und nicht parallel zum Hierarchiegefälle verläuft. Genau wie bei Young ist auch hier V4 das einzige Areal von Level 2, das sich auch mal in Gruppe V_1 befindet, alle anderen sind ausschließlich in Gruppe V_2 . Bei zwei Einteilungen enthält V_1 gar kein Areal von Level 2 und somit keine Verbindung vom Start- zu den Zielarealen. Bei der Aufteilung in 11/15 Areale und der zweiten Aufteilung in 10/16 Areale bilden die beiden Gruppen mit minimalen Verbindungen zwischen den Gruppen definitiv keine Streams, oder zumindest erfüllen sie unsere Anforderungen an Streams nicht. Es gibt dort in der Gruppe V_1 keine Verbindung von V_1 zu einem der Zielareale, da sie keine Areale der zweiten Level aus Tabelle 6.2 enthält. Somit kann diese Gruppe keine Informationen direkt von V_1 erhalten.

Trotz der Ähnlichkeit der Ergebnisse mit denen von Young sind also Zweifel angebracht, ob

V1 zu den Zielarealen STPp, AITd, STPa bilden. Vielleicht sollte man die Berechnung auf solche Gruppen beschränken, die das Grundmerkmal eines Streams erfüllen, nämlich daß sie jeweils eine Verbindung zwischen Start- und Zielarealen bilden. Allerdings ändert sich das Ergebnis kaum, wenn man als notwendige Bedingung nur Einteilungen zuläßt, die eine solche Verbindung garantieren. Das Ergebnis einer solchen Berechnung findet sich in Tabelle 6.6.

Aufteilung	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimal Anzahl Verbindungen	0	0	33	31	39	48	50
maximale Anzahl Verbindungen	0	0	60	84	101	117	128
durchschnittliche Anzahl Verbindungen	0	0	44.04	58.08	70.42	81.39	91.17

Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Anzahl Verbindungen	51	52	53	55	60	62	67
maximale Anzahl Verbindungen	136	144	151	155	156	157	156
durchschnittliche Anzahl Verbindungen	99.80	107.25	113.47	118.40	121.97	124.13	124.85

Tabelle 6.6: Anzahl von Verbindungen zwischen zwei Gruppen, in denen mindestens ein Weg vom V1 zu den Zielarealen führt, abhängig von der Anzahl der Elemente in der Gruppe.

Größe der Gruppen	Anzahl Verbindungen	Aufteilung der Areale in Gruppen	
		V ₁	V ₂
10/16	55	PITv, AITv, PITd, CITd, CITv, TF, TH, A46, V4, VOT	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO, DP, FEF, A7a
11/15	60	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, V4, VOT	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, A7a, PO, DP
12/14	62	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a, V4, VOT	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO, DP
13/13	67	PITv, AITv, PITd, CITd, CITv, TF, TH, FEF, A46, A7a, V4, VOT, DP	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO

Tabelle 6.7: Gruppeneinteilungen für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl Verbindungen zwischen den Gruppen verläuft und in denen eine Verbindung zwischen V1 und den Zielarealen besteht.

Auch mit dieser Einschränkung bildet sich kein eindeutiges Minimum für ungefähr gleich große Gruppen heraus. Insgesamt ändern sich die Werte wenig. In Tabelle 6.7 sind die Aufteilungen in Gruppen, bei denen jeweils das Minimum erreicht wurde, zu sehen. Auch die Einteilung der Gruppen unterscheidet sich kaum von der Einteilung ohne die Einschränkung auf zulässige Gruppen. Da ja aber auch schon vorher ein guter Teil der minimalen Gruppen die Verbindungseigenschaften hatte, war das auch nicht nicht erwarten.

In diesem Fall gibt es für $k = 10, \dots, 13$ jeweils nur eine Einteilung der Areale, die eine minimale Verbindungszahl zwischen den Gruppen besitzt. Bis auf die Einteilung für die Gruppengröße 11/15 entsprechen sie den Einteilungen, die wir auch schon ohne Einschränkung erhalten

hatten. Und in diesem Fall haben nur die Areale A7a und V4 ihre Gruppen getauscht im Vergleich zu der Bedingung ohne Einschränkung. In Abbildung 6.5 ist das Ergebnis noch einmal graphisch dargestellt.

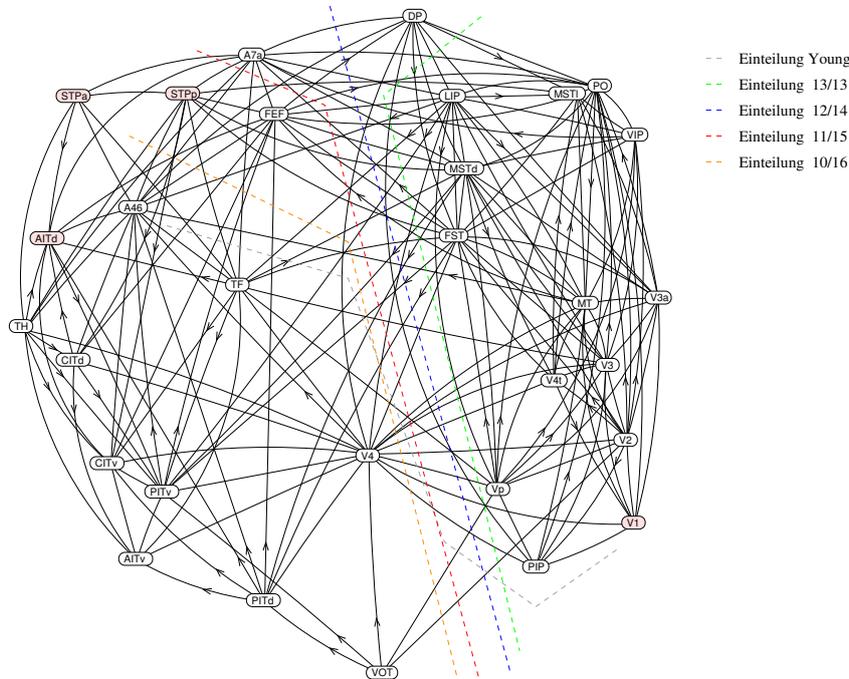


Abbildung 6.5: Einteilung des visuellen Systems, abhängig von der Anzahl der Verbindungen zwischen zwei Gruppen, in denen eine Verbindung zwischen V1 und den Zielarealen besteht.

Die optimalen Aufteilungen der Areale für $k = 10, \dots, 13$ auf die beiden Gruppen V_1 und V_2 sind noch etwas einheitlicher als zuvor. Es gibt jetzt nur noch drei Areale, die beim Ändern der Gruppengröße in eine andere Gruppe verschoben werden. Und das ist die minimale Anzahl, die notwendig ist, um die Veränderung der Gruppengröße zu kompensieren. In Tabelle 6.8 sind noch einmal die Areale aufgelistet, wie sie sich aufgrund der optimalen Aufteilungen für $k = 10, \dots, 13$ auf die beiden Gruppen V_1 und V_2 verteilen.

Aufteilung der Areale in Gruppen	
Areale in V_1	V4, VOT, AITv, PITv, PITd, CITd, CITv, TF, TH, A46
Areale in V_2	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO
Areale in V_1 und V_2	DP, FEF, A7a

Tabelle 6.8: Aufteilungen der Areale anhand der Einteilung für ungefähr gleich große Gruppen ($k = 10, \dots, 13$), bei denen eine minimale Anzahl Verbindungen zwischen den Gruppen verläuft und in denen es Wege von V1 zu den Zielarealen gibt.

Insgesamt bringt also eine Einschränkung der Gruppen auf solche, die eine Verbindung zwischen Start- und Zielarealen herstellen, keine deutliche Veränderung der Ergebnisse. Somit

hat uns dieser Ansatz auf der Suche nach einer Kennzeichnung von Streams nicht weitergeholfen. Insgesamt lassen sich Streams durch das einfache Zählen von Verbindungen nicht erkennen.

6.5.2 Definition über die Hamming-Distanz der Adjazenzen

Streams sollen jeweils durch eine ähnliche Verbindungsstruktur gekennzeichnet sein. Young mißt zum Beispiel nicht direkt Verbindungen, sondern in seiner Analyse wurden Areale, die Ähnlichkeit in Bezug auf ihre Verbindungen aufweisen, nahe beieinander platziert. Das führt zu der Frage, wie man eine Ähnlichkeit in der Verbindungsstruktur von Arealen messen kann. Die einfachste Möglichkeit ist, für zwei Areale zu zählen, zu welchen anderen Arealen nur eines von beiden eine Verbindung hat. Die Areale, bei denen man die wenigsten Unterschiede zählt, haben die größte Ähnlichkeit in ihrer Verbindungsstruktur. Die Unterschiede lassen sich durch die Hamming-Distanz auszudrücken. Die Hamming-Distanz ist eine Metrik, die häufig in der Codierungstheorie verwendet wird. Sie mißt die Anzahl der Unterschiede in zwei Zeichenketten oder Vektoren.

Definition 6.1 Sei F eine endliche Menge und $u = u_1 \dots u_n$, $v = v_1 \dots v_n$ Elemente in F^n . Dann heißt

$$d(u, v) = |\{i | u_i \neq v_i\}|$$

die Hamming-Distanz von u und v .

Die Hamming-Distanz definiert auf F^n eine Metrik, d.h. es gilt für alle $u, v, w \in F^n$

1. $d(u, v) \geq 0$ und $d(u, v) = 0$ genau für $u = v$,
2. $d(u, v) = d(v, u)$ und
3. $d(u, v) \leq d(u, w) + d(w, v)$

Die 3. Bedingung wird auch Dreiecksungleichung genannt.

Bei unserer Untersuchung ist nur wichtig, ob eine Verbindung vorhanden ist oder nicht, deshalb kommen wir mit der Menge $F = \{0, 1\}$ aus. Seien $x, y \in \{0, 1\}^k$, also $x = x_1 \dots x_k$ und $y = y_1 \dots y_k$ mit $x_i, y_i \in \{0, 1\}$ für $i = 1, \dots, k$. Dann ist die Hamming-Distanz von x und y definiert durch

$$hd(x, y) = \sum_{i=1}^n |x_i - y_i|.$$

So ist zum Beispiel für $x = 10011$ und $y = 11111$ die Hamming-Distanz $hd(x, y) = 2$.

Wenn wir die Zeilen oder Spalten der Adjazenzmatrix eines Graphen als Zeichenketten ansehen, können wir sie mit Hilfe der Hamming-Distanz vergleichen. Eine Zeile i in der Adjazenzmatrix gibt die Knoten an, zu denen der Knoten i auslaufende Kanten hat. Die Hamming-Distanz zweier Zeilen i, j gibt also an, zu wievielen Knoten nur i oder nur j eine auslaufende Kante hat. Damit ist sie ein Maß dafür, wie sehr sich die beiden Knoten in Bezug auf ihre auslaufenden Kanten unterscheiden. Analoges gilt für die Spalten der Adjazenzmatrix und die einlaufenden Kanten.

Sei $G = (V, E)$ ein gerichteter Graph mit Adjazenzmatrix $A_G = (a_{i,j})$ und $|V| = n$. Für $u_i, u_j \in V$ sei

$$hd_Z(u_i, u_j) = |\{a_{i,k} \neq a_{j,k} | k \in \{1, \dots, n\}, i \neq k \neq j\}|$$

die Hamming-Distanz auf den Zeilen und

$$hd_S(u_i, u_j) = |\{a_{k,i} \neq a_{k,j} | k \in \{1, \dots, n\}, i \neq k \neq j\}|$$

die Hamming-Distanz auf den Spalten der Adjazenzmatrix. Dabei werden keine Vergleiche mit den Werten auf der Diagonalen durchgeführt, da dieser Eintrag nicht definiert ist.

Damit erhält man also für jedes Paar von Knoten zwei Werte, welche die Anzahl der Unterschiede in ihren auslaufenden und die in ihren einlaufenden Kanten ausdrücken. Diese kann man addieren und erhält dadurch einen Wert, der den Gesamtunterschied der Verbindungsstrukturen dieser beiden Knoten zum Ausdruck bringt. Sei also

$$hd(u_i, u_j) = hd_Z(u_i, u_j) + hd_S(u_i, u_j).$$

Da sowohl hd_Z als auch hd_S die Dreiecksungleichung erfüllen, erfüllt auch hd die Dreiecksungleichung. Eine Metrik ist hd aber nicht, da aus $hd(u, v) = 0$ nicht $u = v$ folgt.

Eine andere Möglichkeit ist es, für die Vergleiche die Werte auf der Diagonalen entweder 0 oder 1 zu setzen und auch Vergleiche mit den Werten auf der Diagonalen zuzulassen. Dabei bewirkt eine 0, daß sich die Distanz erhöht, wenn die beiden Areale untereinander Verbindungen haben, und die 1, daß sich die Distanz erhöht, wenn die beiden Areale nicht, oder nicht in beiden Richtungen, verbunden sind. In diesem Zusammenhang ist es sinnvoller anzunehmen, daß zwei Areale sich ähnlicher sind, wenn sie auch Verbindungen miteinander haben, also die Werte der Diagonalen auf 1 zu setzen.

Sei $G = (V, E)$ ein gerichteter Graph mit Adjazenzmatrix $A_G = (a_{i,j})$, $a_{i,i} = 1$ für alle $i \leq |V|$, und $|V| = n$. Für $u_i, u_j \in V$ sei

$$hd_Z^1(u_i, u_j) = |\{a_{i,k} \neq a_{j,k} | k \in \{1, \dots, n\}\}|$$

die Hamming-Distanz auf den Zeilen und

$$hd_S^1(u_i, u_j) = |\{a_{k,i} \neq a_{k,j} | k \in \{1, \dots, n\}\}|$$

die Hamming-Distanz auf den Spalten der Adjazenzmatrix. In diesem Fall werden also auch Vergleiche mit der Diagonalen durchgeführt. Auch hier definieren wir eine Funktion, die die Werte verbindet. Sei also

$$hd^1(u_i, u_j) = hd_Z^1(u_i, u_j) + hd_S^1(u_i, u_j).$$

Da sowohl hd_Z^1 als auch hd_S^1 die Dreiecksungleichung erfüllen, erfüllt auch hd^1 die Dreiecksungleichung. In diesem Fall ist auch hd^1 eine Metrik.

Der Wert von hd und hd^1 wurde für jedes Arealpaar in der gerichteten Verbindungsmatrix nach Young (Tabelle A.1) berechnet. Die vollständigen Ergebnisse sind im Anhang in Tabelle B.1 und Tabelle B.2 zu finden. Da für jedes Arealpaar u, v gilt $hd(u, v) = hd(v, u)$ und $hd^1(u, v) = hd^1(v, u)$, sind die Tabellen jeweils symmetrisch zur Diagonalen. Mit Hilfe dieser Werte wollen wir jetzt Streams finden.

Sei also V die Menge der Areale. Zwei Areale u, v sind sich bezüglich ihrer Verbindungsstruktur ähnlich, wenn sie einen kleinen Wert $hd(u, v)$ und $hd^1(u, v)$ haben. Andersherum ausgedrückt, je größer der Wert von $hd(u, v)$ und $hd^1(u, v)$, desto unterschiedlicher sind die Verbindungen von u und v . Wir wollen zwei Gruppen in V finden, innerhalb derer die Summe der Werte von hd (und hd^1) für Areale, die beide zu derselben Gruppe gehören, möglichst klein ist. Da die Summe der Werte hd (und hd^1) für alle Arealpaare in V konstant ist, kann man die Summe der Werte innerhalb der Gruppen minimieren, indem man die Summe für solche Arealpaare, die sich in verschiedenen Gruppen befinden, maximiert. Damit erhalten wir das folgende Optimierungsproblem.

Gegeben: Eine Menge von Arealen V , zwei Mengen $V_S \subseteq V$ und $V_Z \subseteq V$, eine Zahl $k \leq |V - (V_S \cup V_Z)|$ und die Bewertungsfunktion $hd : V \times V \rightarrow \mathbb{N}$.

Gesucht: Eine Aufteilung von $V - (V_S \cup V_Z)$ in zwei Mengen V_1, V_2 , $V_1 \cap V_2 = \emptyset$, $|V_1| = k$, mit

$$\sum_{u \in V_1, v \in V_2} hd(u, v) \quad (\text{beziehungsweise} \quad \sum_{u \in V_1, v \in V_2} hd(u, v))$$

maximal.

Bei der Berechnung dieser Summen kann man wieder so vorgehen wie bei dem Zählen der Verbindungen zwischen den Gruppen im letzten Abschnitt. Die Ergebnisse sind in Tabelle 6.9 zu sehen.

Ergebnisse für die Funktion hd							
Aufteilung	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimale Summe	0	500	1012	1480	1920	2322	2686
maximale Summe	0	792	1460	2026	2550	2996	3408
durchschnittliche Summe	0	604.77	1161.16	1669.16	2128.79	2540.03	2902.89
Ergebnisse für die Funktion hd							
Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Summe	3012	3288	3524	3710	3844	3928	3958
maximale Summe	3790	4146	4442	4690	4812	4858	4828
durchschnittliche Summe	3217.37	3483.47	3701.19	3870.52	3991.48	4064.05	4088.24

Ergebnisse für die Funktion hd^1							
Aufteilung	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimale Summe	0	576	1146	1678	2156	2600	2998
maximale Summe	0	820	1546	2166	2758	3300	3774
durchschnittliche Summe	0	667.84	1282.26	1843.26	2350.82	2804.95	3205.66
Ergebnisse für die Funktion hd^1							
Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Summe	3350	3658	3912	4110	4256	4340	4378
maximale Summe	4218	4618	4946	5220	5356	5406	5370
durchschnittliche Summe	3552.94	3846.79	4087.22	4274.22	4407.78	4487.93	4514.64

Tabelle 6.9: Summe der Hamming-Distanzen zwischen den Elementen zweier Gruppen abhängig von der Anzahl der Elemente in der Gruppe

Auch hier ist es so, daß für beide Versionen der Hamming-Distanz die Summen wachsen, wenn sich die Gruppengrößen annähern. Da die Anzahl Werte, die aufsummiert werden, mit

k wachsen, ist dieses auch zu erwarten. Allerdings ist in diesem Fall das Maximum und nicht das Minimum interessant. Und das ist für beide Bewertungen bei einer Aufteilung von 12/14 am größten, und nicht bei 13/13. Das könnte einen Hinweis auf einen Stream mit einer Aufteilung 12/14 sein. Interessanterweise sind die Gruppeneinteilungen für $k = 10, \dots, 13$, bei den Gruppen, wo sich eine maximale Summe der Hamming-Distanz findet, für beiden Hamming-Distanz-Versionen dieselben, und für jede Gruppengröße gibt es auch nur eine optimale Aufteilung. Die Berücksichtigung der Diagonale bei der Berechnung der Distanzen verändert das Ergebnis also nicht. Eine Auflistung der optimalen Gruppen findet sich in Tabelle 6.10

Größe der Gruppen	Summe hd/hd^1	Aufteilung der Areale in Gruppen	
		V_1	V_2
10/16	4690/ 5220	PIT _v , AIT _v , PIT _d , CIT _d , CIT _v , TF, TH, A46, FEF, A7a	V2, V3, V3a, MT, MST _d , MST _l , FST, V _p , V4 _t , PIP, LIP, VIP, PO, DP, V4, VOT
11/15	4812/ 5356	PIT _v , AIT _v , PIT _d , CIT _d , CIT _v , TF, TH, FEF, A46, A7a, VOT	V2, V3, V3a, MT, MST _d , MST _l , FST, V _p , V4 _t , PIP, LIP, VIP, PO, DP, V4
12/14	4858/ 5406	PIT _v , AIT _v , PIT _d , CIT _d , CIT _v , TF, TH, FEF, A46, A7a, V4, VOT	V2, V3, V3a, MT, MST _d , MST _l , FST, V _p , V4 _t , PIP, LIP, VIP, PO, DP
13/13	4828/ 5370	V2, V3, V3a, MT, MST _d , MST _l , FST, V _p , V4 _t , PIP, LIP, VIP, PO	PIT _v , AIT _v , PIT _d , CIT _d , CIT _v , TF, TH, FEF, A46, A7a, V4, VOT, DP

Tabelle 6.10: Gruppeneinteilungen für ungefähr gleich große Gruppen mit maximaler Summe der Hamming-Distanzen

Auch bei dieser Methode der Einteilung unterscheiden sich die Gruppen mit maximaler Summe für $k = 10, \dots, 13$ kaum. Die optimale Einteilung 12/14 unterscheidet sich nur durch ein verschobenes Areal von den optimalen Aufteilungen für die Gruppengrößen 11/15 und 13/13. Für die Einteilungen 10/16 und 11/15 enthält die Gruppe V_1 kein Areal aus dem Level 2 von Tabelle 6.2. Damit können diese Einteilungen keine Streams sein. Die absolut optimale Einteilung mit 12/14 hingegen erfüllt die Bedingung für Streams, daß es in beiden Gruppen mindestens einen Weg vom Startareal zu den Zielarealen gibt. Eine graphische Darstellung dieser Ergebnisse ist in Abbildung 6.6 gegeben.

Auch hier wird deutlich, daß die Einteilung eher zwischen anterioren und posterioren Arealen als zwischen dorsalen und ventralen trennt. Dies ist sogar noch deutlicher der Fall als bei der Einteilung einfach nur nach der Anzahl der Verbindungen. Insbesondere scheint diese Einteilung zwischen Arealen auf niedrigen und hohen Hierarchiestufen zu trennen. Das steht aber im Widerspruch zu der Forderung, daß die Streams parallel zur Hierarchie verlaufen. Es ist zwar wiederum eine Ähnlichkeit mit den Ergebnissen von Young, aber kein dorsaler und ventraler Stream zu erkennen.

Die optimalen Aufteilungen der Areale für $k = 10, \dots, 13$ auf die beiden Gruppen V_1 und V_2 sind auch wieder sehr einheitlich. Es gibt auch hier nur drei Areale, die beim Ändern der

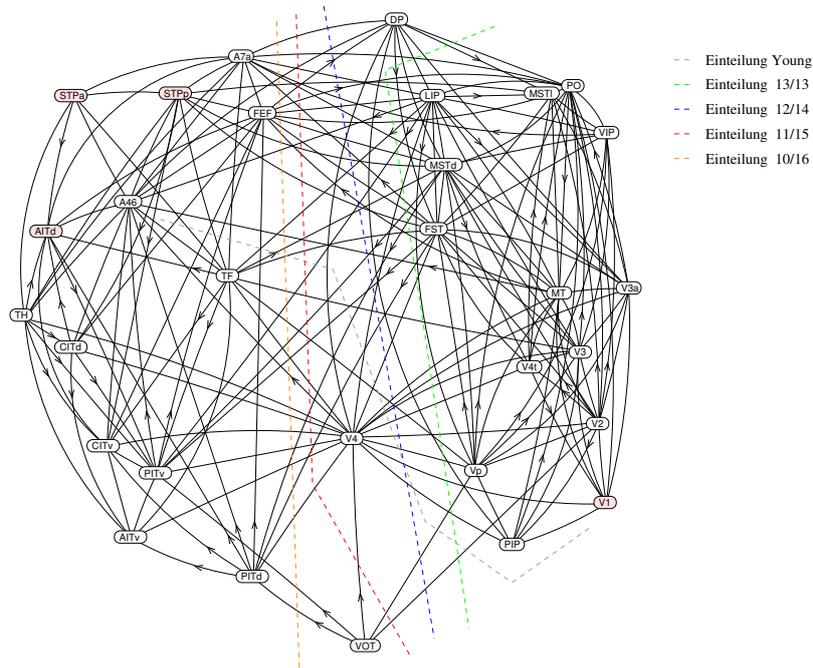


Abbildung 6.6: Einteilung des visuellen Systems abhängig von der Summe der Hamming-Distanzen zwischen den Arealen der zwei Gruppen

Gruppengröße in eine andere Gruppe verschoben werden. Eine Zuordnung der Arealen zu den beiden Gruppen findet sich in Tabelle 6.11.

Aufteilung der Arealen in Gruppen	
Areale in V_1	AITv, PITv, PITd, CITd, CITv, TF, TH, A46, FEF, A7a
Areale in V_2	V2, V3, V3a, MT, MSTd, MSTl, FST, Vp, V4t, PIP, LIP, VIP, PO
Areale in V_1 und V_2	V4, VOT, DP

Tabelle 6.11: Aufteilungen der Arealen anhand der Einteilung für ungefähr gleich große Gruppen, bei denen die Summe der Hamming-Distanzen minimal war.

Das Ergebnis ist auch insofern eindeutig, als es mit beiden Bewertungsfunktionen erreicht wurde. Trotz dieser Eindeutigkeit, und obwohl das absolute Maximum nicht bei der Gruppeneinteilung 13/13 liegt, erfüllen die gefundenen Gruppen nicht die Erwartungen, die an Streams gestellt werden. Auch dieser Ansatz hat leider nicht zu dem gewünschten Ergebnis einer Streamerkennung geführt.

6.5.3 Definition über kürzeste Wege

Da wir aber ja eigentlich Informationswege beschreiben wollen, ist es vielleicht etwas zu kurz gegriffen, bei der Einteilung der Arealen in zwei Gruppen nur Verbindungen oder Ähnlichkeit

ten in der Verbindungsstruktur zu betrachten. Die Annahme ist ja, daß Informationen auf parallelen Wegen in Streams verarbeitet werden. Die Information soll also vom Startareal zu einem der Zielareale gelangen, in der Regel innerhalb eines Streams. Deswegen ist eine weitere Möglichkeit zum Auffinden von Streams, Wege zu betrachten, die von V1 auf Level 1 zu den Arealen STPp, AITd und STPa auf Level 4 führen. Dabei gehen wir davon aus, daß die Information möglichst schnell weiter geleitet wird, und beschränken uns deshalb auf kürzeste Wege.

Wir suchen also jetzt eine Aufteilung der Areale in zwei disjunkte Mengen, so daß die Anzahl der kürzesten Wege, die durch Areale aus beiden Mengen laufen, minimal wird. Auch hier wird damit automatisch die Anzahl der Wege, die nur durch eine der Mengen laufen, maximiert. Um diese Mengen zu bestimmen, benötigen wir zuerst die Anzahl der Wege, die über jede Kante laufen, um diese dann als Einteilungskriterium zu verwenden. Wenn man diese Kantenbewertung hat, kann man anstelle der Anzahl der Kanten zwischen beiden Mengen wie in Abschnitt 6.5.1 die Summe der kürzesten Wege über die Kanten zählen.

Wenden wir uns also zunächst dem Problem zu, wie man die Anzahl der kürzesten Wege, die über eine Kante laufen, bestimmt. Dabei wollen wir nicht alle möglichen kürzesten Wege zählen, sondern nur die, die von einem festen Startknoten zu einem der vorgegebenen Zielknoten führen. Dabei sollen die Zielknoten alle dieselbe Entfernung vom Startknoten haben.

Wir verwenden wieder die in Abschnitt 6.4 für einen stark zusammenhängende gerichteten Graphen $G = (V, E)$ und einen Knoten $s \in V$ definierte Funktion $f_{s,G} : V \rightarrow \mathbb{N}_0$ mit

$$f_{s,G}(v) = \min\{k \mid s = u_1, \dots, u_k = v \text{ ist ein Weg von } s \text{ nach } v \text{ in } G\}.$$

Mit Hilfe dieser Funktion können wir jetzt das Problem definieren.

Gegeben sei ein gerichteter stark zusammenhängender Graph $G = (V, E)$ und ein Knoten $s \in V$ sowie die Menge

$$V' = \{v \in V \mid f_{s,G}(v) \geq f_{s,G}(u) \text{ für alle } u \in V\}.$$

Gesucht wird für jede Kante $e \in E$ die Anzahl der kürzesten Wege in G , die vom Startknoten s über e zu einem beliebigen Knoten in V' führen.

Alle diese kürzesten Wege haben die Länge $f_{s,G}(v) - 1$, $v \in V'$. Da wir kürzeste Wege suchen, kann jeder Weg nur durch einen Knoten auf jeder der durch $f_{s,G}$ bestimmten Leveln (vergleiche Abschnitt 6.4) führen. Das heißt, für jeden kürzesten Weg $s = u_1, \dots, u_k = v$, $v \in V'$ gilt $f_{s,G}(u_i) = i$ für $i = 1, \dots, k$. Umgekehrt ist jeder Weg, der aus jedem Level genau einen Knoten enthält, ein kürzester Wert vom Startknoten zu einem Zielknoten. Das heißt, wir brauchen nur Kanten zu betrachten, die zwischen Knoten auf unterschiedlichen Leveln verlaufen. Alle anderen Kanten erhalten den Wert 0.

Für einen Knoten $v \in V$ sei $\varepsilon(v)$ die Anzahl der kürzesten Wege, die von s zu v führen, $\alpha(v)$ die Anzahl der kürzesten Wege, die von v zu einem Zielknoten $v' \in V'$ führen. Für eine Kante $e \in E$ sei $\omega_E(e)$ die Anzahl der kürzesten Wege von s zu einem Zielknoten $v' \in V'$, die über e führen. Um die Anzahl der Wege zu ermitteln, die über eine Kante $e = (v_1, v_2)$ führen, muß man die Anzahl der Wege, die vom Startknoten zu v_1 führen, mit der Anzahl der Wege, die von v_2 zu einem Zielknoten führen, multiplizieren, also $\varepsilon(v)$ und $\alpha(v)$. Also müssen wir zuerst die Anzahl der Wege, die vom Startknoten zu einem beliebigen Knoten v hinführen, $\varepsilon(v)$, und die Anzahl der Wege, die von dem Knoten v zu einem der Zielknoten führen, $\alpha(v)$,

ermitteln. Multipliziert man $\varepsilon(v)$ mit $\alpha(v)$, so erhält man zusätzlich die Anzahl der Wege, die vom Startknoten über v zu einem Zielknoten führen.

Betrachten wir wieder den Beispielgraphen aus Abbildung 6.2. Da kürzeste Wege nur über Kanten zwischen Leveln laufen können, wurden alle anderen Kanten entfernt. Für die verbleibenden Kanten u, v gilt $f_{\text{Start},G}(u) + 1 = f_{\text{Start},G}(v)$. Der resultierende Graph ist in Abbildung 6.7 zu sehen.

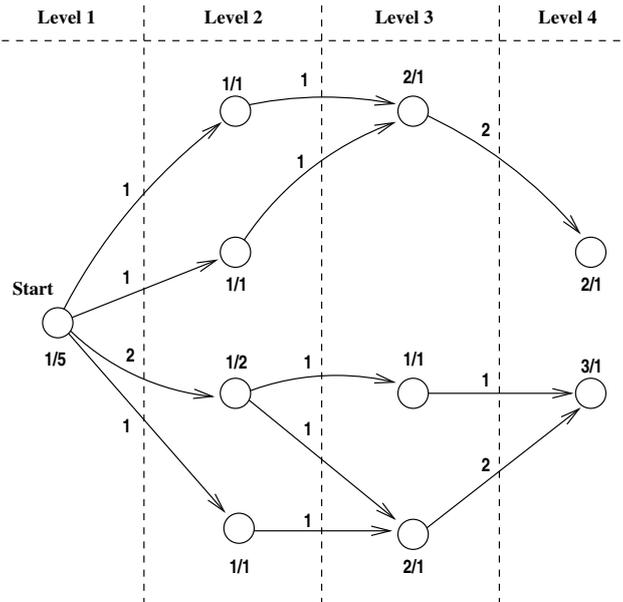


Abbildung 6.7: Beispiel für die Berechnung von $\varepsilon(v)$, $\alpha(v)$ und $\omega((u, v))$ für Knoten v und Kanten (u, v) von einem Graph mit Startknoten Start. Markierungen an den Knoten v : $\varepsilon(v)/\alpha(v)$. Markierungen an den Kanten (u, v) : $\omega((u, v)) = \varepsilon(u) \cdot \alpha(v)$.

Für einen Knoten v ist die Anzahl der Wege, die von s zu v führen, die Summe aller Wege von s zu denjenigen Knoten, von denen v einlaufende Kanten erhält. Entsprechend ist die Anzahl der Wege, die von v zum einem Zielknoten führen, die Summe aller Wege von Knoten, zu denen v auslaufenden Kanten hat, zu einem Zielknoten. Also

$$\varepsilon(v) = \sum_{(u,v) \in E} \varepsilon(u) \quad \text{und} \quad \alpha(v) = \sum_{(v,u) \in E} \alpha(u).$$

Offensichtlich ist die Anzahl der aus s auslaufenden Wege gleich der Summe der in allen Zielknoten einlaufenden Wege. Wir setzen $\varepsilon(s) = 1$ und $\alpha(v') = 1$ für alle $v' \in V'$. Dann können wir die Werte der anderen Knoten berechnen, indem wir die Knoten abhängig von ihrer Entfernung von s betrachten. Für die Berechnung von ε werden die Knoten in aufsteigender Entfernung von s und für die Berechnung von α in absteigender Entfernung von s betrachtet. Dazu benötigen wir einen Levelgraph, in dem zu jedem Knoten seine Entfernung vom Startknoten s bekannt ist.

Algorithmus Anzahl kürzester Wege

Eingabe: Levelgraph $G = (V, E)$, Knoten s

Ausgabe: Für jede Kante $(u, v) \in E$ die Anzahl der kürzesten Wege vom Startknoten zu einem Zielknoten, die über diese Kante führen.

für alle Knoten $u \in V$ {

$$\varepsilon(u) = 0$$

$$\alpha(u) = 0$$

}

$$\varepsilon(s) = 1$$

für $i = 1$ bis Levelanzahl-1 {

für jeden Knoten u auf Level i

für alle Kanten (u, w)

$$\text{wenn Level von } w > i \text{ dann } \varepsilon(w) = \varepsilon(w) + \varepsilon(u)$$

}

für alle Knoten u mit maximalen Level

$$\alpha(u) = 1$$

für $i = \text{Levelanzahl}$ herab bis 2 {

für jeden Knoten u auf Level i

für alle Kanten (w, u)

$$\text{wenn Level von } w < i \text{ dann } \alpha(w) = \alpha(w) + \alpha(u)$$

}

für alle Kanten $e = (u, w) \in E$ mit $\text{Level}(u) < \text{Level}(w)$

$$\omega_E(e) = \varepsilon(u) * \alpha(w)$$

Der Wert von ω wurde für die Verbindungen im visuellen System mit dem Startareal V1 berechnet. Da wir die Zielareale in Abschnitt 6.4 mit Hilfe einer Breitensuche bestimmt haben, stimmen sie mit den hier benötigten Zielknoten überein. Insgesamt gibt es 37 kürzeste Wege von V1 zu einem der drei Zielareale. Die Ergebnisse sind im Anhang in Tabelle B.3 zu finden. Damit erhält man eine Kantenbewertung $\omega_E : E \rightarrow \mathbb{N}_0$. Um Streams zu finden, möchte man nun die Anzahl der Wege zwischen den Gruppen minimieren.

Gegeben: Ein gerichteter Graph $G = (V, E)$, zwei Mengen $V_S \subseteq V$ und $V_Z \subseteq V$, eine Zahl $k \leq |V - (V_S \cup V_Z)|$ und die Kantenbewertung ω_E .

Gesucht: Eine Aufteilung von $V - (V_S \cup V_Z)$ in zwei Mengen V_1, V_2 , $V_1 \cap V_2 = \emptyset$, $|V_1| = k$, mit

$$\sum_{(u,v) \in E, u \in V_1, v \in V_2} \omega_E((u, v))$$

minimal.

Wie in den beiden vorherigen Abschnitten können wir also für $k = 0 \dots 13$ und alle möglichen Einteilungen in Gruppen der Größe k und $26 - k$ die Anzahl der Wege berechnen, die zwischen diesen beiden Gruppen verlaufen. Tabelle 6.12 zeigt für alle Gruppengrößen jeweils die minimale Anzahl der kürzesten Wege von V1 zu einem der Zielareale, die zwischen den Gruppen verlaufen.

Die Minima liegen fast alle bei 0, aber die Gesamtzahl der Wege ist ja auch relativ klein.

Aufteilung	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimale Anzahl Wege	0	0	0	0	0	0	0
maximale Anzahl Wege	0	13	21	26	31	34	37
durchschnittliche Anzahl Wege	0	2.85	5.46	7.86	10.2	11.95	13.66

Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Anzahl Wege	0	0	0	1	2	3	4
maximale Anzahl Wege	37	37	37	37	37	37	37
durchschnittliche Anzahl Wege	15.14	16.39	17.42	18.22	18.78	19.13	19.24

Tabelle 6.12: Minimale Anzahl der kürzesten Wege von V1 zu einem der Zielareale, die zwischen den Gruppen verlaufen, sortiert nach Gruppengröße.

Allerdings gibt es trotzdem kein globales Minimum bei ungefähr gleichen Gruppengrößen, sondern wenn sich die Gruppen in der Größe annähern, wächst auch das Minimum. Es erreicht erst bei einer Verteilung 10/16 einen Wert größer als 0, wächst dann aber mit jedem weiteren Element um 1. Das Maximum wird hingegen schon bei einer Einteilung von 6/20 erreicht. Dabei befinden sich auf Level 2 und auf Level 3 jeweils mehr als 6 Areale. Das bedeutet, daß es Areale gibt, die auf keinem kürzesten Weg liegen. Tabelle 6.13 zeigt Gruppeneinteilungen, bei denen eine minimale Anzahl von kürzesten Wegen von V1 zu einem der Zielareale zwischen den beiden Gruppen verläuft und bei denen die Gruppen sich in ihrer Größe nicht unterscheiden.

Hier gibt es keine eindeutigen Gruppeneinteilungen, sondern für alle Gruppengrößen mit $k = 10, \dots, 13$ immer mehrere Möglichkeiten, die sich allerdings kaum von einander unterscheiden. Dies wird auch in Abbildung 6.8 deutlich.

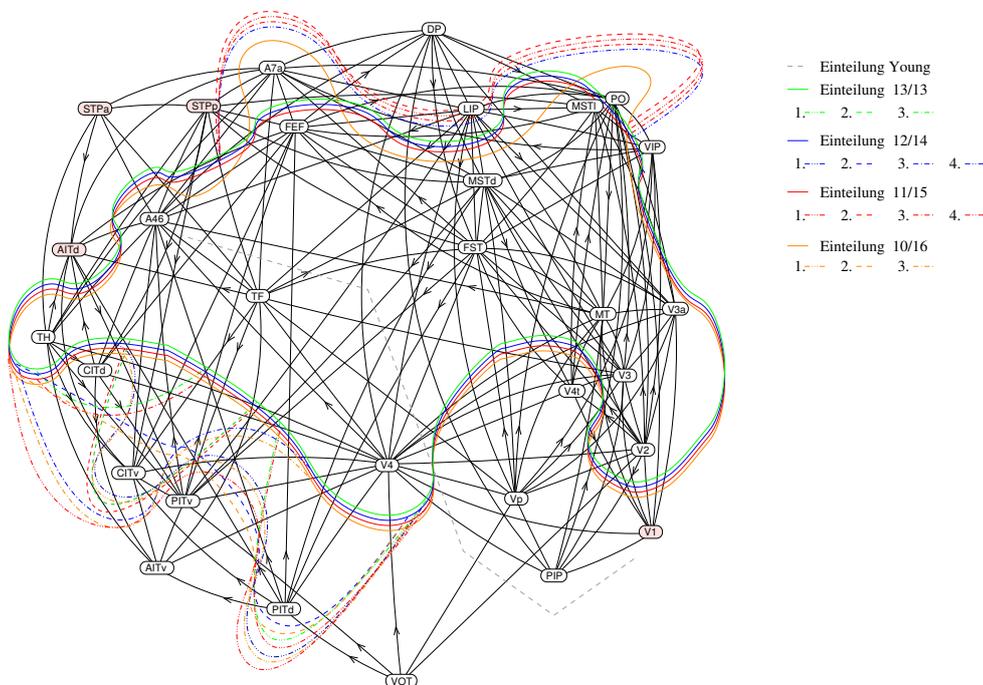


Abbildung 6.8: Einteilung des visuellen Systems, abhängig von der Anzahl der kürzesten Wege zwischen zwei Gruppen.

Größe der Gruppen	Anzahl Wege	Aufteilung der Areale in Gruppen	
		V_1	V_2
10/16	1	PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITd	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, CITv, PO, A7a
		PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, CITd, PO, A7a
		PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PITd	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, TF, TH, FEF, A46, CITd, CITv, PO, A7a
11/15	2	PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, CITd, CITv
		PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITd, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, PO, A7a
		PITd, PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, CITd, TF, TH, FEF, A46, PO, A7a
		PITd, PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITd	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, CITv, TF, TH, FEF, A46, PO, A7a
12/14	3	PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, CITd
		PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITd	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46, CITv
		AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITd, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, PITv, TF, TH, FEF, A46
		PITd, PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, CITd, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PO, TF, TH, FEF, A46, A7a
13/13	4	PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITd, PITd	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, CITv, TF, TH, FEF, A46
		PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITd, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, PITd, TF, TH, FEF, A46
		PITd, PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP, PO, A7a, CITv	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, CITd, TF, TH, FEF, A46

Tabelle 6.13: Gruppenaufteilungen für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl kürzester Wege zwischen den Gruppen verläuft.

Dafür unterscheiden sich diese Aufteilungen doch deutlich von denen, die in den vorigen beiden Abschnitten gefunden wurden. Leider haben diese Aufteilungen den Nachteil, daß V_2 zwar immer ein Areal aus Level 2 enthält (PIP), aber über dieses Areal kein kürzester Weg läuft. Das heißt, wenn nicht zusätzlich ein weiteres Areal von Level 2 in V_2 ist, dann gibt es keinen kürzesten Weg, der durch V_1 führt, was V_1 als Stream ungeeignet macht. In einigen Aufteilungen ist zusätzlich das Areal PO in V_1 , dann führen aber nur 3 der 37 kürzesten Wege durch die Knotengruppe V_1 . Wenn also die Anzahl der kürzesten Wege, die innerhalb einer Gruppe verlaufen, einen Stream definierten, dann kann V_1 kein Stream sein.

Trotz der relativ vielen verschiedenen Gruppeneinteilungen, die zu einer minimalen Anzahl der kürzesten Wege zwischen Gruppen führen, gibt es nur wenige Areale, die sich für $k = 10, \dots, 13$ nicht immer in derselben Gruppe befinden. Einen Überblick gibt Tabelle 6.14.

Aufteilung der Areale in Gruppen	
Areale in V_1	PITv, AITv, Vp, VOT, V4t, PIP, LIP, VIP, DP,
Areale in V_2	V2, V3, V3a, MT, MSTd, MSTl, FST, V4, TF, TH, FEF, A46
Areale in V_1 und V_2	CITd, CITv, PITd, PO, A7a

Tabelle 6.14: Aufteilungen der Areale anhand der Einteilung für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl kürzester Wege zwischen den Gruppen verläuft.

Es läßt sich also auch auf diese Art kein Stream finden, was auch an der relativ geringen Anzahl von kürzesten Wegen, die von V_1 durch den Graphen laufen, liegen kann. So ist die Aufteilung nur von wenigen Knoten abhängig. Insgesamt gibt es neun Areale, über die überhaupt keine kürzesten Wege laufen. Deshalb gibt es auch bis zu einer Gruppengröße von 9/17 Aufteilungen, bei denen keine kürzesten Wege zwischen den Gruppen verlaufen.

6.5.4 Definition über kürzeste und um eins längere Wege

Da die Information nicht unbedingt auf dem kürzesten Weg weitergeleitet wird, ergibt es Sinn, auch längere Wege zu betrachten. Wir wollen jetzt für die Analyse noch Wege hinzunehmen, die auch eine Kante innerhalb einer Level verwenden, also eine Länge haben, die um eins länger ist als die der kürzesten Wege. Man könnte natürlich auch noch längere Wege betrachten, aber Wege, die um zwei länger sind als kürzeste Wege, können Kreise enthalten, was nicht sinnvoll für die Informationsweiterleitung ist. Deshalb werden wir hier nur Wege hinzunehmen, die um eins länger sind als kürzeste Wege. Die Berechnung dieser Wege ist ähnlich der Berechnung für die kürzesten Wege.

Sei $G = (V, E)$ ein Graph und $f_{s,G}, \varepsilon, \alpha$ und ω definiert wie im letzten Abschnitt. Sei $max = \max\{f_{s,G}(v) - 1 | v \in V\}$. Für einen Knoten $v \in V$ sei $\varepsilon_1(v)$ die Anzahl der Wege der Länge $max + 1$, die von s zu v führen, $\alpha_1(v)$ die Anzahl der Wege der Länge $max + 1$, die von v zu einem Zielknoten $v' \in V'$ führen, und für eine Kante $e \in E$ sei $\omega_1(e)$ die Anzahl der Wege mit Länge $max+1$, die über e führen. Sei $(u, v) \in E$. Dann kann nur dann $\omega_1((u, v)) > 0$ sein, wenn $f_{s,G}(u) \leq f_{s,G}(v)$, da sonst der Weg von s zu einem Zielareal länger ist als $max+1$. Wir können uns bei der Berechnung also auf Kanten $(u, v) \in E$ mit $f_{s,G}(u) \leq f_{s,G}(v)$ beschränken. Wege, die die Länge $max + 1$ haben, enthalten genau eine Kante (u, v) , für die gilt $f_{s,G}(u) = f_{s,G}(v)$.

Das heißt, bis auf diese eine Kante entspricht der Verlauf eines solchen Weges dem Verlauf der kürzesten Wege in dem Graphen. Es gilt

$$\omega_1((u, v)) = \begin{cases} \varepsilon_1(u) \cdot \alpha(v) + \varepsilon(u) \cdot \alpha_1(v) & \text{wenn } f_{s,G}(u) < f_{s,G}(v), \\ \varepsilon(u) \cdot \alpha(v) & \text{wenn } f_{s,G}(u) = f_{s,G}(v). \end{cases}$$

Ein Beispiel für die Berechnung von ω_1 ist in Abbildung 6.9 zu sehen. Es handelt sich wieder um den Graph aus Abbildung 6.2, aus dem jedoch alle Kanten (u, v) mit $f_{s,G}(u) > f_{s,G}(v)$ entfernt wurden.

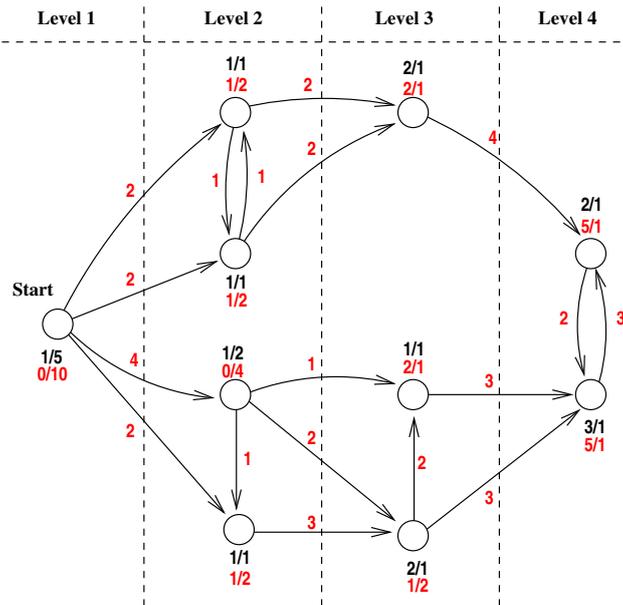


Abbildung 6.9: Beispiel für die Berechnung von $\varepsilon(v)$, $\alpha(v)$, $\varepsilon_1(v)$, $\alpha_1(v)$ sowie $\omega_1((u, v))$ für Knoten v und Kanten (u, v) von einem Graph mit Startknoten Start. Markierungen an den Knoten v in schwarz: $\varepsilon(v)/\alpha(v)$ und in rot: $\varepsilon_1(v)/\alpha_1(v)$. Markierungen an den Kanten (u, v) : $\omega_1((u, v))$.

Die Wege, die zwischen Start- und Zielarealen verlaufen und um 1 länger sind als kürzeste Wege, kann man mit einem ähnlichen Algorithmus berechnen wie die kürzesten Wege. Die Werte von ε_1 und α_1 werden im Gegensatz zu den Werten von ε und α alle mit 0 initialisiert, auch die von s und den Knoten v' mit $v' \in V'$. Dies ist sinnvoll, da es zwar einen kürzesten Weg der Länge 0 von diesen Knoten zu sich selbst gibt, aber keinen der Länge 1.

Algorithmus Anzahl kürzester Wege + 1

Eingabe: Levelgraph $G = (V, E)$ mit Startknoten s

Ausgabe: Für jede Kante die Anzahl der kürzesten und um 1 längeren Wege vom Starknoten zu einem Zielknoten, die über diese Kante führen.

für alle Knoten $u \in V$ {
 $\varepsilon(u) = 0$

$$\begin{aligned} & \varepsilon_1(u) = 0 \\ & \alpha(u) = 0 \\ & \alpha_1(u) = 0 \} \\ \varepsilon(s) = 1 \\ \text{für alle Knoten } u \text{ mit maximalen Level} \\ & \alpha(u) = 1 \\ \text{für } i = 1 \text{ bis Levelanzahl-1 } \{ \\ & \text{für jeden Knoten } u \text{ auf Level } i \\ & \text{für alle Kanten } (u, w) \text{ mit Level von } w = i \\ & \quad \varepsilon_1(w) = \varepsilon_1(w) + \varepsilon(u) \\ & \text{für alle Kanten } (u, w) \text{ mit Level von } w > i \{ \\ & \quad \varepsilon(w) = \varepsilon(w) + \varepsilon(u) \\ & \quad \varepsilon_1(w) = \varepsilon_1(w) + \varepsilon_1(u) \\ & \quad \} \\ & \} \\ \text{für } i = \text{Levelanzahl herab bis } 2 \{ \\ & \text{für jeden Knoten } u \text{ auf Level } i \\ & \text{für alle Kanten } (w, u) \text{ mit Level von } w = i \\ & \quad \alpha_1(w) = \alpha_1(w) + \alpha(u) \\ & \text{für alle Kanten } (w, u) \text{ mit Level von } w < i \{ \\ & \quad \alpha(w) = \alpha(w) + \alpha(u) \\ & \quad \alpha_1(w) = \alpha_1(w) + \alpha(u) \\ & \quad \} \\ & \} \\ \text{für alle Kanten } e = (u, w) \in E \text{ mit Level}(u) \leq \text{Level}(w) \\ \text{wenn Level}(u) < \text{Level}(w) \text{ dann } \{ \\ & \quad \omega(e) = \varepsilon(u) \cdot \alpha(w) \\ & \quad \omega_1(e) = \varepsilon(u) \cdot \alpha_1(w) + \varepsilon_1(u) \cdot \alpha(w) \\ & \quad \} \\ \text{else } \omega(e) = \varepsilon(u) \cdot \alpha(w) \end{aligned}$$

Damit erhält man eine Kantenbewertung $\omega_1 : E \rightarrow \mathbb{N}$. Diese wurde für die Verbindungsmatrix des visuellen System berechnet. Die Ergebnisse befinden sich im Anhang in Tabelle B.4. Um Streams zu finden, möchte man nun die Anzahl der Wege zwischen zwei Gruppen minimieren, die entweder die Länge max oder die Länge $max + 1$ haben.

Gegeben: Ein gerichteter Graph $G = (V, E)$, zwei Mengen $V_S \subseteq V$ und $V_Z \subseteq V$, eine Zahl $k \leq |V - (V_S \cup V_Z)|$ und die Kantenbewertung $w_+ = \omega + \omega_1$.

Gesucht: Eine Aufteilung von $V - (V_S \cup V_Z)$ in zwei Mengen V_1, V_2 , $V_1 \cap V_2 = \emptyset$, $|V_1| = k$, mit

$$\sum_{(u,v) \in E, u \in V_1, v \in V_2} w_+((u, v))$$

minimal.

Dabei ist es auch möglich, daß ein kürzester Weg der Länge $max + 1$ zweimal von der einen Gruppe zur anderen kreuzt, also auch zweimal gezählt wird. Die Anzahl der Wege der Länge

$max + 1$ im gesamten Graph entspricht wieder der Anzahl der Wege über jeden Schnitt zwischen den Leveln, und somit insbesondere der Anzahl von Wegen, die aus s auslaufen.

Um die Bewertungsfunktion w_+ für das visuelle System zu erhalten, wurden die Werte aus Tabelle B.3 und B.4 addiert. Das Ergebnis ist in Tabelle B.5 im Anhang zu finden. Die Gesamtzahl der kürzesten und um 1 längeren Wege im visuellen System ist 608. In diesem Fall gibt es keinen Knoten, über den nicht mindestens einer der zur Einteilung verwendeten Wege läuft. Aus diesen Daten wurden dann die optimale Gruppeneinteilungen für $k = 0, \dots, 13$ berechnet. Tabelle 6.15 zeigt für alle Gruppengrößen jeweils die minimale Anzahl der kürzesten und um 1 längeren Wege von V1 zu einem der Zielareale, die bei optimaler Aufteilung zwischen den Gruppen verlaufen.

Aufteilung	0/26	1/25	2/24	3/23	4/22	5/21	6/20
minimale Anzahl Wege	0	2	14	28	44	64	85
maximale Anzahl Wege	0	272	437	563	642	692	731
durchschnittliche Anzahl Wege	0	88.15	169.26	243.30	310.30	370.25	423.14

Aufteilung	7/19	8/18	9/17	10/16	11/15	12/14	13/13
minimale Anzahl Wege	110	149	194	234	282	326	352
maximale Anzahl Wege	752	772	783	785	785	785	777
durchschnittliche Anzahl Wege	468.98	507.77	539.50	564.18	581.82	592.39	595.92

Tabelle 6.15: Minimale, maximale und durchschnittliche Anzahl der kürzesten und um 1 längeren Wege von V1 zu einem der Zielareale, die zwischen den Gruppen verlaufen, sortiert nach Gruppengröße.

Es läßt sich auch hier kein eindeutiges Minimum der Anzahl der Wege, die zwischen den Gruppen verlaufen, für etwa gleich große Gruppen erkennen. Es wächst mit k stetig und ist am Ende mehr als halb so groß wie die Gesamtzahl der Wege im Graphen und auch fast halb so groß wie das Maximum der Wege zwischen zwei Gruppen. Dagegen stagniert das Maximum für die Gruppengrößen 10/16, 11/15 und 12/14 und nimmt für die Gruppengröße 13/13 sogar leicht ab. Interessant ist auch, daß die maximale Anzahl der Wege zwischen den Gruppen größer als die maximale Anzahl von Wegen in dem Graphen ist. Bei ungünstigen Gruppeneinteilungen gibt es also Wege, die zweimal zwischen den Gruppen wechseln.

Tabelle 6.16 zeigt die Gruppeneinteilungen, bei denen eine minimale Anzahl von kürzesten und um 1 längeren Wegen von V1 zu einem der Zielareale zwischen den beiden Gruppen verläuft und bei denen die Gruppen ungefähr gleich viele Areale enthalten.

In dieser Bedingung gibt es bis auf die Aufteilung 13/13 wieder nur eine Möglichkeit für eine optimale Aufteilung bei ungefähr gleich großen Gruppen. Eine Darstellung der Ergebnisse ist in Abbildung 6.10 zu sehen. Das Ergebnis ist in dieser Hinsicht tatsächlich eindeutiger als bei der Differenzierung nur nach den kürzesten Wegen.

Die Aufteilungen scheinen irgendwo zwischen denen der Aufteilung nach der Anzahl der Verbindungen und denen der Aufteilung nach den kürzesten Wegen zu liegen. Auffällig ist, daß sich die Aufteilungen für die einzelnen Gruppengrößen untereinander deutlich unterscheiden. Es gibt trotz der größtenteils eindeutigen optimalen Aufteilungen für die verschiedenen Gruppengrößen viele Areale, die sich für $k = 10, \dots, 13$ nicht immer in derselben Gruppe befinden. Einen Überblick gibt Tabelle 6.17.

Größe der Gruppen	Anzahl Wege	Aufteilung der Areale in Gruppen	
		V_1	V_2
10/106	234	PITv, AITv, PITd, CITd, CITv, TH, VOT, PIP, V4t, Vp	V2, V3, V3a, MT, MSTd, MSTl, FST, PO, LIP, A7a, FEF, V4, TF, DP, A46, VIP
11/15	282	PITv, AITv, PITd, CITd, CITv, TH, VOT, PIP, V4t, Vp, VIP	V2, V3, V3a, MT, MSTd, MSTl, FST, PO, LIP, A7a, FEF, V4, TF, DP, A46
12/14	326	PITv, AITv, PITd, CITd, CITv, TH, VOT, PIP, V4t, V4, DP, A46	V2, V3, V3a, MT, MSTd, MSTl, FST, PO, LIP, A7a, FEF, Vp, TF, VIP
13/13	352	PITv, AITv, PITd, CITd, CITv, TH, VOT, PIP, Vp, V4, DP, A46, TF	V2, V3, V3a, MT, MSTd, MSTl, FST, PO, LIP, A7a, FEF, V4t, VIP
		PITv, AITv, PITd, CITd, CITv, TH, VOT, PIP, Vp, V4, V4t, A46, TF	V2, V3, V3a, MT, MSTd, MSTl, FST, PO, LIP, A7a, FEF, DP, VIP

Tabelle 6.16: Gruppenaufteilungen für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl kürzester und um 1 längerer Wege zwischen den Gruppen verläuft.

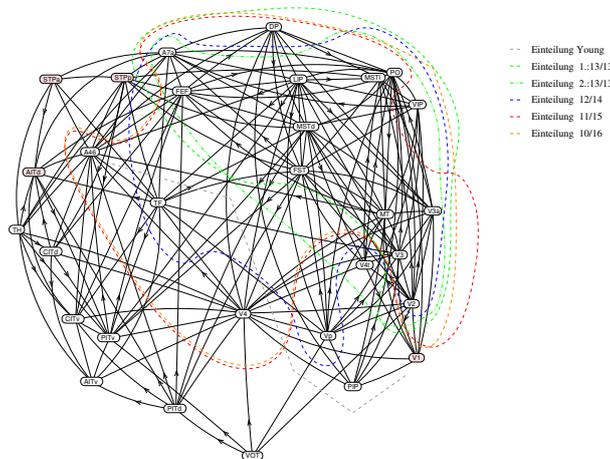


Abbildung 6.10: Einteilung des visuellen Systems abhängig von der Anzahl der kürzesten und um 1 längeren Wege zwischen zwei Gruppen.

Auch hier findet sich kein Ergebnis, aus dem man eindeutig einen Stream bestimmen kann. Zum einen deuten die Zahlen darauf hin, daß auch bei den optimalen Aufteilungen für ungefähr gleich große Gruppengrößen eine große Anzahl der Wege nicht innerhalb, sondern zwischen den Gruppen verlaufen. Außerdem ist die Anzahl der Areale, die bei ungefähr gleich großen Gruppen nicht derselben Gruppe zugeordnet werden, größer als bei den anderen bisher vorgestellten Verfahren.

Aufteilung der Areale in Gruppen	
Areale in V_1	PIT _v , AIT _v , PIT _d , CIT _d , CIT _v , TH, VOT, PIP
Areale in V_2	V2, V3, V3a, MT, MST _d , MST _l , FST, PO, LIP, A7a, FEF
Areale in V_1 und V_2	V4, DP, V4t, V _p , VIP, TF, A46

Tabelle 6.17: Aufteilungen der Areale anhand der Einteilung für ungefähr gleich große Gruppen, bei denen eine minimale Anzahl kürzester und um eins längerer Wege zwischen den Gruppen verläuft.

6.6 Diskussion

Keiner der hier vorgestellten Versuche, Streams mit Hilfe der Verbindungsstruktur zu definieren, hat ein klares Ergebnis geliefert. Es lassen sich mit den verwendeten Definitionen und Methoden also keine Streams auf topologischer Ebene nachweisen. Leider gibt Young auch keine exakte Definition, was er unter einem Stream versteht. Damit konnte das Ergebnis von Young, daß sich innerhalb des visuellen Systems auch topologische Streams nachweisen lassen, nicht bestätigt werden.

Vielleicht erlangt man neue Erkenntnisse, wenn man nicht nur die Verbindungsstruktur, sondern auch die einzelnen Verbindungsstärken berücksichtigt. Allerdings läßt die Datenlage dieses im Moment nicht zu. So ist nur für einen Teil der Verbindungen im visuellen System ihre Stärke bekannt.

Wenn wir die Ergebnisse der verschiedenen Verfahren zusammenfassen, dann ergibt sich die folgende Tabelle 6.18. Dort wurden diejenigen Areale zu Gruppen zusammengefaßt, die bei allen Verfahren bei den Ergebnissen für gleich große Gruppen immer in derselben Gruppe waren.

Aufteilung der Areale in Gruppen	
Areale in V_1	AIT _v , PIT _v
Areale in V_2	V2, V3, V3a, MT, MST _d , MST _l , FST
Areale in V_1 und V_2	V4, VOT, DP, FEF, A7a, V _p , V4t, PIP, LIP, VIP, PO, PIT _d , CIT _d , CIT _v , TF, TH, A46

Tabelle 6.18: Aufteilungen der Areale anhand der Einteilung für ungefähr gleich große Gruppen ($k = 10, \dots, 13$) nach den verschiedenen Verfahren

Wie man sieht, gibt es eine Gruppe von sieben Arealen, die immer in Gruppe V_2 zu finden war. Diese müssen eine große Ähnlichkeit bezüglich ihrer Verbindungseigenschaften aufweisen. Es handelt sich dabei um posteriore Areale. Die Gruppe mit Arealen aus V_1 ist deutlich kleiner, sie enthält nur zwei Areale. Dies liegt auch daran, daß V_1 nie mehr Elemente enthielt als V_2 und somit in der Regel die kleinere Gruppe war. Die meisten Areale lassen sich jedoch keiner

der beiden Gruppen zuordnen, sondern befinden sich je nach Methode mal in der einen und mal in der anderen Gruppe.

Kapitel 7

Das Parzellierungsproblem

7.1 Einführung

Die Kartierung des Gehirns mit seinen vielen strukturellen und funktionellen Aspekten bildet die Grundlage der Bemühungen, zu verstehen, wie sich das Gehirn entwickelt, wie es arbeitet und sich anpaßt. Besonders komplex ist die Architektur und funktionelle Organisation des zerebralen Cortex, der seit mehr als 100 Jahren untersucht wird (siehe zum Beispiel Meynert [Mey68] oder Brodmann, [Bro05, Bro09]). Dabei entstand im Laufe der Zeit wegen der Vielzahl angewandter Methoden und der hohen Beobachterabhängigkeit vieler mikrostrukturell-funktionaler Kriterien eine große Anzahl von Parzellierungsschemata (auch Hirnkarten genannt), die sich in Nomenklatur, Anzahl, Position, Größe, Grenzen und Darstellung der beschriebenen Gehirnstrukturen unterscheiden (vergleiche Abschnitt 2.1). Bislang konnte man sich, auch aufgrund der verschiedenen Zielsetzungen und Methoden unterschiedlicher Studien, auf kein einheitliches Parzellierungsschema einigen [KW05].

Die in Studien gewonnenen Daten wurden lange Zeit in Form von Gehirnregionen und Karten beschrieben, die durch mikrostrukturelle, makroskopische und funktionale Eigenschaften definiert werden. Seit einigen Jahren jedoch werden Daten auch häufig in Bezug auf räumliche Koordinatensysteme erfaßt. Damit gibt es jetzt zusätzlich zu den vielen verschiedenen Parzellierungsschemata zwei grundsätzlich unterschiedliche Methoden, diese zu erfassen: ein koordinatenbasiertes System und ein koordinatenunabhängiges System.

Der Vergleich von Daten aus zwei verschiedenen Hirnkarten kann schwierig sein, da oft nicht klar ist, in welchem Verhältnis Areale aus verschiedenen Studien zueinander stehen. Es wurden zwar wiederholt umfangreiche Aufstellungen zum Vergleich von verschiedenen Karten, Arealdefinitionen und verwandten Terminologien erstellt (z.B. [FE91, You93]), die ja auch in den anderen Kapiteln dieser Arbeit verwendet wurden. Aber im Moment erfordert dieses Handarbeit von menschlichen Experten, die die Ergebnisse aus Studien systematisch zusammenstellen und analysieren.

Es besteht also Bedarf, die Areale in den verschiedenen Hirnkarten zueinander in Beziehung zu setzen, um die angesammelten Daten zu evaluieren, zu vergleichen und besser zu bewerten. Zusätzlich möchte man natürlich auch koordinatenunabhängig erfaßte Areale mit räumlichen Daten in Bezug bringen, damit man auch Informationen, die in den verschiedenen Systemen registriert wurden, zusammenführen kann. Ein Ansatz, das erste Problem zu

lösen, wurde schon in [KW05] vorgestellt. Dort wurden mit Hilfe schon bekannter Arealbeziehungen aus verschiedenen Parzellierungsschemata neue Arealbeziehungen hergeleitet. Dieses Vorgehen hat den Schwachpunkt, daß schon Beziehungen bekannt sein müssen, um neue zu erstellen. Je weniger Beziehungen bekannt sind, desto weniger neue kann man finden. Ein Ansatz, die zweite Aufgabe anzugehen, existiert in der CARET-Software, in der einige koordinatenunabhängige Parzellierungsschemata auf einen Koordinatenraum übertragen wurden ([Ess04, DDE01]). Diese Darstellung soll hier verwendet werden, um Beziehungen zwischen Regionen der abgebildeten Karten festzustellen.

In diesem Kapitel soll also ein Verfahren vorgestellt werden, mit dem sich logische Beziehungen zwischen kortikalen Arealen abzuleiten. Es sollen Kriterien hergeleitet werden, wie man aus der räumlichen Repräsentation der Hirnkarten die logische Beziehung von Arealen in diesen Karten ablesen kann. Zu einer gegebenen Gehirnkarte und einem Areal in dieser Karte (beziehungsweise eine gegebene Region im Koordinatenraum) soll für andere Areale in den räumlich registrierten Hirnkarten festgestellt werden, ob sie mit dem vorgegebenen Areal identisch sind, es enthalten, in ihm enthalten sind, sich mit ihm überschneiden oder ob die Areale disjunkt sind. Die Ergebnisse sollen dann mit den schon bekannten Beziehungen aus der CoCoMac-Datenbank verglichen werden, um die Güte der Aussagen zu bewerten.

Ziel ist es, die auf räumlichen Daten basierende Beziehungsklassifikation mit den eingetragenen Beziehungen aus CoCoMac zu vergleichen. Dies kann auf Fehler in beiden Darstellungen aufmerksam machen. Diskrepanzen können sowohl darauf hinweisen, daß die in CoCoMac eingetragene Beziehung falsch ist, als auch, daß die Areale in der räumlichen Darstellung falsch abgebildet sind.

Die in diesem Kapitel behandelten Themen sind Teile der Dissertation von Herrn Gleb Bezgin. Die erzielten Ergebnisse entstanden in Zusammenarbeit mit Herrn Bezgin, Herrn Prof. Dr. Egon Wanke und Herrn Prof. Dr. Rolf Kötter und sind teilweise bereits in [BWKK08] veröffentlicht. In den nachfolgenden Abschnitten werden in erster Linie die algorithmischen Fragestellungen formal aufgearbeitet. Dagegen konzentriert sich die Arbeit von Herrn Bezgin stärker auf die Implementierung der Algorithmen und auf die neurologische Bedeutung der Ergebnisse.

7.2 Verwendete Daten

Es soll also ein Verfahren entwickelt werden, um logische Beziehungen zwischen kortikalen Arealen in verschiedenen Gehirnkarten zu bestimmen - also ob sie identisch, ineinander enthalten, überlappend oder disjunkt sind.

Grundlage hierfür ist eine zweidimensionale Repräsentation der Oberfläche des Makaken-Cortex. Diese besteht aus einem Netzwerk von Knotenpunkten auf einer Schablone der kortikalen Oberfläche. Für jede Gehirnregion ist eine Menge von Knotenpunkten im Oberflächenraster angegeben, auf die die Region abgebildet ist. Diese Daten wurden mit Hilfe der CARET-Software generiert, die vom Van Essen Laboratory erstellt wurde ([Ess04], <http://brainmap.wustl.edu/caret>). Die CARET-Software (Computerized Anatomical Reconstruction and Editing Toolkit) bietet die Möglichkeit, verschiedene Oberflächendarstellungen vom Menschen-, Makaken-, Ratten- sowie Mäusegehirn zusammen mit strukturellen und funktionellen Daten zu verarbeiten. Die Information, welche Hirnstruktur welcher Menge von

Knoten im Oberflächenraaster zugeordnet sind, werden in der „Surface Management System“ (SuMS) Datenbank gespeichert, ([DDE01], <http://sumsdb.wustl.edu:8081/sums/index.jsp>).

Anschaulich kann man sich diese Daten wie folgt vorstellen: Die Cortexoberfläche wird mit einem Raster von Datenpunkten versehen, die bestimmte Koordinaten repräsentieren. Wenn man jetzt auf dieser Oberfläche die Areale einer Karte einzeichnet, kann man anschließend entscheiden, welche Punkte zu welchem Areal gehören. Anschließend wird diese Darstellung auf eine Fläche im zweidimensionalen Raum projiziert. Durch diese Projektion der gewölbten Oberfläche aus dem dreidimensionalen Raum in den zweidimensionalen Raum liegen vorher benachbarte Datenpunkte nicht unbedingt nebeneinander. Deshalb werden für jeden Datenpunkt auch die Punkte in seiner Nachbarschaft angegeben. Die räumliche Darstellung wird also reduziert auf eine Menge von diskreten Punkten mit Nachbarschaftsbeziehungen. Die verschiedenen Hirnkarten wurden so realisiert, daß zu einem Areal in einer Karte immer alle Datenpunkte angegeben werden, die in dem Areal enthalten sind. Ebenso sind für einen Datenpunkt alle Areale der verschiedenen Karten abrufbar, zu denen er gehört. Dabei sollten alle Hirnregionen, die zu derselben Hirnkarte gehören, durch paarweise disjunkte Mengen von Knoten repräsentiert sein. Ein Knoten kann aber durchaus zu mehr als einer Hirnregion gehören, wenn diese sich in verschiedenen Hirnkarten befinden. Eine Darstellung des primären visuellen Cortex aus verschiedenen Karten durch CARET ist in Abbildung 7.1 zu sehen.



Abbildung 7.1: Darstellung des primären Cortex (V1) aus verschiedenen Karten, wie er durch die CARET Software repräsentiert wird. Umrisse in magenta: V1 aus [Bro09]; in rot: V1 aus [vBB47]; in schwarz: V1 aus [LE00]. Links ist eine laterale und in der Mitte eine mediale Ansicht zu sehen, rechts dagegen eine flache Kartenrepräsentation. Wie man leicht sieht, weisen die Grenzen in den verschiedenen Karten Abweichungen voneinander auf. Dabei sollte dieses Areal in allen drei Darstellungen dieselben Umrisse haben.

Hier wurde das „F99UA1 Monkey Hemisphere Template“ aus der SuMS-Datenbank mit den folgenden anatomischen Daten verwendet:

1. B09, Brodmanns Parzellierung des menschlichen Gehirns, übertragen auf das Makakengehirn [Bro09];
2. BB47, Parzellierungsschema des Makaken nach von Bonin und Bailey [vBB47];
3. UD86, die Studie von Ungerleider und Desimone [UD86];

4. PG91, Parzellierungsschema von Preuss und Goldman-Rakic [PGR91];
5. FV91, die Karte, die für die vergleichende Studie von Felleman und Van Essen angefertigt wurde [FE91]
6. PHT00 Makaken Paritionierung für den stereotaxischen Atlay von Paxinos, Huang und Toga [PHT00];
7. LV00, die Karte aus der Studie von Lewis und Van Essen [LE00].

Für eine Reihe von Arealen dieser Karten sind in der CoCoMac-Datenbank logische Beziehungen angegeben. Diese Beziehungen zwischen zwei Arealen sind kodiert als I (Identity), S (Subarea), L (Larger area), O (Overlap), D (Disjoint). Dabei wurden nur solche Beziehungsangaben verwendet, die direkt aus der Literatur entstammen und keine, die von dem Algorithmus ORT generiert wurden (vergleiche Abschnitte 7.1 und 2.4). Ein genaue Auflistung der Areale und ihrer Beziehungen ist im Anhang in Tabelle B.6 zu finden. Damit ist es möglich, die in CoCoMac gespeicherten Beziehungen mit dem Ergebnis des zu entwickelnden Verfahrens zu vergleichen. Wir haben also verschiedene Karten und ihre Abbildung auf Knoten in einem Netzwerk zur Verfügung, sowie für einige dieser Areale und Karten schon bekannte logische Beziehungen.

7.3 Formalisierung der Begriffe

Die aus CARET bezogenen Strukturen wollen wir jetzt formalisieren. Wir können das Netzwerk aus Datenpunkten als ungerichteten Graphen auffassen, wobei die angegebenen Nachbarschaftsbeziehungen den Kanten entsprechen. Sei $G = (V, E)$ ein ungerichteter Graph. Dann ist die Nachbarschaft von einem Knoten $v \in V$ definiert durch

$$N(v) = \{ u \in V \mid \{v, u\} \in E \}.$$

Wir erhalten damit folgende formale Definition für Karten.

Definition 7.1 Sei $G = (V, E)$ ein ungerichteter Graph und $\mathcal{A}_K = \{A_{K,1}, \dots, A_{K,n}\}$ eine endliche Menge von Objekten (Arealen). Ein Karte K ist eine Abbildung von der Knotenmenge V in die Menge \mathcal{A}_K vereinigt mit der leeren Menge, also $K : V \rightarrow \mathcal{A}_K \cup \emptyset$, für die gilt

$$K(v) = A_K \Leftrightarrow \exists u \in V \text{ mit } K(u) = A_K \text{ und } v \in N(u)$$

für alle $v \in V$.

Wir schreiben $v \in A_K$, wenn $v \in K^{-1}(A_K) = \{ u \in V \mid K(u) = A_K \}$, wenn also v im Urbild von A_K liegt, und sagen dann „ v gehört zu A_K “. Somit ist A_K auch die Menge aller Knoten, die zu A_K gehören. Entsprechend sei $|A_K| := |\{ u \in V \mid K(u) = A_K \}|$.

Diese Definition einer Karte impliziert, dass der Teilgraph, der von den Knoten induziert wird, die zu einem Areal A_K gehören, zusammenhängend ist. Dies ist sinnvoll, da ein anatomisches Areal auf der Oberfläche des Cortex auch eine zusammenhängende Fläche ist. Wir können ein Areal A_K also als eine Teilmenge von V auffassen. Da eine Karte K eine Abbildung ist,

kann jeder Knoten $v \in V$ nur zu einem Areal in \mathcal{A}_K gehören. Die Bildmengen verschiedener Karten werden als disjunkt angesehen und Areale verschiedener Karten sind durch ihre Indizes unterschieden. Damit gehört jedes Areal nur zu der Bildmenge einer Karte. Sei \mathcal{K} eine endliche Menge von Karten. Dann ist

$$\mathcal{A}_{\mathcal{K}} = \bigcup_{K \in \mathcal{K}} \mathcal{A}_K$$

die Menge aller Areale in allen Karten.

7.4 Bestimmung der Beziehungen zwischen den Arealen

Für die Bestimmung der logischen Beziehungen zwischen Arealen wollen wir einen einfachen Kennwert finden, an dem man das Ergebnis leicht ablesen kann. Dazu hilft die folgende Überlegung. Eine Menge A ist nur dann in einer Menge B enthalten, wenn der Schnitt $A \cap B$ gleich A ist. Für zwei Mengen A und B gilt also:

$$A \subseteq B \Leftrightarrow A \cap B = A \Leftrightarrow |A \cap B| = |A|$$

Mithilfe dieser Aussage können wir alle logischen Mengenbeziehungen zwischen zwei Mengen charakterisieren (siehe Tabelle 7.1 für $A \cap B \neq \emptyset$).

Verhältnis $A_1 \cap A_2$ zu A_1 bzw. $ A_1 \cap A_2 $ zu $ A_1 $	Verhältnis $A_1 \cap A_2$ zu A_2 bzw. $ A_1 \cap A_2 $ zu $ A_2 $	Beziehung A_1 und A_2
$A_1 \cap A_2 = A_1$ \Updownarrow $ A_1 \cap A_2 = A_1 $	$A_1 \cap A_2 = A_2$ \Updownarrow $ A_1 \cap A_2 = A_2 $	A_1 und A_2 sind identisch
$A_1 \cap A_2 = A_1$ \Updownarrow $ A_1 \cap A_2 = A_1 $	$A_1 \cap A_2 \subsetneq A_2$ \Updownarrow $ A_1 \cap A_2 < A_2 $	A_1 ist in A_2 enthalten
$A_1 \cap A_2 \subsetneq A_1$ \Updownarrow $ A_1 \cap A_2 < A_1 $	$A_1 \cap A_2 = A_2$ \Updownarrow $ A_1 \cap A_2 = A_2 $	A_1 enthält A_2
$A_1 \cap A_2 \subsetneq A_1$ \Updownarrow $ A_1 \cap A_2 < A_1 $	$A_1 \cap A_2 \subsetneq A_2$ \Updownarrow $ A_1 \cap A_2 < A_2 $	A_1 und A_2 sind überlappend

Tabelle 7.1: Größenverhältnisse und Beziehungen von Arealen mit $A_1 \cap A_2 \neq \emptyset$

Nun könnte man im ersten Ansatz annehmen, daß sich das Problem, zu entscheiden, in welcher Beziehung zwei Areale aus verschiedenen Karten stehen, wie folgt formulieren läßt.

Gegeben sei ein Areal A_K einer Karte K . Für jedes Areal $A'_{K'}$ aus einer Karte $K' \neq K$ mit $A_K \cap A'_{K'} \neq \emptyset$ ist zu berechnen, ob $A_K \subseteq A'_{K'}$ und ob $A'_{K'} \subseteq A_K$.

Damit sind schon zwei der möglichen Beziehungen abgedeckt, und man könnte auch alle weiteren logischen Beziehungen ableiten: Ist $A_K \subseteq A'_{K'}$ und $A'_{K'} \subseteq A_K$, dann sind A_K und $A'_{K'}$ identisch. Ist $A_K \not\subseteq A'_{K'}$ und $A'_{K'} \not\subseteq A_K$, dann überlappen sich A_K und $A'_{K'}$ (vergleiche Tabelle 7.1). Gibt es keine gemeinsamen Knoten, so sind die Areale disjunkt. Für diese Mengenvergleiche muß man nur die Größe der Areale und ihrer Schnitte kennen.

Diese einfache Überlegung funktioniert deshalb nicht, weil die Grenzen der einzelnen Areale nicht unbedingt mit der gleichen Präzision bestimmbar sind, die das Raster der Knoten aufweist. Auch hier ist die genaue Festlegung der Grenzen vom Beobachter abhängig, und die Areale wurden in Handarbeit auf die dem Netzwerk zugrunde liegende Oberfläche übertragen. Dabei ist davon auszugehen, daß es zu Ungenauigkeiten bei der Übertragung gekommen ist, auch dadurch, daß die Merkmale, die die Arealränder definieren, in einigen Fällen keine klar erkennbare Grenze bilden, sondern der Übergang von einem Merkmal zum anderen Raum einnimmt. Deshalb hilft die exakte mathematische Lösung nicht weiter. Wie man schon in Abbildung 7.1 sehen konnte, haben auch identische Areale nicht unbedingt alle Knoten gemeinsam. Andersherum kann es Areale geben, die ein paar Knoten gemeinsam haben, aber dennoch disjunkt sind. Es muß also geklärt werden, wie man diese Ungenauigkeiten in der Auswertung berücksichtigt. Im folgenden sollen dazu zwei Verfahren vorgestellt werden, um entsprechende Entscheidungskriterien zu erhalten.

7.4.1 Verfahren 1

Dieser Ansatz verwendet nur Mengentheorie, ohne Berücksichtigung der Nachbarschaftsbeziehungen. Dazu verwenden wir folgende Definition.

Definition 7.2 *Seien $A_{K_1} \in K_1, A_{K_2} \in K_2$ zwei Areale in verschiedenen Karten. Dann ist die relative Größe des Schnittes $A_{K_1} \cap A_{K_2}$ im Areal A_{K_1} definiert durch*

$$R(A_{K_1} \cap A_{K_2} | A_{K_1}) = \frac{|A_{K_1} \cap A_{K_2}|}{|A_{K_1}|},$$

und die relative Größe des Schnittes $A_{K_1} \cap A_{K_2}$ im Areal A_{K_2} definiert durch

$$R(A_{K_1} \cap A_{K_2} | A_{K_2}) = \frac{|A_{K_1} \cap A_{K_2}|}{|A_{K_2}|}.$$

Da der Schnitt nicht mehr Elemente als eines der Areale enthalten kann, gilt

$$0 \leq R(A_{K_1} \cap A_{K_2} | A_{K_1}) \leq 1 \quad \text{und} \quad 0 \leq R(A_{K_1} \cap A_{K_2} | A_{K_2}) \leq 1.$$

Es soll also die relative Größe des Schnittes zweier Areale im Bezug auf die beiden Ausgangsareale genutzt werden, um Informationen über deren Beziehung zu erhalten. Dabei wird die relative und nicht die absolute Größe verwendet, um eine Normierung und damit eine Vergleichbarkeit zwischen Arealen verschiedener Größen zu erreichen.

Aus den beiden Größen $R(A_{K_1} \cap A_{K_2} | A_{K_1})$ und $R(A_{K_1} \cap A_{K_2} | A_{K_2})$ sollen jetzt analog zu Tabelle 7.1 logische Beziehungen für die beiden beteiligten Areale A_{K_1} und A_{K_2} abgeleitet werden. Eine Übersicht, wie sich die logischen Beziehungen aus den Größen ergeben, findet sich in Tabelle 7.2.

$r_1 = R(A_{K_1} \cap A_{K_2} A_{K_1})$	$r_2 = R(A_{K_1} \cap A_{K_2} A_{K_2})$	logische Beziehung zwischen A_{K_1} und A_{K_2}
$r_1 = 1$	$r_2 = 1$	A_{K_1} und A_{K_2} sind identisch
$r_1 = 1$	$r_2 \ll 1$	A_{K_1} ist in A_{K_2} enthalten
$r_1 \ll 1$	$r_2 = 1$	A_{K_1} enthält A_{K_2}
$0 \ll r_1 \ll 1$	$0 \ll r_2 \ll 1$	A_{K_1} und A_{K_2} überlappen sich
$r_1 = 0$	$r_2 = 0$	A_{K_1} und A_{K_2} sind disjunkt

Tabelle 7.2: Übersicht über die logischen Beziehungen, die sich aus den Größen der beiden Werte $R(A_{K_1} \cap A_{K_2} | A_{K_1})$ und $R(A_{K_1} \cap A_{K_2} | A_{K_2})$ ergeben.

Dabei ist Tabelle 7.2 eine direkte Übertragung von Tabelle 7.1 auf die neu eingeführten Begriffe. Die Frage, wie man die Ungenauigkeiten in den Daten berücksichtigt, ist damit noch nicht geklärt. Sie läßt sich aber umformulieren in die Frage, wann man zwei Werte noch als gleich ansieht, auch wenn sie nicht exakt gleich sind. Wir wollen uns diesem Problem grafisch nähern. Bei der Berechnung der relativen Größe eines Schnittes erhält man für jedes der beiden beteiligten Areale einen Wert in dem reellen Intervall $[0, 1]$ und damit insgesamt einen Wert in $[0, 1] \times [0, 1]$. Nehmen wir einmal an, wir lassen als Gleichheit noch eine Abweichung von 25% von den vorgegebenen Werten zu. Wenn man diese Annahmen in ein zweidimensionales Koordinatensystem überträgt, ergibt sich ein Bild wie in Abbildung 7.2.

Wir gehen aber aus Symmetriegründen davon aus, daß die Grenze des Bereiches „Identity“ einen konstanten Abstand zum Punkt $(1, 1)$ und die Grenze des Bereiches „Disjoint“ einen konstanten Abstand zum Punkt $(0, 0)$ hat. Diese beiden Bereiche beschreiben also Kreisabschnitte mit dem Mittelpunkt $(1, 1)$ beziehungsweise $(0, 0)$. Die Grenze des Bereichs „Subarea“ sollte parallel zur y-Achse, die Grenze des Bereichs „Larger Set“ parallel zur x-Achse verlaufen. Weil eine eindeutige Zuordnung eines jeden Punktes im Quadrat zu einer Beziehung erreicht werden soll, gehen wir davon aus, daß alle Bereiche disjunkt sind. Deshalb beginnen die Bereiche für „Subarea“ und „Larger Set“ erst dort, wo der Bereich für „Identity“ endet. Der Bereich „Overlap“ nimmt den Raum im Quadrat ein, der sich außerhalb aller anderen Bereiche befindet. Dabei ist die eigentliche Größe der einzelnen Bereiche unbekannt und soll im folgenden möglichst genau bestimmt werden.

In Abbildung 7.2 fällt sofort auf, daß die Einteilung symmetrisch zur Winkelhalbierenden ist. Diese Eigenschaft wollen wir nutzen, um die vorhandenen Daten soweit wie möglich zu konzentrieren. Da die Aussagen „ A_1 enthält A_2 “ und „ A_2 ist in A_1 enthalten“ dasselbe bedeuten, können wir uns auf eine von beiden beschränken. Wir wollen im folgenden nur noch

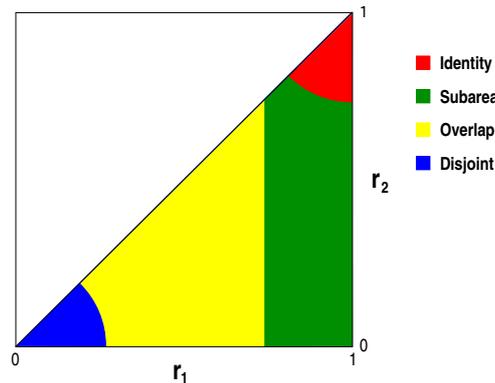


Abbildung 7.3: Grafische Darstellung der Überlegungen aus Tabelle 7.2 mit der Einschränkung $r_1 = R(A_{K_1} \cap A_{K_2} | A_{K_1}) \geq r_2 = R(A_{K_1} \cap A_{K_2} | A_{K_2})$ und einer angenommenen Toleranz für Gleichheit von 25%.

daß die Arealen von 1 bis a , die Karten von 1 bis k und die Knoten von 1 bis n durchnummeriert sind. Wir verwenden zur Berechnung ein zweidimensionales Array, dessen Größe in beiden Dimensionen der Anzahl aller Areale a entspricht. Dieses wird mit 0 initialisiert, und in ihm wird für jedes Arealpaar die Anzahl der Knoten im Schnitt gespeichert. Der Algorithmus geht der Reihe nach alle n Knoten durch und überprüft für jede Karte K , ob der Knoten in dieser Karte einem Areal zugeordnet wird. Wenn ja, erhöht er einen Zähler, der die Anzahl der Knoten in jedem Areal zählt, um eins und überprüft, ob in den noch nicht betrachteten Karten dieser Knoten auf ein weiteres Areal abgebildet wird. Wenn ja, erhöht er den Zähler für die jeweilige Schnittmenge um eins.

Algorithmus: Berechnen der Größe der Schnittmengen aller Areale

```

initialisiere leeres Array ArealMatrix[a][a]
für  $i = 1, \dots, a$ 
  für  $j = 1, \dots, a$ 
    ArealMatrix[i][j]=0
für  $i = 1, \dots, n$ 
  für  $j = 1, \dots, k$ 
    wenn Knoten  $i$  gehört zu einem Areal  $A_1$  in Karte  $j$  dann {
      ArealMatrix[A1][A1] ++
      für  $l = j + 1, \dots, k$ 
        wenn Knoten  $i$  gehört zu einem Areal  $A_2$  in Karte  $l$  dann {
          ArealMatrix[A1][A2]++
        }
    }

```

Am Ende steht in der i -ten Zeile und i -ten Spalte des Arrays die Anzahl der Knoten in Areal i , und in der i -ten Zeile und j -ten Spalte, $i < j$, die Anzahl der Knoten im Schnitt von Areal i und Areal j . Die Laufzeit des Algorithmus ist in $O(a^2 + n \cdot k^2)$.

Nachdem die Größe der Schnittmengen bestimmt war, wurden damit für alle Arealpaare $A_{K_1}, A_{K_2} \in \mathcal{A}_{\mathcal{K}}$, deren Beziehung zueinander in der CoCoMac-Datenbank vermerkt ist, die

Bewertungen $R(A_{K_1} \cap A_{K_2} | A_{K_1})$ und $R(A_{K_1} \cap A_{K_2} | A_{K_2})$ berechnet. Die genauen Werte der einzelnen Beziehungen sind im Anhang in Tabelle B.6 zu finden. Insgesamt haben wir 56 „Identity“- , 96 „Disjoint“- 32 „Subarea“- und sechs „Overlap“-Beziehungen. Wenn wir die Ergebnisse als Koordinaten auffassen, können wir für jedes Arealpaar einen Punkt in ein Koordinatensystem eintragen. Die Ergebnisse der Berechnung sind in Abbildung 7.4 grafisch dargestellt.

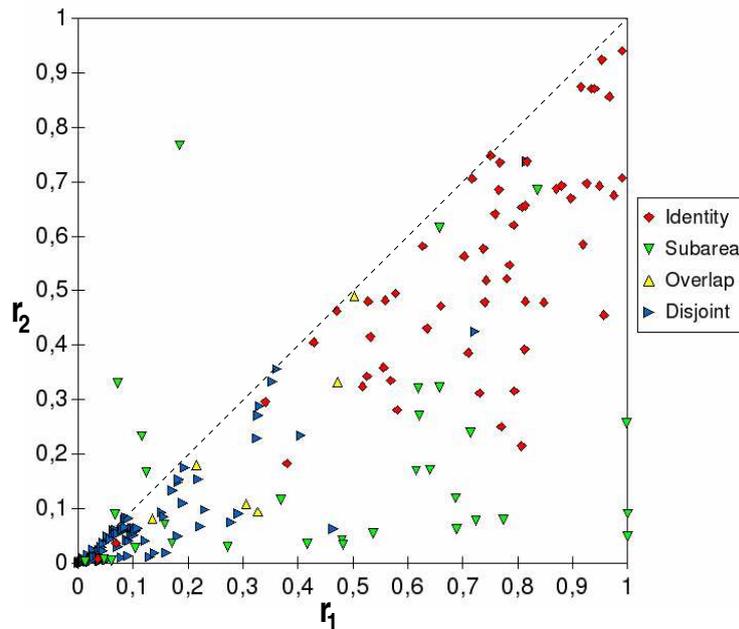


Abbildung 7.4: Grafische Darstellung der berechneten Werte für die einzelnen Beziehungen. $r_1 = R(A_1 \cap A_2 | A_1)$, $r_2 = R(A_1 \cap A_2 | A_2)$

Wie man sieht, hat sich die Hoffnung, daß alle Punkte der „Subarea“-Beziehungen unterhalb der Winkelhalbierenden liegen, nicht erfüllt. Fünf von den 32 Punkten für diese Beziehung liegen oberhalb. Das bedeutet, daß in diesen fünf Fällen das enthaltene, also kleinere Areal mit seiner Gesamtfläche prozentual weniger Anteil am Schnitt hat, als das größere Areal. Insbesondere der Punkt $(0,185;0,767)$ fällt auf, der zu dem Arealpaar BB47-IA und B09-15 gehört. Diese Beziehung sollte überprüft werden. Alle fünf „Subarea“ Beziehungen oberhalb der Diagonalen bestehen aus Arealpaaren, von denen ein Areal aus der Karte BB47 und das andere aus der Karte B09 stammt.

Die vier verschiedenen Punktgruppen gehen ineinander über und lassen auf den ersten Blick keine Einteilung in verschiedene Bereiche zu. Trotzdem möchte wir jetzt eine durch diese Punkte gegebene Einteilung der Fläche finden, aufgrund derer wir neu berechnete Daten für Arealpaare einer der vier Gruppen zuordnen können. Wir möchten also zu jedem Punkt in der Ebene sagen können, was für eine Beziehung er repräsentiert. Dabei sollte berücksichtigt werden, daß die vier Gruppen unterschiedlich viele Ausgangspunkte haben, abhängig von der registrierten Zahl der Beziehungen dieser Art.

Das ergibt folgendes Problem: Gegeben sei eine endliche Menge von Punkten $P \subset [0, 1] \times [0, 1] \subset \mathbb{R}^2$, eine Abbildung $f : P \rightarrow \{I, S, O, D\}$ und ein Punkt $q \in [0, 1] \times [0, 1]$. Gesucht wird eine Zuordnung $q \mapsto B \in \{I, S, O, D\}$, die die Verteilung der Punkte in P möglichst gut

reflektiert.

Wie kann man mit diesen Daten jetzt die gesuchten Bereiche für die verschiedenen Beziehungsarten finden? Im Einklang mit unseren Vorüberlegungen gehen wir davon aus, daß die Grenzen der Bereiche für die „Identity“- und die „Disjoint“-Beziehungen mit einem Radius r_I um den Punkt $(1, 1)$ beziehungsweise mit einem Radius r_D um den Punkt $(0, 0)$ verläuft. Der Bereich „Subarea“ hat eine Grenze parallel zur y-Achse. Um die beiden Radien und die Distanz d_S zur Geraden $x = 1$ zu finden, wurden in Schritten der Größe 0,01 alle möglichen Werte für r_I , r_D und d_S durchprobiert und dabei erfaßt, wieviele Datenpunkte einer Gruppe sich jeweils im richtigen Gebiet befinden. Dabei wurde ein Punkt dann zum „Identity“-Bereich gerechnet, wenn sein Abstand zum Punkt $(1, 1)$ kleiner oder gleich r_I war, zum „Disjoint“-Bereich, wenn sein Abstand zum Punkt $(0, 0)$ kleiner oder gleich r_D war, zum Bereich „Subarea“, wenn sein Abstand zur Geraden $x = 1$ kleiner oder gleich d_S war und er nicht schon zu einem der beiden anderen Bereiche gehörte. Der Bereich „Overlap“ enthielt alle Punkte, die zu keinem der drei anderen Bereiche gehörte. Dabei wurden die Radien r_I und r_D jeweils so gewählt, daß sich ihre Bereiche nicht überlappen, aber im Verlauf der Suche trotzdem für beide Radien alle entsprechenden Zahlen zwischen 0 und $\sqrt{2}$ im Rahmen der Inkremente von 0,01 ausprobiert wurden.

Da die Daten in verschiedene Gruppen eingeteilt sind, die nicht gleich viele Vertreter haben, ist es aber nicht sinnvoll, die absolute Zahl der falsch platzierten Punkte als Kriterium für die Güte zu nehmen. Vielmehr ist es adäquater, den Prozentsatz der falsch platzierten Punkte für jede Gruppe zu berechnen. Es soll also ein normiertes Maß für die Punkte verwandt werden. Wir erhalten für unsere vier Beziehungarten also vier Prozentsätze. Da die Güte der Zuordnung nur so gut ist wie die Güte für die am schlechtesten abgebildete Gruppe, sollte das Maximum der Fehlerprozentätze minimiert werden. Gibt es mehrere Lösungen mit minimalen, so sollte als nächstes der nächstgrößere Wert zum Vergleich herangezogen werden usw.

Die bei dieser Berechnung gefundenen optimalen Fehlerwerte sind in Tabelle 7.3 zu sehen.

Beziehung	Fehler
Identity	25%
Subarea	43,75%
Overlap	33,33%
Disjoint	23,96%

Tabelle 7.3: Prozentsätze der Punkte, die bei einer optimalen Einteilung nicht richtig zugeordnet werden, geordnet nach Beziehungsart.

Mit diesen Fehlerprozenten sind die folgenden Werte für die Bereiche verbunden: $r_I = 0,74$, $r_D = 0,15$ und $0,64 \leq d_S \leq 0,67$. Der letzte Wert läßt sich nicht eindeutig bestimmen, da es keine „Overlap“- und „Subarea“-Punkte mit einem x-Wert im Bereich von 0,33 bis 0,36 gibt. Diese beiden Punktgruppen sind für die Bestimmung dieser Grenze relevant. Wir werden hier aber als Grenze $d_S = 0,64$ festlegen, da dieser Wert am kleinsten ist, und damit am nächsten an die Theorie heran kommt. Für eine grafische Darstellung siehe Abbildung 7.5. Der hohe Fehler bei den „Subarea“-Punkten erklärt sich durch die fünf Punkte, die außerhalb des Dreiecks liegen, da diese alle einen $r_2 = R(A_{K_1} \cap A_{K_2} | A_{K_2})$ -Wert von weniger als 0,2 haben. Wenn die Breite d_S bis dorthin reichen würde, würden auch alle „Overlap“-Punkte in

diesem Gebiet liegen. Diese haben aufgrund ihrer geringen Anzahl bei der Fehlerberechnung ein größeres Gewicht als die Punkte in den anderen drei Gruppen.

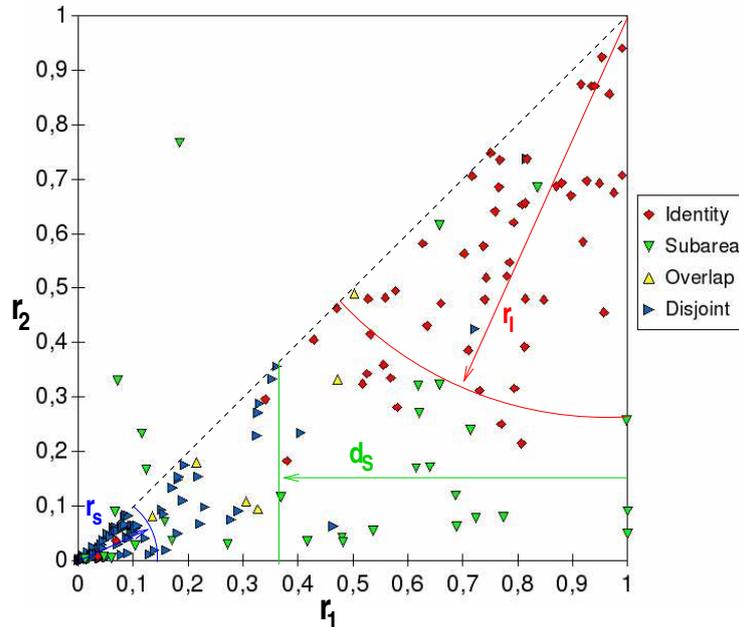


Abbildung 7.5: Grafische Darstellung der berechneten Bereiche für die einzelnen Beziehungen. $r_1 = R(A \cap B|A)$, $r_2 = R(A \cap B|B)$

Insgesamt sind die Fehler alle recht groß, so daß man mit der Einteilung nicht wirklich zufrieden sein kann. Außerdem sehen die Bereiche bis auf den für „Disjoint“ nicht so aus wie erwartet. So nimmt der Bereich für „Identity“ mehr als $\frac{2}{3}$ der zur Verfügung stehenden Fläche ein. Dagegen besitzt der „Overlap“-Bereich, der theoretisch am größten sein sollte, weniger als $\frac{1}{12}$ der Gesamtfläche. Allerdings ist zu bedenken, daß die in der Tabelle 7.2 angegebenen Zahlen bei Idealbedingungen gelten. Bei den vorliegenden Daten ist es aber so, daß zum Beispiel Grenzen leicht verschoben sein können oder ein Knoten vielleicht bei der Abbildung vergessen wurde. So erhält man auch bei Arealen, die bekanntermaßen identisch sind, keine Schnittgrößen, die der Größe der Areale entsprechen. Um diese Abweichungen besser zu berücksichtigen, soll jetzt noch ein zweites Verfahren vorgestellt werden.

7.4.2 Verfahren 2

Hier wollen wir uns auf einer anderen Art der Frage widmen, innerhalb welcher Grenzen die Werte noch unseren Zielvorgaben entsprechen. Dazu wollen wir diesmal die Werte dem Problem anpassen, indem wir eine Gewichtung einführen.

Die oben vorgestellte Herangehensweise vernachlässigt die räumlichen Informationen, also die Nachbarschaftsinformationen der Knoten, die ja für die Areale vorhanden sind. Wie schon erwähnt, induzieren die Knoten eines Areals einen zusammenhängenden Teilgraphen (denn in der Anatomie stellt ein Areal eine zusammenhängende Fläche dar). Diese Eigenschaft wollen wir uns im folgenden zu Nutze machen. Die schon angesprochenen Ungenauigkeiten bei der Übertragung der anatomischen Areale auf das Netzwerk sollten den Rand stärker als das

Innere der Areale betreffen, da die grundsätzliche Position klarer ist als die genauen Umrisse ist. Deshalb sollte eine Übereinstimmung im Zentrum der Areale beim Vergleich stärker ins Gewicht fallen als eine Übereinstimmung der äußeren Sektionen.

Während es in Bezug auf eine Fläche intuitiv klar ist, was das Innere und was der Rand ist, müssen wir diese Begriffe in Bezug auf einen Teilgraphen unseres Netzwerkes erst noch definieren.

Sei $G = (V, E)$ ein Graph und $A \subseteq V$ ein Areal. Dann ist der *Rand* von A die Menge aller Knoten $v \in A$, die einen Nachbarn u haben, der nicht in A liegt, also $u \notin A$. Das *Innere* von A bilden dann alle Knoten von A , die nicht zum Rand gehören. Die Entfernung eines Knotens $v \in A$ zum Rand kann man dann als das Minimum der Weglängen aller Wege von v zu einem Randknoten von A definieren. Diese Idee soll jetzt in ein Knotengewicht übertragen werden.

Wir definieren die folgende Gewichtungsfunktion $f_{V'} : V' \rightarrow \mathbb{N}$ auf einer Knotenmenge $V' \subseteq V$, die einen Knoten ein umso größeres Gewicht zuordnet, je weiter er vom Rand von V' entfernt ist:

$$f_{V'}(v) = \begin{cases} \min\{f(u) + 1 | u \in N(v)\} & \text{falls } N(v) \subset V', \\ 1 & \text{sonst,} \end{cases}$$

wobei $N(v)$ die Menge der Nachbarn von v in G ist (vergleiche Kapitel 3).

Ein Beispiel für die Gewichtung einer Knotenmenge aufgrund ihrer Nachbarschaftsbeziehungen ist in Abbildung 7.6 gegeben. Knoten die Nachbarn außerhalb der Menge V' haben, also am Rand liegen, erhalten den Wert 1. Die Werte steigen dann an, je weiter man sich innerhalb des Teilgraphen von diesem Rand entfernt. Dabei wurde der Wert 1 und nicht 0 für die Knoten auf dem Rand gewählt, da jeder Knoten des Areals ein positives Gewicht erhalten soll.

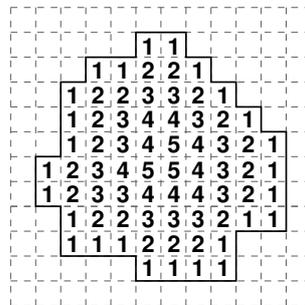


Abbildung 7.6: Beispiel für die Berechnung der Gewichte. Dabei werden zwei Quadrate als benachbart angesehen, wenn sie eine gemeinsame Linie haben.

Mithilfe der Gewichtungsfunktion $f_{V'}$ können wir analog zur relativen Größe des Schnittes in einem Areal die relative Größe des Schnittes in einem gewichteten Areal definieren.

Definition 7.3 Seien $A, B \subseteq V$ zwei Areale. Dann ist die gewichtete relative Größe des Schnittes von $A \cap B$ im Areal A definiert durch

$$R_g(A \cap B | A) = \frac{\sum_{v \in A \cap B} f_A(v)}{\sum_{v \in A} f_A(v)},$$

und die gewichtete relative Größe des Schnittes von $A \cap B$ im Areal B definiert durch

$$R_g(A \cap B|B) = \frac{\sum_{v \in A \cap B} f_B(v)}{\sum_{v \in B} f_B(v)}.$$

Es gilt

$$0 \leq R_g(A \cap B|A) \leq 1 \quad \text{und} \quad 0 \leq R_g(A \cap B|B) \leq 1.$$

Der Zusammenhang zwischen den Werten von $R_g(A \cap B|A)$ und $R_g(A \cap B|B)$ und den Beziehungen der Areale A und B zueinander ist theoretisch analog zu den einfachen relativen Größen, vergleiche Tabelle 7.2. Wie auch dort gelten die Zusammenhänge in dieser exakten Form nur für genaue Daten, die hier ja nicht vorliegen. Allerdings sollte die Gewichtungsfunktion wesentliche Gemeinsamkeiten stärker berücksichtigen, so daß die Hoffnung besteht, in diesen Fällen näher an die theoretischen Vorgaben zu kommen. Kleinere Unterschiede in den Randbereichen von Arealen in verschiedenen Karten verringern die Werte des gewichteten relativen Schnittes weniger als die des relativen Schnittes. Ein Beispiel hierfür ist in Abbildung in Abbildung 7.7 zu sehen. Die beiden Areale enthalten gleich viele Knoten, und ihr Inneres, also jeweils alle Knoten, die ein größeres Gewicht als 1 haben, stimmen überein. Die Ränder hingegen weisen jeweils Knoten auf, die nicht im Schnitt liegen. Die beiden gewichteten Größen sind größer als die ungewichteten, deuten also stärker in Richtung Identität. Auffällig ist auch, daß sich im gewichteten Fall die Werte für die beiden unterscheiden, obwohl beide Mengen gleich viele Knoten enthalten. Da die Bewertung jedoch nicht von der Zahl der Knoten abhängt, sondern vom Aufbau des Graphen, ist dieses nicht weiter überraschend.

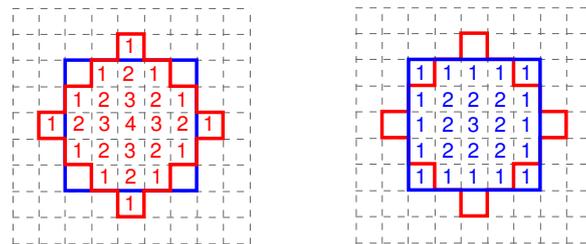


Abbildung 7.7: Beispiel für die Veränderung der Schnittgröße bei kleinen Unterschieden am Rand der Areale: A_1 sei das rote Areal und A_2 das blaue. Dann ist $R(A_1 \cap A_2|A_1) = R(A_1 \cap A_2|A_2) = \frac{21}{25}$, aber $R_g(A_1 \cap A_2|A_1) = \frac{40}{44}$ und $R_g(A_1 \cap A_2|A_2) = \frac{31}{35}$.

Es kann aber bei den gewichteten relativen Größen im Vergleich zu den ungewichteten nicht nur zu einer Verschiebung der Werte für beide Areale in dieselbe Richtung, sondern auch zu einer Umkehrung der Größenverhältnisse kommen, wie das Beispiel in Abbildung 7.8 zeigt. Im gewichteten Fall werden die Knoten im Zentrum von Areal A_2 stärker berücksichtigt, während es für Areal A_1 kaum einen Unterschied zur ungewichteten Bewertung gibt. Das heißt, die gewichteten Größen verstärken nicht einfach die Tendenzen der ungewichteten Größen, sondern ergeben insgesamt eine andere Bewertung.

Um die gewichteten Werte zu bestimmen, muß zuerst für ein Areal die Knotenbewertung berechnet werden. Im folgenden ist ein Algorithmus angegeben, der in einem ungerichteten Graphen $G = (V, E)$ für eine gegebene Knotenmenge $A \subseteq V$ die Knotengewichte berechnet. Zuerst werden in einer Vorverarbeitungsphase alle Knoten gefunden, die Nachbarn außerhalb von A haben. Diese erhalten den Wert 1 und werden in eine Queue Q_1 eingefügt.

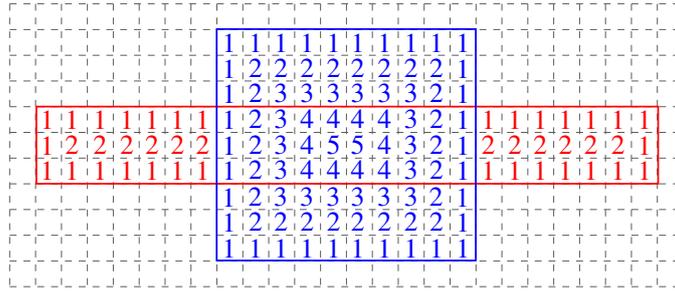


Abbildung 7.8: Beispiel für eine Umkehrung der Größenverhältnisse bei den relativen und den gewichteten relativen Größen des Schnittes zweier Areale: A_1 sei das rote Areal und A_2 das blaue. Dann ist $R(A_1 \cap A_2|A_1) = \frac{5}{12} > R(A_1 \cap A_2|A_2) = \frac{1}{3}$, aber $R_g(A_1 \cap A_2|A_1) = \frac{20}{47} < R_g(A_1 \cap A_2|A_2) = \frac{43}{95}$.

Q_1 enthält zu Beginn jedes Schleifendurchlaufs die Knoten, die gerade einen Wert k erhalten haben. Diese werden im aktuellen Durchlauf entnommen und ihre Nachbarn überprüft. Alle Nachbarn dieser Knoten, die noch keinen Wert erhalten haben, bekommen den Wert $k + 1$ und werden in Q_2 eingefügt. Wenn keine Knoten mehr in Q_1 enthalten sind, wird der Inhalt von Q_2 in Q_1 übertragen.

Algorithmus zur Berechnung der Knotengewichte:

Eingabe: Graph $G = (V, E)$, $A \subseteq V$

Ausgabe: Funktion $f_A : A \rightarrow \mathbb{N}$

initialisiere leere Queues Q_1, Q_2

für alle $a \in A$ sei $f(a)$ nicht definiert

für alle $a \in A$ {

 wenn ein $v \in N(a)$ mit $n \notin A$ existiert dann {

$f(a)=1$

 füge a in Q_1 ein

 }

}

solange $Q_1 \neq \emptyset$ {

 solange $Q_1 \neq \emptyset$ {

 entnehme den nächsten Knoten a aus Q_1

 für alle $v \in N(a)$ {

 wenn $f(v)$ ist noch nicht definiert und $n \in A$ dann {

 setze $f(v) = f(a) + 1$

 füge v in Q_2 ein

 }

 }

 }

 Setze $Q_1 = Q_2$

 Setze $Q_2 = \emptyset$

}

In einem Durchlauf der inneren Schleife werden also immer genau die Nachbarn der Knoten, die im letzten Durchlauf einen Wert bekommen haben, mit einem Wert versehen. Der Wert, der vergeben wird, ist während eines Durchlaufs der inneren Schleife immer derselbe: Im ersten Durchlauf 2, im zweiten 3, usw. Wenn der Algorithmus terminiert, haben alle Knoten einen Wert erhalten.

In Q_1 werden die Knoten verwaltet, die in dem aktuellen Durchlauf betrachtet werden. In Q_2 werden die Knoten verwaltet, die im nächsten Durchlauf betrachtet werden. Man kommt auch mit nur einer Queue und einer Schleife aus. Dazu fügt man die Knoten direkt in Q_1 ein und läßt Q_2 sowie die äußere Schleife einfach weg.

Jeder Knoten aus A (und alle seine Nachbarn) wird in der Vorphase einmal betrachtet. Außerdem wird jeder Knoten insgesamt einmal in eine Queue eingefügt und einmal entnommen und alle seine Nachbarn betrachtet. Insgesamt ergibt sich eine Laufzeit in $O(m \cdot k)$, wobei m die Anzahl der Knoten in A und k die Anzahl der zu Knoten in A inzidenten Kanten ist.

Die Ergebnisse der Berechnung von $R_g(A \cap B|A)$ und $R_g(A \cap B|B)$ für die Arealpaare, die auch im letzten Abschnitt verwendet wurden, sind in Abbildung 7.4 grafisch dargestellt. Die genauen Werte der einzelnen Beziehungen sind im Anhang in Tabelle B.6 zu finden. Dabei haben sich für einige Beziehungen die Größenverhältnisse geändert, das heißt es gibt einige Beziehungen zwischen Arealen A und B mit $R(A \cap B|A) > R(A \cap B|B)$ aber $R_g(A \cap B|A) < R_g(A \cap B|B)$. Dies ist auch für eine „Subarea“-Beziehung der Fall, so daß hier sogar sechs Datenpunkte über der Diagonalen liegen. Die Beziehungen, bei denen eine Umkehrung der Größenverhältnisse durch die Gewichtung aufgetreten ist, wurden in der Tabelle B.6 markiert.

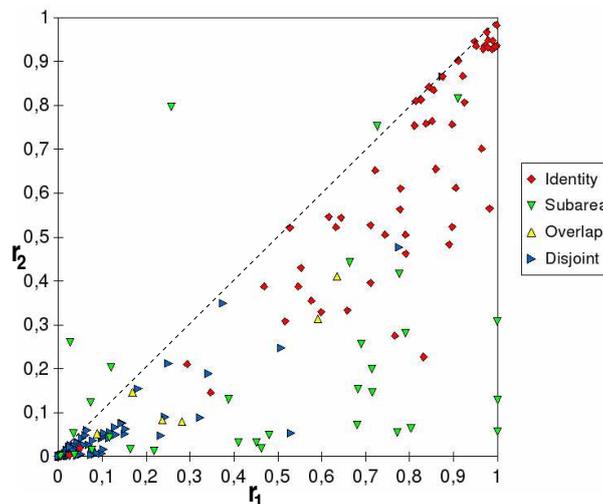


Abbildung 7.9: $r_1 = R_g(A \cap B|A)$ und $r_2 = R_g(A \cap B|B)$

Tatsächlich scheinen sich in diesem Fall die „Disjoint“-Punkte im Schnitt näher an $(0, 0)$, die „Identity-Punkte“ näher an $(1, 1)$ und die „Subarea“-Punkte näher an der Geraden $x = 1$ zu befinden. Es wurde wieder nach dem im letzten Abschnitt beschriebenen Verfahren Bereiche für die vier Punktgruppen berechnet. Die dabei aufgetretenen optimalen Fehlerwerte sind in Tabelle 7.4 zu sehen.

Im Vergleich zu dem ersten Verfahren ist der Fehler für die „Identity“-Punkte leicht gestiegen,

Beziehung	Fehler
Identity	26,76%
Subarea	43,75%
Overlap	33,33%
Disjoint	23,96%

Tabelle 7.4: Prozentsätze der Punkte, die bei einer optimalen Einteilung nicht richtig zugeordnet werden, geordnet nach Beziehungsart.

alle anderen Werte sind gleich geblieben (vergleiche Tabelle 7.3). Da die Optimierung versucht, die Werte mit Priorität nach steigender Größe zu optimieren, (der kleinste Wert also zuerst, danach der nächstgrößere), ist dieses als ein etwas schlechteres Ergebnis als das des ersten Verfahrens zu werten, da ja ein Fehlerwert größer ist. Die damit verbundenen Grenzen sind $0,61 \leq r_I \leq 0,62$, $r_D = 0,1$ und $0,62 \leq d_S \leq 0,71$. Hier lassen sich gleich zwei der Werte nicht eindeutig bestimmen, da in dem fraglichen Bereich keine relevanten Punkte liegen. Wir werden aber jeweils wieder den kleinsten Wert annehmen, da dieser am nächsten an dem theoretischen Wert von 0 liegt. Damit ist $r_I = 0,61$ und $d_S = 0,62$. Für eine grafische Darstellung der Ergebnisse siehe Abbildung 7.10.

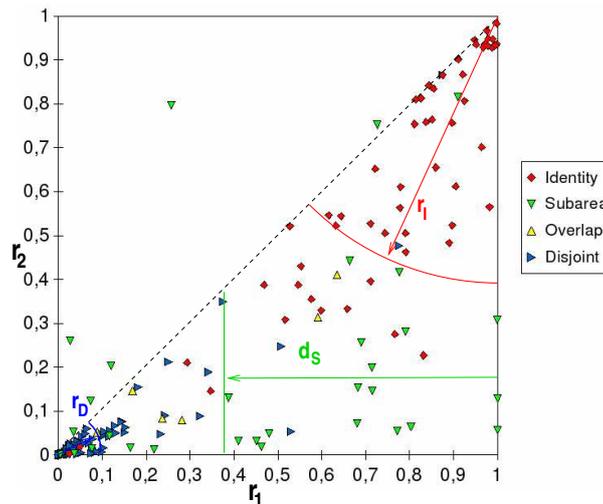


Abbildung 7.10: Grafische Darstellung der berechneten Bereiche für die einzelnen Beziehungen. $r_1 = R_g(A \cap B|A)$ und $r_2 = R_g(A \cap B|B)$.

Auch in diesem Fall wurden die sechs „Subarea“-Punkte, die über der Diagonalen liegen, als Fehler gezählt. Einer der Punkte hat zwar einen geringeren Abstand als d_S von der Geraden $x = 1$, aber auch einen geringeren Abstand als r_I von dem Punkt $(1,1)$. Die Bereiche für „Identity“ und „Disjoint“ sind deutlich kleiner geworden im Vergleich zum ersten Verfahren. Dagegen hat sich die Grenze für den Bereich „Subarea“ kaum geändert. Trotzdem sind aufgrund der Verkleinerung der beiden Radien die Bereiche „Subarea“ und „Overlap“ größer geworden, wobei der „Subarea“-Bereich mehr profitiert hat und jetzt rund die Hälfte der Dreieckfläche einnimmt. Trotz der kleineren Radien r_I und r_D liegen im ersten Fall nur ein Punkt weniger und im zweiten Fall sogar genau so viele „Identity“-Punkte beziehungsweise „Disjoint“-Punkte in den entsprechenden Bereichen wie vorher. Damit ist in Bezug auf

die theoretischen Überlegungen dieses Ergebnis besser als das des ersten Verfahrens. Das Ziel, für die Punkte eine Verteilung zu erhalten, die besser mit den Erwartungen übereinstimmt, wurde allerdings auch nur für die „Identity“- und die „Disjoint“-Punkte erreicht. Für die „Subarea“-Punkte scheint sich die Streuung der Punkte durch die Gewichtung eher noch vergrößert zu haben.

7.4.3 Verfahren 3

Die Grenzen der Bereiche für die „Identity“- und die „Disjoint“-Punkte als Radien um die Punkte $(1, 1)$ beziehungsweise $(0, 0)$ zu wählen, scheint gut zu funktionieren. Vielleicht ist aber nicht sinnvoll, die Grenze des „Subarea“-Bereiches als Abstand von der Geraden $x = 1$ zu wählen, auch wenn sich dieses an mathematischen Überlegung orientiert. Eine alternative Möglichkeit die Grenze zwischen dem „Subarea“-Bereich und dem „Overlap“-Bereich zu bestimmen, ergibt sich aus folgender Überlegung: Betrachte zwei Areale A und B , die einen gemeinsamen Schnitt haben (siehe auch Abbildung 7.11). Wenn man weiß, daß die Grenzen der Areale ungenau sind, ist man umso eher bereit zu sagen, daß A wahrscheinlich in B enthalten ist, je größer B im Verhältnis zu A ist. Dies ist unabhängig von der absoluten Größe des Schnittes. Und die Klassifikation der Beziehungen in CoCoMac sind durch menschliche Einschätzung entstanden.

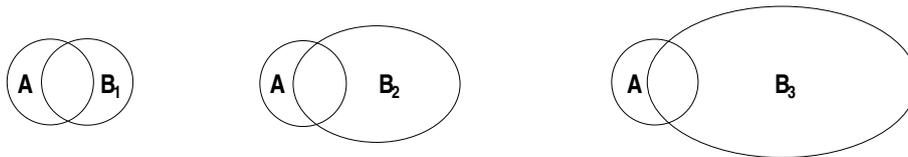


Abbildung 7.11: Ein Areal A , geschnitten mit den drei Arealen B_1 , B_2 und B_3 . Je größer das andere Areal wird, desto eher hat man den Eindruck, daß A in ihm enthalten ist. Der Schnitt ist aber in allen drei Fällen gleich groß.

Das heißt, je kleiner die relative Größe des Schnittes in Areal B ist, desto kleiner kann sie auch in Areal A werden, ohne daß die Beziehung „ A ist in B enthalten“ unwahrscheinlicher wird. Damit ergibt sich eine Abhängigkeit der Grenze des „Subarea“-Bereichs von beiden Koordinaten, nicht wie bisher nur von der Größe des Schnittes in A . Nach dieser Überlegung läßt sich die Grenze des „Subarea“-Bereichs durch eine Linie beschreiben, die mit der Geraden $x = 1$ einen Winkel α_S um den Punkt $(1, 1)$ bildet. Auch hier nehmen wir wieder an, daß alle Bereiche disjunkt sind, um eine eindeutige Zuordnung jedes Punktes zu ermöglichen. Eine grafische Darstellung einer solchen Aufteilung der Bereiche ist in Abbildung 7.12 zu sehen.

Es wurden analog zu dem im Abschnitt 7.4.1 beschriebenen Verfahren Bereiche für die vier Punktgruppen berechnet. Der Bereich für „Subarea“ wurde diesmal durch den Winkel α_S bestimmt, der in Schritten der Größe $0,5^\circ$ von 0° bis 45° erhöht wurde. Durch diese Wahl ergibt sich eine ähnliche Schrittweite wie für die bisherige Grenze, die in 100 Abstufungen verschoben wurde. Da keine Winkel von mehr als 45° betrachtet werden, liegen alle Punkte oberhalb der Winkelhalbierenden immer außerhalb des Gebietes für „Subarea“ und werden somit bei jeder Einteilung als Fehler gewertet. Die Berechnung wurde sowohl für die relativen Schnittgrößen als auch für die ungewichteten relativen Schnittgrößen durchgeführt. Die jeweiligen Fehler sind in Tabelle 7.5 zu sehen.

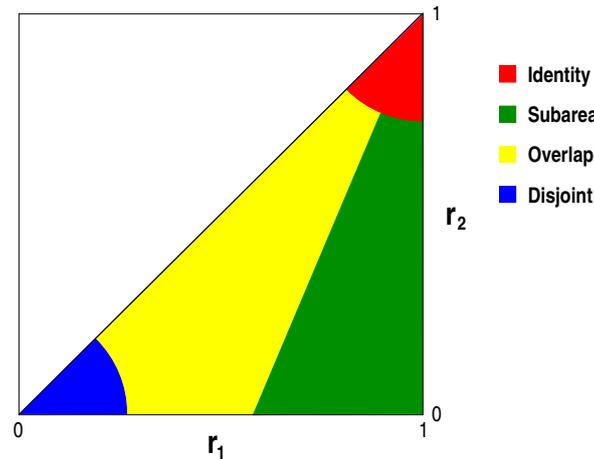


Abbildung 7.12: $r_1 = R_g(A \cap B|A)$ und $r_2 = R_g(A \cap B|B)$

Beziehung	Fehler	
	Schnitt ungewichtet	Schnitt gewichtet
Identity	28,57%	26,79%
Subarea	40,63%	43,75%
Overlap	16,67%	33,33%
Disjoint	23,96%	23,96%

Tabelle 7.5: Prozentsätze der Punkte, die bei einer optimalen Einteilung nicht richtig zugeordnet werden können, abhängig von der Gewichtung des Schnittes und der Art der Beziehung.

Wenn man diese Ergebnisse mit denen der bisherigen Einteilung in Tabelle 7.4 und Tabelle 7.3 vergleicht, sieht man, daß sich die Werte für die gewichteten Schnitte nicht unterscheiden. Dagegen tritt hier bei den ungewichteten Daten zwar eine leichte Verschlechterung für die „Identity“-Punkte, dafür aber auch eine leichte Verbesserung bei den „Subarea“-Punkten und eine deutliche Verbesserung bei den „Overlap“-Punkten ein. Da es von diesen nur sechs gibt, ist dieser deutliche Unterschied trotzdem nur auf einen Punkt zurückzuführen, der jetzt innerhalb des „Overlap“-Bereiches liegt. Genauso ist es bei den „Subarea“-Punkten, von denen befindet sich jetzt einer mehr im „Subarea“-Bereich. Dafür liegen jetzt zwei „Identity“-Punkte außerhalb des „Identity“-Bereiches, die bei der anderen Einteilung noch innerhalb lagen. Die mit den Fehlerwerten assoziierten Grenzen sind für die ungewichteten Daten $r_I = 0.71$, $r_D = 0.15$ und $37 \leq \alpha_S \leq 37,5$, sowie $r_I = 0.62$, $r_D = 0.1$ und $35.5 \leq \alpha_S \leq 37,5$ für die gewichteten Daten. Es wurde der jeweils kleinste mögliche Wert für α_S gewählt, da auch hier die Annahme vorliegt, daß der „Subarea“-Bereich so nah wie möglich an der Geraden $x = 1$ liegt. Ein grafische Darstellung der Ergebnisse ist in Abbildung 7.13 zu sehen.

Der Bereich für „Identity“ unterscheidet sich kaum und der für „Disjunkt“ gar nicht von den jeweiligen vorherigen Einteilungen für die relative Schnittgröße (Verfahren 1, vergleiche Abbildung 7.5) und die gewichtete relative Schnittgröße (Verfahren 2, vergleiche Abbildung 7.10). Dagegen hat die Größe des „Overlap“-Bereiches jeweils leicht zugenommen und damit die Größe des „Subarea“-Bereiches leicht abgenommen. Trotzdem scheint der „Subarea“-Bereich immer noch zu groß und der „Overlap“-Bereich zu klein, verglichen mit den theoretischen

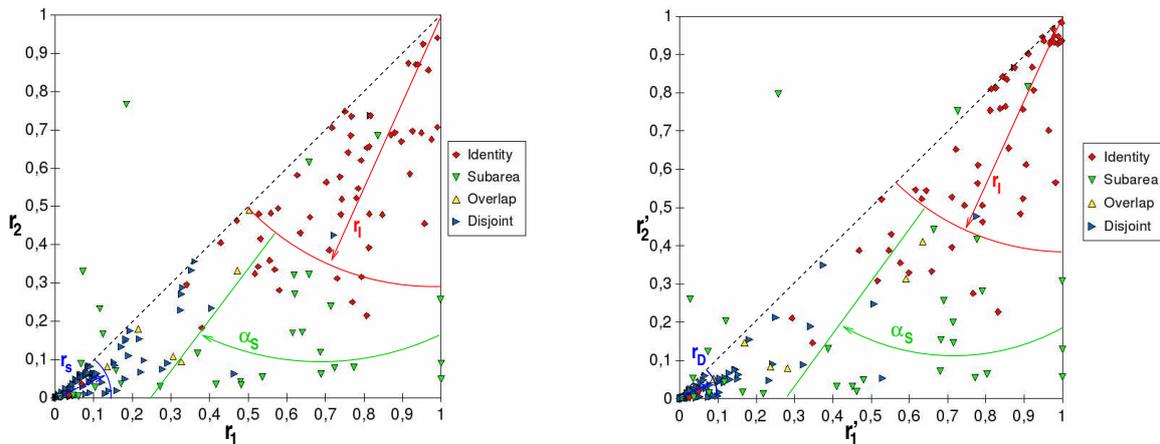


Abbildung 7.13: Grafische Darstellung der berechneten Bereiche für die einzelnen Beziehungen. $r_1 = R(A \cap B|A)$, $r_2 = R(A \cap B|B)$ und $r'_1 = R_g(A \cap B|A)$, $r'_2 = R_g(A \cap B|B)$

Überlegungen.

7.5 Diskussion

Die vorgestellten Verfahren liefern wichtige Hinweise auf Fehler in den verwendeten Datenbeständen. Alle Beziehungen von Arealpaaren, deren korrespondierenden Punkte nicht in dem Bereich liegen, zu dem sie laut CoCoMac gehören, sollten überprüft werden. Es ist wahrscheinlich, daß entweder die in CoCoMac registrierte Beziehung falsch ist, oder die Darstellung mindestens eines der Areale in CARET nicht der Definition des Areals entspricht und korrigiert werden sollte.

Wenn man neue Beziehungen zwischen Arealen aus verschiedenen Hirnkarten herleiten will, so kann eine Einordnung der relativen Größe der Schnitte in die vorgestellten Aufteilungen auch einen ersten Hinweis auf die Art der Beziehung geben. Diese sollte dann aber noch einmal von Hand überprüft werden. Allerdings kann man sich beim Registrieren neuer Beziehungen wohl auf die Areale beschränken, deren Schnittmenge von Rasterpunkten nicht leer ist. Bei allen anderen Arealpaaren können wir annehmen, daß sie disjunkt sind.

Das Verfahren eignet sich aber nicht für sichere Aussagen über die Beziehung von Arealen aufgrund ihrer Repräsentation aus dem CARET-System. Dafür sind die vier Punktmengen, die für die Beziehungsarten stehen, zu wenig voneinander abgrenzbar. Es lagen in allen drei vorgestellten Verfahren immer mehr als ein Viertel der Punkte außerhalb des Bereichs, zu dem sie eigentlich gehören sollten. Dies liegt aber wohl nicht an den Verfahren, sondern an der Menge und Qualität der Daten. Für eine verlässliche Berechnung der Bereiche bräuchte man mehr Daten, insbesondere zu „Overlap“- und „Subarea“-Beziehungen. Es besteht die Hoffnung, daß man mit mehr Daten die Bereiche besser definieren kann. Fehler würden zwar auch dann auftreten, aber wahrscheinlich nicht mehr so sehr ins Gewicht fallen. Sollte allerdings auch bei einer besseren Datenlage keine vernünftige Einteilung in disjunkte Bereiche möglich sein, so wäre das ein Grund, nach weiteren Verfahren der Einteilung zu suchen.

Vor allem die relative Größe der Schnitte von Arealen mit der Beziehung „Subarea“ scheint

nicht der Theorie zu entsprechen. Zum einen liegen die korrespondierenden Punkte teilweise oberhalb der Winkelhalbierenden, zum anderen sind sie auch unterhalb der Diagonalen weit verstreut. Dadurch lassen sich die einzelnen Bereiche für die vier Beziehungen schlecht abgrenzen. Eine Einteilung nur für die anderen drei Beziehungen fiel wohl eindeutiger aus. Allerdings ist auch auffällig, daß sich die „Overlap“-Punkte nur in einem kleinen Bereich nahe der Diagonalen befinden und nicht über den ganzen Datenraum verteilt sind. Dies kann dadurch zu erklären sein, daß es insgesamt nur sechs Punkte sind. Eine größere Datenmenge könnte sich anders verhalten. Für diesen kleinen Datensatz hingegen verhalten sich die „Overlap“-Punkte, wie man es nach der in Abschnitt 7.4.1 vorgestellten Theorie von den „Subarea“-Punkten erwarten würde und umgekehrt. Wenn sich aber ein größerer Datensatz genauso verhält, so wäre dieses ein starker Hinweis auf eine systematische Abweichung von der mathematischen Theorie. Um dieser Rechnung zu tragen, sollte man dann die Bereiche für die Beziehungsarten wohl nach dem dritten Verfahren aus Abschnitt 7.4.3 bestimmen. Die resultierende Einteilung spiegelt eine Verteilung der „Overlap“-Punkte nahe der Diagonalen wieder und ermöglicht gleichzeitig viel Raum für den „Subarea“-Bereich.

Die Verfahren wurden für eine Repräsentation der Areale als Rasterpunkte im zweidimensionalen Raum entwickelt. Ein ähnliches Vorgehen wäre aber auch für Datenpunkte im dreidimensionalen möglich. Für entsprechende Daten eines Netzwerkes im dreidimensionalen Raum gelten alle hier angeführten Überlegungen, da sie nur Mengenrelationen und Nachbarschaftsbeziehungen verwenden. Diese sind unabhängig von der Dimension des Raumes, in dem die Rasterpunkte liegen. Insgesamt können die beschriebenen Methoden für alle Daten verwendet werden, die als Knotenmenge beziehungsweise als ungerichtete Graphen dargestellt werden können.

Literaturverzeichnis

- [AKKW06] M. Abraham, R. Kötter, A. Krumnack, and E. Wanke. A connectivity rating for vertices in networks. In *Proceedings of the 4th IFIP International Conference on Theoretical Computer Science-TCS*, pages 283–298. Springer-Verlag, 2006.
- [BBK⁺02] P. Barone, A. Batardiere, K. Knoblauch, P. Giroud, M. Berland, A. Dumas, and H. Kennedy. Early specification of the hierarchical organization of visual cortical areas in the macaque monkey. *Cerebral Cortex*, 12(5):453–465, 2002.
- [BBKK00] P. Barone, A. Batardiere, K. Knoblauch, and H. Kennedy. Laminar distribution of neurons in extrastriate areas projecting to visual areas v1 and v4 correlates with the hierarchical rank and indicates the operation of a distance rule. *Journal of Neuroscience*, 20(9):3263–3281, 2000.
- [BLS99] A. Brandstädt, V.B. Le, and J.P. Spinrad. *Graph Classes: A Survey*. SIAM Monographs on Discrete Math. and Applications. Springer-Verlag, Berlin, Heidelberg, New York, 1999.
- [BM86] H.-J. Bandelt and H.M. Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, 41:182–208, 1986.
- [Bod98] H.L. Bodlaender. A partial k -arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209:1–45, 1998.
- [Bro05] K. Brodmann. Beiträge zur histologischen Lokalisation der Großhirnrinde. III. Mitteilung: Die Rindenfelder der niederen Affen. *Journal für Psychologie und Neurologie*, 4:177–226, 1905.
- [Bro09] K. Brodmann. *Vergleichende Lokalisationslehre der Großhirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues*. Barth, Leipzig, 1909.
- [BWKK08] G. Bezgin, E. Wanke, A. Krumnack, and R. Kötter. Deducing logical relationships between spatially registered cortical parcellations under conditions of uncertainty. *submitted to Neural Networks*, 2008.
- [CK98] F. Crick and C. Koch. Constraints on cortical and thalamic projections: the no-strong-loops hypothesis. *Nature*, 391(6664):245–250, 1998.
- [Dan63] G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, N.J., 1963.

- [DBCM06] S.G. Dickstein, K. Bannon, F. Xavier Castellanos, and M.P. Milham. The neural correlates of attention deficit hyperactivity disorder: an ale meta-analysis. *J. Child Psychol. Psychiatry*, 47:1051–1062, 2006.
- [DDE01] J. Dickson, H. Drury, and D. Van Essen. 'the surface management system' (sums) database: a surface-based database to aid cortical surface reconstruction, visualization and analysis. *Philosophical Transactions of the Royal Society London, Biological Sciences*, 356:1277–1292, 2001.
- [Dir61] G. Dirac. On rigid circuit graphs. *Abh. Math. Sem. Univ. Hamburg*, 25:71–76, 1961.
- [DP94] X. Deng and C.H. Papadimitriou. On the complexity of cooperative solution concepts. *Methods of Operations Research*, 19(2):257–266, 1994.
- [Ess04] D.C. Van Essen. Surface-based approaches to spatial localization and registration in primate cerebral cortex. *NeuroImage*, 23:97–107, 2004.
- [FE91] D.J. Felleman and D.C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, 1:1–47, 1991.
- [FG65] D.R. Fulkerson and O.A. Gross. Incidence matrices and interval graphs. *Pacific J. Math.*, 15:835–855, 1965.
- [GGAM⁺04] D. Gómez, E. González-Arangüena, C. Manuel, G. Owen, M. del Pozo, and J. Tejada. Splitting graphs when calculating Myerson value for pure overhead games. *Mathematical Methods of Operations Research*, 59:479–489, 2004.
- [GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, San Francisco, 1979.
- [GSW95] G.J. Goodhill, M.W. Simmen, and D.J. Willshaw. An evaluation of the use of multidimensional scaling for understanding brain connectivity. *Philosophical Transactions of the Royal Society London, Biological Sciences*, 348(1325):265–280, 1995.
- [HKSS02] C.C. Hilgetag, R. Kötter, K.E. Stephan, and O. Sporns. Computational methods for the analysis of brain connectivity. In G. A. Ascoli, editor, *Computational Neuroanatomy- Principles and Methods*, pages 295–335, Totowa, NJ, 2002. Humana Press.
- [HM90] P.L. Hammer and F. Maffray. Completely separable graphs. *Discrete Applied Mathematics*, 27:85–99, 1990.
- [HOY96] C.C. Hilgetag, M.A. O'Neil, and M.P. Young. Indeterminante organization of the visual system. *Science*, 271:776–777, 1996.
- [HS58] A. Hajnal and J. Surányi. Über die Auflösung von Graphen in vollständige Teilgraphen. *Ann. Univ. Sci. Budapest, Eötvös Sect. Math.*, 1:113–121, 1958.
- [KÖ4] R. Kötter. Online retrieval, processing, and visualization of primate connectivity data from the CoCoMac database. *Neuroinformatics*, 2004.

- [KAB⁺00] C. Köbbert, R. Apps, I. Bechmann, J.L. Lanciego, J. Mey, and S. Thanos. Current concepts in neuroanatomical tracing. *Progress in Neurobiology*, 62(4):327–351, 2000.
- [Kar78] R.M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23:309–311, 1978.
- [Kar84] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.
- [KB85] H. Kennedy and J. Bullier. A double-labelling investigation of the afferent connectivity to cortical areas v1 and v2 of the macaque monkey. *The Journal of neuroscience*, 5:2815–2830, 1985.
- [KH04] M. Kaiser and C.C. Hilgetag. Edge vulnerability in neural and metabolic networks. *Biological Cybernetics*, 90:311–317, 2004.
- [KMAY07] M. Kaiser, R. Martin, P. Andras, and M.P. Young. Simulation of robustness against lesions of cortical networks. *European Journal of Neuroscience*, 25:3185–3192, 2007.
- [KRK⁺07] R. Kötter, A. T. Reid, A. Krumnack, E. Wanke, and O. Sporns. Shapley ratings in brain networks. *Frontiers in Neuroinformatics*, 2007.
- [KS03] R. Kötter and K.E. Stephan. Network participation indices: characterizing component roles for information processing in neural networks. *Neural Networks*, 16:1261–1275, 2003.
- [KSH⁺04] A. Keinan, B. Sandbank, C.C. Hilgetag, I. Meilijson, and E. Ruppin. Fair attribution of functional contribution in artificial and biological networks. *Neural Comput.*, 16:1887–1915, 2004.
- [KW05] R. Kötter and E. Wanke. Mapping brains without coordinates. *Philosophical Transactions of the Royal Society London, Biological Sciences*, 360:751–766, 2005.
- [LE00] J.W. Lewis and D.C. Van Essen. Architectonic parcellation of parieto-occipital cortex and interconnected cortical regions in the macaque monkey. *J. Comp. Neurol.*, 428:79–111, 2000.
- [Mey68] T. Meynert. *Der Bau der Großhirnrinde und seine örtlichen Verschiedenheiten, nebst einem pathologisch-anatomischen Corollarium*. Engelmann, Leipzig, 1868.
- [MG95] A.D. Milner and M.A. Goodale. *The visual brain in action*. Oxford University Press, Oxford, 1995.
- [MUM83] M. Mishkin, L.G. Ungerleider, and K.A. Macko. Object vision and spatial vision: two cortical pathways. *Trends in Neurosciences*, 6:414–417, 1983.
- [Mye77] R.B. Myerson. Graphs and cooperations in games. *Methods of Operations Research*, 2:255–229, 1977.

- [NP04] S.D. Nikolopoulos and L. Palios. Hole and antihole detection in graphs. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pages 850–859. ACM-SIAM, 2004.
- [Owe86] G. Owen. Values of graph-restricted games. *SIAM Journal of Algebraic and Discrete Methods*, 7(2):210–220, 1986.
- [PGR91] T.M. Preuss and P.S. Goldman-Rakic. Myelo- and cytoarchitecture of the granular frontal cortex and surrounding regions in the strepsirhine primate galago and the anthropoid primate macaca. *J. Comp. Neurol.*, 310:429–474, 1991.
- [PHT00] G. Paxinos, Xu-F. Huang, and A.W. Toga. *The Rhesus Monkey Brain in Stereotaxic Coordinates*. Academic Press, San Diego, 2000.
- [PP01] M. Petrides and D.N. Pandya. Comparative cytoarchitectonic analysis of the human and the macaque ventrolateral prefrontal cortex and corticocortical connection patterns in the monkey. *European Journal of Neuroscience*, 16:291–310, 2001.
- [PS98] C.H. Papadimitiou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications Inc., Mineola, New York, 1998.
- [PSK02] R.E. Passingham, K.E. Stephan, and R. Kötter. The anatomical basis of functional localization in the cortex. *Nature Reviews Neuroscience*, 3:606–616, 2002.
- [Ros70] D.J. Rose. Triangulated graphs and elimination process. *J. Math. Analys. Appl.*, 32:597–609, 1970.
- [RS86] N. Robertson and P.D. Seymour. Graph minors II. Algorithmic aspects of tree width. *Journal of Algorithms*, 7:309–322, 1986.
- [RTL76] D.J. Rose, R.E. Trajan, and G.S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5:266–283, 1976.
- [SCKH04] O. Sporns, D. Chialvo, M. Kaiser, and C.C. Hilgetag. Organization, development and function of complex brain networks. *Trends Cogn. Sci.*, 8:418–425, 2004.
- [SGW94] M.W. Simmen, G.J. Goodhill, and D.J. Willshaw. Scaling and brain connectivity. *Nature*, 369:448–449, 1994.
- [SH⁺00] K.E. Stephan, C.C. Hilgetag, , G.A.P.C. Burns, M.A. O’Neil, M.P. Young, and R. Kötter. Computational analysis of funktional connectivity between areas of primate cerebral cortex. *Philosophical Transactions of the Royal Society London, Biological Sciences*, 355:111–126, 2000.
- [Sha53] L.S. Shapley. A value for n -person games. In H.W. Kuhn and A.W. Tucker, editors, *Contributions to the Theory of Games II*, pages 307–317, Princeton, 1953. Princeton University Press.
- [SHK07] O. Sporns, C. Honey, and R. Kötter. Identification and classification of hubs in brain networks. *PLoS ONE*, 10:e1049, 2007.

- [SKB⁺01] K.E. Stephan, L. Kamper, A. Bozkurt, G.A.P.C. Burns, M.P. Young, and R. Kötter. Advanced database methodology for the collation of connectivity data on the macaque brain (CoCoMac). *Philosophical Transactions of the Royal Society London, Biological Sciences*, 356:1159–1186, 2001.
- [Spo03] O. Sporns. Graph theory methods for the analysis of neural connectivity patterns. In R. Kötter, editor, *Neuroscience Databases: a Practical Guide*, pages 171–186, Boston, 2003. Küwer.
- [Tar85] R. Tarjan. Decomposition by clique separators. *Discrete Mathematics*, 55:221–232, 1985.
- [TY84] R.E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test chordality of graphs, acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM Journal on Computing*, 13:566–579, 1984.
- [UD86] L.G. Ungerleider and R. Desimone. Cortical connections of visual area mt in the macaque. *J. Comp. Neurol.*, 248:190–222, 1986.
- [UM82] L.G. Ungerleider and M. Mishkin. Two cortical visual systems. In D.G. Ingle, M.A. Goodale, and R.J.Q. Mansfield, editors, *Analysis of Visual Behavior*, pages 549–586. MIT, Cambridge, MA, 1982.
- [vBB47] G. von Bonin and P. Bailey. *The neocortex of Macaca mulatta*. University of Illinois Press, Urbana, Illinois, 1947.
- [vdBP02] R. van den Brink and P. Borm. Digraph competitions and cooperative games. *Theory and Decision*, 53:327–342, 2002.
- [VFJ⁺04] J. Vezoli, A. Falchier, B. Jouve, K. Knoblauch, M. Young, and H. Kennedy. Quantitative analysis of connectivity in the visual cortex: Extracting function from structure. *The Neuroscientist*, 10(5):476–482, 2004.
- [VV19] C. Vogt and O. Vogt. Allgemeinere ergebnisse unserer hirnforschung. 1.-4. mitteilung. *Journal für Psychologie und Neurologie*, 25:279–461, 1919.
- [Wal40] A.E. Walker. A cytoarchitectural study of the prefrontal areas of macaque monkey. *J. Comp. Neurol.*, 98:59–86, 1940.
- [Wie87] M. Wiegens. Recognizing outerplanar graphs in linear time. In *Proceedings of Graph-Theoretical Concepts in Computer Science*, volume 246 of *LNCS*, pages 165–176. Springer-Verlag, 1987.
- [You92] M.P. Young. Objective analysis of the topological organization of the primate cortical visual system. *Nature*, 358(6382):152–155, 1992.
- [You93] M.P. Young. The organization of neural systems in the primate cerebral cortex. *Proceedings: Royal Society of London, Biological Sciences*, 252:13–18, 1993.
- [Zil04] K. Zilles. Architecture of the Human Cerebral Cortex. Regional and Laminar Organization. In G. Paxinos and J.K. Mai, editors, *The Human Nervous System*, pages 997–1055, San Diego, CA, 2004. Elsevier. 2nd edition.

LITERATURVERZEICHNIS

- [ZR98] K. Zilles and G. Rehkämper. *Funktionelle Neuroanatomie: Lehrbuch und Atlas*. Springer, Berlin, 1998.

Anhang A

Verwendete Verbindungsdaten

	V1	V2	V3	Vp	V3a	V4	VOT	V4t	MT	MSTd	MSTl	FST	PITd	PITv	CITd	CITv	AITd	AITv	STPp	STPa	TF	TH	PO	PIP	LIP	VIP	DP	A7a	FEF	A46
V1	-	1	1	0	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V2	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
V3	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Vp	0	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
V3a	1	1	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
V4	1	1	1	1	1	-	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
VOT	0	1	0	1	0	1	-	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
V4t	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MT	1	1	1	1	1	1	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MSTd	0	1	1	1	1	1	0	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MSTl	0	1	0	0	1	0	0	0	1	0	-	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FST	0	0	0	0	0	1	0	1	1	1	1	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
PITd	0	0	0	0	0	0	0	0	0	0	0	0	-	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
PITv	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
CITd	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CITv	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AITd	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AITv	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
STPp	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
STPa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PIP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LIP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VIP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A7a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FEF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A46	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabelle A.1: Verbindungsmatrix des visuellen Systems beim Makaken nach Young [You92].

W40	10	11	12	13	14	24	25	45	46	8A	8B	9
10	-	X	3	1	3	X	3	0	0	X	0	X
11	3	-	3	3	X	X	X		0	0	0	3
12	3	3	-	3	3	1	3	0	X	X	X	X
13	1	X	3	-	3	1	3	0	0	0	X	X
14	3	X	X	3	-	X	3	0	0	0	0	X
24	0	3	1	0	X	-	3	1	1	1	X	3
25	3	X	3	2	3	3	-	0	X	0	X	3
45	1	3	3	1	1	1	0	-	X	2	X	X
46	3	2	X	0	0	1	1	2	-	X	X	X
8A	X	0	2	0	0	1	0	2	X	-		X
8B						3			X	X	-	X
9	X	X	X	1	0	3	X	X	X	X	X	-

Tabelle A.2: Verbindungsmatrix für den präfrontalen Cortex W40 [Wal40] zusammengestellt aus der CoCoMac-Datenbank nach [SKB⁺01]. 0 steht für nachgewiesen abwesene Verbindungen, X für vorhandene Verbindungen einer unbestimmten Stärke und die Zahlen 1, 2 und 3 für Verbindungen geringer, mittlerer und großer Stärke. Nicht gefüllte Felder stehen für einen ungeklärten Verbindungszustand.

W40 - 0	10	11	12	13	14	24	25	45	46	8A	8B	9
10	-	1	1	1	1	1	1	0	0	1	0	1
11	1	-	1	1	1	1	1	0	0	0	0	1
12	1	1	-	1	1	1	1	0	1	1	1	1
13	1	1	1	-	1	1	1	0	0	0	1	1
14	1	1	1	1	-	1	1	0	0	0	0	1
24	0	1	1	0	1	-	1	1	1	1	1	1
25	1	1	1	1	1	1	-	0	1	0	1	1
45	1	1	1	1	1	1	0	-	1	1	1	1
46	1	1	1	0	0	1	1	1	-	1	1	1
8A	1	0	1	0	0	1	0	1	1	-	0	1
8B	0	0	0	0	0	1	0	0	1	1	-	1
9	1	1	1	1	0	1	1	1	1	1	1	-

Tabelle A.3: Verbindungsmatrix für den präfrontalen Cortex, Version W40-0. Hier wurden alle unbekanntenen Verbindungen als nicht vorhanden gewertet. Alle vorhandenen Verbindungen sind durch eine 1 gekennzeichnet, alle nicht-vorhandenen durch eine 0.

W40 – 1	10	11	12	13	14	24	25	45	46	8A	8B	9
10	-	1	1	1	1	1	1	0	0	1	0	1
11	1	-	1	1	1	1	1	1	0	0	0	1
12	1	1	-	1	1	1	1	0	1	1	1	1
13	1	1	1	-	1	1	1	0	0	0	1	1
14	1	1	1	1	-	1	1	0	0	0	0	1
24	0	1	1	0	1	-	1	1	1	1	1	1
25	1	1	1	1	1	1	-	0	1	0	1	1
45	1	1	1	1	1	1	0	-	1	1	1	1
46	1	1	1	0	0	1	1	1	-	1	1	1
8A	1	0	1	0	0	1	0	1	1	-	1	1
8B	1	1	1	1	1	1	1	1	1	1	-	1
9	1	1	1	1	0	1	1	1	1	1	1	-

Tabelle A.4: Verbindungsmatrix für den präfrontalen Cortex , Version W40-1. Hier wurden alle unbekanntes Verbindungen als vorhanden gewertet. Alle vorhandenen Verbindungen sind durch eine 1 gekennzeichnet, alle nicht-vorhandenen durch eine 0.

W40 – 2 neu	10	11	12	13	14	24	25	45	46	8A	8B	9
10	-	2	3	1	3	X	3	0	X	X	X	3
11	3	-	3	3	X	X	2	0	X	0	0	3
12	3	X	-	3	3	1	3	X	X	X	X	2
13	1	X	3	-	3	1	3	0	X	0	X	X
14	3	X	3	3	-	X	X	0	0	0	0	X
24	0	2	X	X	X	-	3	0	3	1	0	3
25	3	X	X	2	X	2	-	0	X	0	X	3
45	1	3	3	1	1	1	0	-	X	X	X	X
46	3	2	X	0	0	3	1	2	-	X	3	X
8A	0	X	2	0	0	1	0	X	X	-	0	X
8B	0	X	X	0	0	0	X	X	X	X	-	X
9	X	X	X	1	0	X	X	X	X	X	X	-

Tabelle A.5: Aktualisierte Version der Verbindungsmatrix für den präfrontalen Cortex W40 [Wal40] zusammengestellt aus der CoCoMac-Datenbank. Ob die Verbindung 11 → 45 existiert ist nicht geklärt. Hier wurde sie als nicht vorhanden gewertet.

Tabelle A.6: Verfeinerung der Klassifizierung der Verbindungen im visuellen System nach Felleman und Van Essen [FE91]. Anhand der verschiedenen laminaren Muster wurden die Verbindungen in fünf Klassen eingeteilt. Bei supragranularem Ursprung (S) und Ziel in Schicht 4 (F) wurde die Verbindung als stark aufsteigend (A+), bei bilaminarem Ursprung (B) und Ziel in Schicht 4 (F) als aufsteigend (A), bei bilaminarem Ursprung (B) und Ziel allen Schichten (C) als lateral (L), bei bilaminarem Ursprung (B) und Ziel in supra- und infragranularen Schichten (M) als absteigend (D) und bei infragranularem Ursprung (I) und Ziel in supra- und infragranularen Schichten (M) als stark absteigend (D+) klassifiziert. (Vergleiche dazu auch Abschnitt 5.1 und den Abschnitt 5.4.2.) Kann eine Verbindung nicht eindeutig zu einer Klasse zugeordnet werden, so wird sie zu allen in Frage kommenden Klassen gezählt. Richtung gibt an, welche Zuordnung Felleman und Van Essen vorgenommen haben. Dabei steht u für unbekannte Richtung bezüglich der Hierarchie.

Von	Nach	laminares Muster		Richtung in [FE91] (A,L,D)	D+ (I→M)	D (B→M)	L (B→C)	A (B→F)	A+ (S→F)
		Ursprung (S,B,I)	Ziel (F,C,M)						
VP	V4	B	F	A	0	0	0	2	0
V3A	V4	B	F	A	0	0	0	2	0
V3A	MT	B	F	A	0	0	0	2	0
V3A	PO	B		u	0	1	1	1	0
PIP	V4	B		u	0	1	1	1	0
PIP	7a	B		u	0	1	1	1	0
PO	MT	B		u	0	1	1	1	0
V4	V4t	B	F/C	L/A?	0	0	1	1	0
V4	DP	B		u	0	1	1	1	0
V4t	MT	B		u	0	1	1	1	0
MIP	7a	B		u	0	1	1	1	0
MDP	7a	B		u	0	1	1	1	0
DP	7a	B	F	A	0	0	0	2	0
FST	MSTl	B	C	L	0	0	2	0	0
FST	LIP	B	C/M?	L/D?	0	1	2	0	0
FST	7a	B		u	0	1	1	1	0
VIP	MSTd	B		u	0	1	1	1	0
VIP	MSTl	B		u	0	1	1	1	0
VIP	FEF	B		u	0	1	1	1	0
VIP	7a	B		u	0	1	1	1	0
LIP	MSTd	B	C/F	L/A?	0	0	1	1	0
LIP	MSTl	B		u	0	1	1	1	0
LIP	7a	B		u	0	1	1	1	0
MSTd	7a	B	C	L	0	0	2	0	0
7a	FEF	B	C/M?	L/D?	0	1	2	0	0
7a	46	B	F/C	A/L?	0	1	1	2	0
FEF	STPp	B	M(s)	D?	0	1	0	0	0
FEF	46	B	F/C	A/L?	0	1	1	2	0
V2	V1	B	M	D	0	2	0	0	0
V3	V1	B	M	D	0	2	0	0	0
V4	V1	B	M(s)	D?	0	1	0	0	0
VP	V2	B	M	D	0	2	0	0	0

Fortsetzung auf der nächsten Seite

Anhang A. Verwendete Verbindungsdaten

Von	Nach	laminares Ursprung (S,B,I)	Muster Ziel (F,C,M)	Richtung in [FE91] (A,L,D)	D+ (I→M)	D (B→M)	L (B→C)	A (B→F)	A+ (S→F)
V4	V2	B	M	D	0	2	0	0	0
MT	V2	B	M	D	0	2	0	0	0
VP	V3	B		u	0	1	1	1	0
PIP	V3	B		u	0	1	1	1	0
V4t	V3	B		u	1	1	1	1	1
MT	V3	B	M	D	0	2	0	0	0
V3A	VP	B		u	0	1	1	1	0
PIP	VP	B		u	0	1	1	1	0
V4t	PO	B		u	0	1	1	1	0
MT	PO	B	C/M	L/D?	0	1	2	0	0
MIP	PO	B		u	0	1	1	1	0
MDP	PO	B		u	0	1	1	1	0
MSTd	PO	B	M	D	0	2	0	0	0
LIP	PO	B	C/F	L/A?	0	0	1	1	0
VIP	PO	B		u	0	1	1	1	0
7a	PO	B	M(I)	D?	0	1	0	0	0
V4t	V4	B		u	0	1	1	1	0
MT	V4	B	C/M	L/D?	0	1	2	0	0
VOT	V4	B	M	D	0	2	0	0	0
LIP	V4	B	M	D	0	2	0	0	0
PITd	V4	B	M	D	0	2	0	0	0
PITv	V4	B	M	D	0	2	0	0	0
MT	V4t	B	C	L	0	0	2	0	0
MSTd	V4t	B	M	D?	0	1	0	0	0
FST	V4t	B	M	D	0	2	0	0	0
VIP	MT	B		u	0	1	1	1	0
LIP	DP	B	C/M	L/D?	0	1	2	0	0
MSTd	DP	B	C	L	0	0	2	0	0
7a	DP	B	M	D	0	2	0	0	0
MSTd	FST	B	M/C	L/D?	0	1	2	0	0
MSTl	FST	B	C/F	L/A?	0	0	1	1	0
PITd	FST	B		u	0	1	1	1	0
PITv	FST	B		u	0	1	1	1	0
LIP	FST	B		u	0	1	1	1	0
VIP	FST	B		u	0	1	1	1	0
MSTd	LIP	B	C	L?	0	0	1	0	0
7a	MSTd	B	M	D	0	2	0	0	0
PITd	MSTd	B		u	0	1	1	1	0
PITv	MSTd	B		u	0	1	1	1	0
7a	MSTl	B		u	0	1	1	1	0
AITd	7a	B		u	0	1	1	1	0
FEF	7a	B		u	0	1	1	1	0
STPp	FEF	B	M(s)	D?	0	1	0	0	0
46	FEF	B		u	0	1	1	1	0
STPa	AITd	B		u	0	1	1	1	0
TF	46	B		u	0	1	1	1	0
MSTd	MT	B/I	M	D	2	2	0	0	0
MSTl	MT	B/I	M	D	2	2	0	0	0
STPa	7a	B/I		u	1	1	1	1	0

Fortsetzung auf der nächsten Seite

Von	Nach	laminares Muster		Richtung in [FE91] (A,L,D)	D+	D	L	A	A+
		Ursprung (S,B,I)	Ziel (F,C,M)						
STPp	7a	B/I		u	1	1	1	1	0
TF	FST	B/S		A?	0	1	1	1	1
46	7a	B/S	M	D?	0	2	0	1	1
V3A	V1	I		D	1	0	0	0	0
V4t	V1	I		D	1	0	0	0	0
MT	V1	I	M	D	2	0	0	0	0
V3A	V3	I	M	D	2	0	0	0	0
V4	V3	I	M	D	2	0	0	0	0
MSTd	V3	I		D	1	0	0	0	0
LIP	V3	I		D	1	0	0	0	0
VIP	V3	I		D	1	0	0	0	0
TF	V3	I		D	1	0	0	0	0
V4	VP	I		D	1	0	0	0	0
MT	VP	I	M	D	2	0	0	0	0
MSTd	VP	I		D	1	0	0	0	0
TF	VP	I		D	1	0	0	0	0
MT	V3A	I	C/M	D/L?	2	0	1	0	0
LIP	V3A	I	M	D	2	0	0	0	0
CITd	V4	I		D	1	0	0	0	0
CITv	V4	I		D	1	0	0	0	0
TF	V4	I	M	D	2	0	0	0	0
TH	V4	I		D	1	0	0	0	0
FST	MT	I	M	D	2	0	0	0	0
FEF	FST	I		D	1	0	0	0	0
STPp	FST	I		D	1	0	0	0	0
TF	MSTd	I		D	1	0	0	0	0
FEF	MSTd	I		D	1	0	0	0	0
FEF	MSTl	I		D	1	0	0	0	0
TH	PITv	I		D	1	0	0	0	0
TH	CITd	I		D	1	0	0	0	0
TH	CITv	I		D	1	0	0	0	0
TF	CITv	I		D	1	0	0	0	0
TH	7a	I		D	1	0	0	0	0
STPa	STPp	I	M	D	2	0	0	0	0
TF	STPp	I		D	1	0	0	0	0
TH	STPp	I		D	1	0	0	0	0
TF	STPa	I		D	1	0	0	0	0
TH	STPa	I		D	1	0	0	0	0
TF	AITv	I		D	1	0	0	0	0
TH	AITv	I		D	1	0	0	0	0
STPp	MSTd	I/B		D	1	1	0	0	0
STPp	MSTl	I/B		D?	1	1	0	0	0
CITd	PITv	I/B		D?	1	1	0	0	0
CITv	PITv	I/B		D?	1	1	0	0	0
AITd	PITv	I/B		D?	1	1	0	0	0
AITv	PITv	I/B		D?	1	1	0	0	0
TF	PITv	I/B		D?	1	1	0	0	0
AITd	CITv	I/B		D?	1	1	0	0	0
AITv	CITv	I/B		D?	1	1	0	0	0

Fortsetzung auf der nächsten Seite

Anhang A. Verwendete Verbindungsdaten

Von	Nach	laminares Muster		Richtung in [FE91] (A,L,D)	D+	D	L	A	A+
		Ursprung (S,B,I)	Ziel (F,C,M)						
TF	7a	I/B		D?	1	1	0	0	0
TH	46	I/B		D?	1	1	0	0	0
V1	V2	S	F	A	0	0	0	0	2
V1	V3	S	F	A	0	0	0	0	2
V1	V4	S		A	0	0	0	0	1
V1	MT	S	F	A	0	0	0	0	2
V1	PO	S		A	0	0	0	0	1
V1	MSTl	S		A	0	0	0	0	1
V2	V3	S	F	A	0	0	0	0	2
V2	VP	S	F	A	0	0	0	0	2
V2	V3A	S	F	A	0	0	0	0	2
V2	V4	S	F	A	0	0	0	0	2
V2	V4t	S	F	A	0	0	0	0	2
V2	MT	S	F	A	0	0	0	0	2
V2	PO	S		A	0	0	0	0	1
V2	MSTd	S	F	A	0	0	0	0	2
V2	MSTl	S		A	0	0	0	0	1
V2	FST	S		A?	0	0	0	0	1
V2	FEF	S		A	0	0	0	0	1
V3	V4	S	F	A	0	0	0	0	2
V3	V4t	S	F	A	0	0	0	0	2
V3	MT	S	F	A	0	0	0	0	2
V3	PO	S		A	0	0	0	0	1
V3	MSTd	S	F	A	0	0	0	0	2
V3	FST	S		A	0	0	0	0	1
V3	FEF	S		A	0	0	0	0	1
VP	MT	S	F	A	0	0	0	0	2
VP	PO	S		A	0	0	0	0	1
VP	MSTd	S	F	A	0	0	0	0	2
VP	FST	S	F	A	0	0	0	0	2
VP	FEF	S		A	0	0	0	0	1
V3A	FST	S		A	0	0	0	0	1
V3A	FEF	S		A	0	0	0	0	1
PIP	PO	S		A	0	0	0	0	1
PO	MSTd	S		A	0	0	0	0	1
PO	MSTl	S		A	0	0	0	0	1
PO	LIP	S		A	0	0	0	0	1
PO	FEF	S		A	0	0	0	0	1
V4	FST	S		A	0	0	0	0	1
V4	PITd	S	F	A	0	0	0	0	2
V4	PITv	S	F	A	0	0	0	0	2
V4	CITd	S	F	A	0	0	0	0	2
V4	CITv	S	F	A	0	0	0	0	2
V4	AITv	S		A	0	0	0	0	1
V4	FEF	S		A?	0	0	0	0	1
V4	46	S		A	0	0	0	0	1
V4t	MSTl	S		A?	0	0	0	0	1
V4t	FST	S		A	0	0	0	0	1
V4t	FEF	S		A	0	0	0	0	1

Fortsetzung auf der nächsten Seite

Von	Nach	laminares Muster		Richtung in [FE91] (A,L,D)	D+	D	L	A	A+
		Ursprung (S,B,I)	Ziel (F,C,M)						
MT	FEF	S	F	A	0	0	0	0	2
MT	46	S		A	0	0	0	0	1
DP	MSTl	S		A?	0	0	0	0	1
FST	FEF	S	F/M	A/D?	0	1	0	1	2
LIP	FEF	S	F	A	0	0	0	0	2
LIP	46	S	F	A	0	0	0	0	2
MSTd	FEF	S	F	A	0	0	0	0	2
MSTl	FEF	S	F	A	0	0	0	0	2
PITd	FEF	S		A	0	0	0	0	1
PITv	FEF	S		A	0	0	0	0	1
PITd	46	S		A	0	0	0	0	1
PITv	46	S		A	0	0	0	0	1
CITd	FEF	S		A	0	0	0	0	1
CITv	FEF	S		A	0	0	0	0	1
CITd	46	S		A	0	0	0	0	1
CITv	46	S		A	0	0	0	0	1
FEF	AITd	S		A	0	0	0	0	1
STPp	STPa	S	F	A	0	0	0	0	2
STPp	46	S	C	A/L?	0	0	1	0	1
STPa	46	S		A	0	0	0	0	1
46	AITd	S		A	0	0	0	0	1
46	STPp	S	M	A/D?	0	1	0	0	1
V3A	DP	S/B		A?	0	1	1	1	1
V3A	MSTd	S/B		A?	0	1	1	1	1
V3A	MSTl	S/B		A?	0	1	1	1	1
PIP	DP	S/B		A?	0	1	1	1	1
PO	7a	S/B		A?	0	1	1	1	1
V4	MT	S/B	C	L/A?	0	1	2	1	0
MT	MSTd	S/B	F	A	0	0	0	2	2
MT	MSTl	S/B	F	A	0	0	0	2	2
MT	FST	S/B	F	A	0	0	0	2	2
DP	MSTd	S/B		A?	0	1	1	1	1
FST	MSTd	S/B	C	L/A?	0	1	2	1	0
PITd	CITv	S/B		A?	0	1	1	1	1
PITd	AITd	S/B		A?	0	1	1	1	1
PITd	AITv	S/B		A?	0	1	1	1	1
PITv	CITv	S/B		A?	0	1	1	1	1
PITv	AITv	S/B		A?	0	1	1	1	1
CITd	AITd	S/B		A?	0	1	1	1	1
CITd	AITv	S/B		A?	0	1	1	1	1
CITv	AITd	S/B		A?	0	1	1	1	1
CITv	AITv	S/B		A?	0	1	1	1	1
FEF	CITd	S/B		A?	0	1	1	1	1
FEF	CITv	S/B		A?	0	1	1	1	1
46	CITd	S/B	M	A/D?	0	1	0	1	1
46	CITv	S/B	M	A/D?	0	1	0	1	1
PO	DP	S/B?		A?	0	1	1	1	1
MSTd	PITd	S?		A?	0	1	1	1	1
MSTd	PITv	S?		A?	0	1	1	1	1

Fortsetzung auf der nächsten Seite

Anhang A. Verwendete Verbindungsdaten

Von	Nach	laminares Muster Ursprung (S,B,I)	Muster Ziel (F,C,M)	Richtung in [FE91] (A,L,D)	D+	D	L	A	A+
					(I→M)	(B→M)	(B→C)	(B→F)	(S→F)
V1	V3A		F	A	0	0	0	2	2
V1	PIP		F	A	0	0	0	2	2
V1	V4t				1	1	1	1	1
V2	PIP			u	1	1	1	1	1
V2	VOT			u	1	1	1	1	1
V2	VIP			u	1	1	1	1	1
V3	VP				1	1	1	1	1
V3	V3A		F	A	0	0	0	1	1
V3	PIP		F	A	0	0	0	1	1
V3	LIP		F	A	0	0	0	1	1
V3	VIP		F	A	0	0	0	1	1
V3	TF		F	A	0	0	0	1	1
VP	V3A		F/C	A/L?	0	0	1	1	1
VP	PIP		F	A	0	0	0	1	1
VP	VOT		F	A	0	0	0	1	1
VP	LIP				1	1	1	1	1
VP	VIP		F	A	0	0	0	1	1
VP	TF		F	A	0	0	0	1	1
V3A	LIP				1	1	1	1	1
PIP	MT			u	1	1	1	1	1
PO	V4t				1	1	1	1	1
PO	MIP				1	1	1	1	1
PO	MDP				1	1	1	1	1
PO	VIP			u	1	1	1	1	1
V4	VOT				1	1	1	1	1
V4	LIP		F	A	0	0	0	1	1
V4	TF		F	A	0	0	0	1	1
V4	TH		F	A	0	0	0	1	1
V4t	MSTd				1	1	1	1	1
MT	LIP			u	1	1	1	1	1
MT	VIP		F	A	0	0	0	1	1
VOT	PITd		F	A	0	0	0	1	1
VOT	PITv		F	A	0	0	0	1	1
DP	FST			u	1	1	1	1	1
DP	LIP		C/F	L/A?	0	0	2	1	1
DP	FEF				1	1	1	1	1
DP	46		F	A	0	0	0	1	1
FST	PITd		M/C	D/L?	1	1	1	0	0
FST	PITv		M/C	D/L?	1	1	1	0	0
FST	VIP		M/C	D/L?	1	1	1	0	0
FST	STPp		F	A	0	0	0	0	1
FST	TF		C	L	0	0	1	1	1
VIP	LIP			u	1	1	1	1	1
LIP	PITv		C?	L?	0	1	1	1	0
LIP	TF		F/C?	A/L?	0	0	1	1	1
MSTd	TF				1	1	1	1	1
MSTd	STPp		F	A	0	0	0	1	1
MSTl	7a				1	1	1	1	1
MSTl	STPp		F	A	0	0	0	1	1

Fortsetzung auf der nächsten Seite

Von	Nach	laminares Muster Ursprung (S,B,I)	Ziel (F,C,M)	Richtung in [FE91] (A,L,D)	D+ (I→M)	D (B→M)	L (B→C)	A (B→F)	A+ (S→F)
PITv	CITd				1	1	1	1	1
PITv	AITd				1	1	1	1	1
PITv	TF				1	1	1	1	1
PITv	TH				1	1	1	1	1
CITd	STPp				1	1	1	1	1
CITv	STPp				1	1	1	1	1
CITd	TH				1	1	1	1	1
CITv	TH				1	1	1	1	1
CITv	TF				1	1	1	1	1
7a	AITd		F	A	0	0	0	1	1
7a	STPa		F/M?	A/D?	1	1	0	1	1
7a	STPp		F/M?	A/D?	1	1	0	1	1
7a	TF		F	A	0	0	0	1	1
7a	TH		F	A	0	0	0	1	1
STPp	TF			u	1	1	1	1	1
STPp	TH			u	1	1	1	1	1
STPa	TF			u	1	1	1	1	1
STPa	TH			u	1	1	1	1	1
AITd	STPa				1	1	1	1	1
AITd	TF				1	1	1	1	1
AITd	TH				1	1	1	1	1
AITv	TF		F	A	0	0	0	1	1
AITv	TH		F/C	A/L?	0	0	1	1	1
46	TF		M/C	D/L?	1	1	1	0	0
46	TH		C	L?	0	0	1	0	0
PIP	V1			u	1	1	1	1	1
PO	V1			u	1	1	1	1	1
MSTl	V1				1	1	1	1	1
V3	V2		M	D	1	1	0	0	0
V3A	V2		M	D	1	1	0	0	0
PIP	V2				1	1	1	1	1
VOT	V2			u	1	1	1	1	1
V4t	V2				1	1	1	1	1
PO	V2				1	1	1	1	1
MSTd	V2		M	D	1	1	0	0	0
MSTl	V2		M	D	1	1	0	0	0
FST	V2				1	1	1	1	1
VIP	V2				1	1	1	1	1
FEF	V2				1	1	1	1	1
PO	V3				1	1	1	1	1
FST	V3		M	D	1	1	0	0	0
FEF	V3				1	1	1	1	1
VOT	VP		M	D	1	1	0	0	0
PO	VP				1	1	1	1	1
FST	VP				1	1	1	1	1
LIP	VP		M	D	1	1	0	0	0
VIP	VP				1	1	0	0	0
FEF	VP				1	1	1	1	1
V4	V3A		M	D	1	1	0	0	0

Fortsetzung auf der nächsten Seite

Anhang A. Verwendete Verbindungsdaten

Von	Nach	laminares Ursprung (S,B,I)	Muster Ziel (F,C,M)	Richtung in [FE91] (A,L,D)	D+ (I→M)	D (B→M)	L (B→C)	A (B→F)	A+ (S→F)
PO	V3A				1	1	1	1	1
DP	V3A		M	D	1	1	0	0	0
MSTd	V3A		M	D	1	1	0	0	0
MSTl	V3A		M	D	1	1	0	0	0
FST	V3A		M	D	1	1	0	0	0
FEF	V3A				1	1	1	1	1
V4	PIP		M	D	1	1	0	0	0
MT	PIP			u	1	1	1	1	1
PO	PIP				1	1	1	1	1
DP	PIP		M	D	1	1	0	0	0
7a	PIP				1	1	1	1	1
DP	PO		C/M?	L/D?	1	1	2	0	0
MSTl	PO		M	D	1	1	0	0	0
FEF	PO			u	1	1	1	1	1
DP	V4		M	D	1	1	0	0	0
FST	V4		M	D	1	1	0	0	0
AITv	V4			u	1	1	1	1	1
FEF	V4				1	1	1	1	1
46	V4				1	1	1	1	1
MSTl	V4t				1	1	1	1	1
FEF	V4t				1	1	1	1	1
LIP	MT		M	D	1	1	0	0	0
FEF	MT		M?	D?	1	1	0	0	0
46	MT				1	1	1	1	1
7a	MIP				1	1	1	1	1
7a	MDP				1	1	1	1	1
PITd	VOT				1	1	1	1	1
PITv	VOT				1	1	1	1	1
FST	DP				1	1	1	1	1
MSTl	DP				1	1	1	1	1
FEF	DP			u	1	1	1	1	1
46	DP			u	1	1	1	1	1
7a	FST				1	1	1	1	1
MSTd	VIP		C	L	0	0	1	0	0
MSTl	VIP		C/M	L/D?	1	1	2	0	0
LIP	VIP			u	0	1	1	1	0
FEF	VIP		M	D	1	1	0	0	0
7a	VIP				1	1	1	1	1
MSTl	LIP				1	1	1	1	1
PITv	LIP				1	1	1	1	1
7a	LIP		M	D	1	1	0	0	0
FEF	LIP		M	D	1	1	0	0	0
TF	LIP				1	1	1	1	1
46	LIP				1	1	1	1	1
FEF	PITd				1	1	1	1	1
FEF	PITv				1	1	1	1	1
46	PITd				1	1	1	1	1
46	PITv				1	1	1	1	1
CITv	PITd				1	1	1	1	1

Fortsetzung auf der nächsten Seite

Von	Nach	laminares Muster		Richtung in [FE91] (A,L,D)	D+ (I→M)	D (B→M)	L (B→C)	A (B→F)	A+ (S→F)
		Ursprung (S,B,I)	Ziel (F,C,M)						
AITd	PITd				1	1	1	1	1
AITv	PITd				1	1	1	1	1
STPp	CITd		C	L	0	0	1	0	0
STPp	CITv		C	L	0	0	1	0	0
AITd	CITd				1	1	1	1	1
AITv	CITd				1	1	1	1	1
AITd	FEF		M(s)	D?	1	1	0	0	0
46	STPa		M	D	1	1	0	0	0
TF	AITd			u	1	1	1	1	1
TH	AITd			u	1	1	1	1	1
AITd	46		M(s)	D?	1	1	0	0	0

Anhang B

Berechnete Daten

V1	0	14	13	12	15	27	16	10	16	25	16	25	25	21	27	22	27	29	23	35	26	30	27	21	9	23	18	18	35	36	36			
V2	14	0	13	10	9	29	20	12	12	25	24	21	25	31	30	35	37	31	33	34	30	34	30	35	21	13	21	16	22	41	34	44		
V3	13	13	0	9	14	22	25	15	9	20	19	16	28	30	33	36	38	30	34	33	31	38	18	18	18	22	17	21	32	35	41			
Vp	12	10	9	0	15	27	18	16	18	23	20	21	23	25	28	31	33	25	31	28	28	33	25	17	21	18	18	33	34	36				
V3a	15	9	14	15	0	26	19	9	13	20	19	20	24	28	29	34	36	30	30	33	25	34	16	8	16	11	21	32	27	37	33			
V4	27	29	22	27	26	0	31	31	25	28	35	30	34	28	37	34	36	26	40	39	33	38	34	26	8	16	11	21	32	27	37	33		
VOT	16	20	25	18	19	31	0	16	26	25	22	27	13	19	14	19	19	15	27	18	22	19	25	13	27	20	24	31	32	32	32			
V4t	10	12	15	16	9	31	16	0	18	25	18	21	15	21	20	25	27	21	25	24	24	24	25	21	9	17	10	20	31	32	34			
MT	16	12	9	18	13	25	26	18	0	21	20	17	27	31	32	37	39	37	33	38	30	39	17	21	19	16	20	35	34	46	46			
MSTd	25	25	20	23	20	28	25	25	21	0	13	12	34	36	33	36	32	38	32	31	27	26	32	31	27	36	20	24	14	17	25	24	29	33
MSTl	16	24	19	20	19	35	22	18	20	13	0	17	25	29	24	29	29	31	27	26	32	31	27	36	20	19	16	22	23	26	30	30		
FST	25	21	16	21	20	30	27	21	17	12	17	0	30	32	29	32	34	32	28	31	27	34	18	26	16	17	25	28	31	33	33	33		
PITd	21	25	28	23	24	34	13	15	27	34	25	30	0	8	9	12	16	14	20	17	19	14	30	20	24	23	21	24	23	25	25	25		
PITv	27	31	30	25	28	28	19	21	31	36	29	32	8	0	13	10	14	10	22	21	21	18	34	26	28	27	25	24	23	21	21	24	20	
CITd	22	30	33	28	29	37	14	20	32	33	24	29	9	13	0	5	11	13	19	12	18	9	31	21	25	28	20	21	24	20	21	24	20	
CITv	27	35	36	31	34	34	19	25	37	36	29	32	12	10	5	0	10	10	22	15	19	12	18	9	31	21	25	28	20	21	24	20		
AITd	29	37	38	33	36	36	19	27	39	32	29	34	16	14	11	10	0	12	14	11	21	14	30	28	28	29	23	20	19	13	17	17		
AITv	23	31	30	25	30	26	15	21	37	38	31	32	14	10	13	10	12	0	22	15	23	16	34	22	30	29	27	28	27	17	17	17		
STPp	35	33	34	31	30	40	27	25	33	32	27	28	20	22	19	22	14	22	0	11	19	16	28	28	34	24	23	21	14	17	17	17		
STPa	26	34	33	28	33	39	18	24	38	31	26	31	17	21	12	15	11	11	0	16	9	29	25	29	26	22	13	22	14	14	14	14		
TF	30	30	31	28	25	33	22	24	30	27	32	27	19	21	18	19	21	23	19	16	0	9	33	25	23	24	24	23	24	24	24	24		
TH	27	35	38	33	34	38	19	25	39	36	31	34	14	18	9	12	14	16	16	9	9	0	34	26	30	31	23	22	23	17	17	17		
P0	21	21	18	25	16	34	25	21	17	20	19	18	30	34	31	36	30	34	28	29	33	34	0	18	20	13	17	26	27	35	35	35		
PIP	9	13	18	17	8	26	13	9	21	24	23	26	20	26	21	26	28	22	34	25	25	26	18	0	20	17	23	32	33	31	31	31		
LIP	23	21	22	21	16	34	27	17	19	14	19	16	24	28	25	30	28	30	24	29	23	30	20	20	0	15	15	22	25	31	31	31		
VIP	18	16	17	18	11	35	20	10	16	17	16	17	23	27	28	33	29	29	23	26	24	31	13	17	15	0	18	25	20	36	36	36		
DP	18	22	21	18	21	37	24	20	20	25	22	25	21	25	20	25	23	27	21	22	24	23	17	23	15	18	0	21	20	30	30	30		
ATa	35	41	32	33	32	42	31	31	35	24	23	28	24	24	21	22	20	28	14	13	23	22	26	32	22	25	21	0	19	15	15	15		
FEF	36	34	35	34	27	37	32	26	34	29	26	31	23	23	23	24	23	27	17	22	24	23	27	33	25	20	19	0	16	16	16	16		
A46	36	44	41	36	37	33	32	34	46	33	30	33	25	21	20	17	13	17	17	14	24	17	35	31	36	30	15	16	0	0	0	0		

Tabelle B.1: Hier ist für jedes Arealpaar im visuellen System (nach Young [You92]) die Hamming Distanz der Adressen angegeben. Die Verbindungen der beiden betrachteten Arealen untereinander sowie zu sich selbst blieben dabei unberücksichtigt.

	V1	V2	V3	V3a	V4	V4t	VOT	V4t	MT	MSTl	MSTl	FST	PITd	PITv	CITd	CITv	AITd	AITv	STPp	STPa	TF	TH	PO	PIP	LIP	VIP	DP	A7a	FEF	A46
V1	0	14	13	16	15	27	20	12	16	29	20	29	25	31	26	31	33	27	39	30	34	31	21	9	27	22	22	39	40	40
V2	14	0	13	10	9	29	20	14	12	25	24	23	29	35	34	39	41	35	37	38	34	39	23	15	25	18	26	45	38	48
V3	13	13	0	11	14	22	29	15	9	20	23	16	32	34	37	40	42	34	38	37	31	42	20	18	22	17	25	36	39	45
Vp	16	10	11	0	15	27	18	20	18	23	24	23	27	29	32	35	37	29	35	32	28	37	27	17	23	20	22	37	38	40
V3a	15	9	14	15	0	26	23	13	13	20	19	20	28	32	33	38	40	34	34	37	29	38	18	12	18	15	21	36	31	41
V4	27	29	22	27	26	0	33	31	25	32	39	30	34	28	37	34	40	26	44	43	33	38	26	34	39	37	46	41	35	36
VOT	20	20	29	18	23	33	0	20	30	29	26	31	15	21	18	23	23	19	31	22	26	23	29	17	31	24	28	35	36	36
V4t	12	14	15	20	13	31	20	0	18	27	20	21	19	25	24	29	31	25	29	28	28	29	23	13	21	14	24	35	30	38
MT	16	12	9	18	13	25	30	18	0	21	20	17	31	35	36	41	43	41	37	42	34	43	19	21	19	16	24	39	36	48
MSTd	29	25	20	23	20	32	29	27	21	0	17	12	36	38	37	40	36	42	32	35	29	40	20	28	14	17	25	24	29	37
MSTl	20	24	23	24	19	39	26	20	20	17	0	17	29	33	28	33	33	35	27	30	36	35	19	27	21	16	24	27	26	34
FST	29	23	16	23	20	30	31	21	17	12	17	0	32	34	33	36	38	36	28	35	27	38	22	30	16	17	27	30	31	37
PITd	25	29	32	27	28	34	15	19	31	36	29	32	0	12	13	14	18	16	24	21	23	18	34	24	28	27	25	28	25	27
PITv	31	35	34	29	32	28	21	25	35	38	33	34	12	0	15	10	16	10	26	25	23	20	38	30	30	31	29	28	25	23
CITd	26	34	37	32	33	37	18	24	36	37	28	33	13	15	0	9	13	15	21	16	22	11	35	25	29	32	24	25	24	20
CITv	31	39	40	35	38	34	23	29	41	40	33	36	14	10	9	0	10	10	24	19	21	14	40	30	34	37	29	26	23	17
AITd	33	41	42	37	40	40	23	31	43	36	33	38	18	16	13	10	0	16	18	13	23	16	34	32	32	33	27	20	19	13
AITv	27	35	34	29	34	26	19	25	41	42	35	36	16	10	15	10	16	0	26	19	23	16	38	26	34	33	31	32	31	21
STPp	39	37	38	35	34	44	31	29	37	32	27	28	24	26	21	24	18	26	0	11	19	16	32	38	28	27	25	14	17	17
STPa	30	38	37	32	37	43	22	28	42	35	30	35	21	25	16	19	13	19	11	0	16	9	33	29	33	30	26	13	26	14
TF	34	34	31	28	29	33	26	28	34	29	36	27	23	23	22	21	23	23	19	16	0	13	37	29	25	28	28	23	28	24
TH	31	39	42	37	38	38	23	29	43	40	35	38	18	20	11	14	16	16	16	9	13	0	38	30	34	35	27	22	27	17
P0	21	23	20	27	18	38	29	23	19	20	19	22	34	38	35	40	34	38	32	33	37	38	0	20	20	13	17	26	29	39
PIP	9	15	18	17	12	26	17	13	21	28	27	30	24	30	25	30	32	26	38	29	29	30	20	0	24	21	23	36	37	35
LIP	27	25	22	23	18	34	31	21	19	14	21	16	28	30	29	34	32	34	28	33	25	34	20	24	0	15	15	22	25	33
VIP	22	18	17	20	15	39	24	14	16	17	16	17	27	31	32	37	33	33	27	30	28	35	13	21	15	0	22	27	24	40
DP	22	26	25	22	21	37	28	24	24	25	24	27	25	29	24	29	27	31	25	26	28	27	17	23	15	22	0	21	22	30
A7a	39	45	36	37	36	46	35	35	39	24	27	30	28	28	25	26	20	32	14	13	23	22	26	36	22	27	21	0	19	15
FEF	40	38	39	38	31	41	36	30	36	29	26	31	25	25	24	23	19	31	17	26	28	27	29	37	25	24	22	19	0	16
A46	40	48	45	40	41	35	36	38	48	37	34	37	27	23	20	17	13	21	17	14	24	17	39	35	33	40	30	15	16	0

Tabelle B.2: Hier ist für jedes Arealpaar im visuellen System (nach Young [You92]) die Hamming Distanz der Adjazenzen angegeben. Die Verbindung eines Areals mit sich selbst wurden dabei als Vorhanden betrachtet, also in der Adjazenzmatrix die Diagonalewerte auf 1 gesetzt.

	V1	V2	V3	Vp	V3a	V4	VOt	V4t	MT	MSTd	MSTl	FST	PtD	PtV	CtD	CtV	AItD	AItV	STPa	TF	TH	PO	PIP	LIP	VIP	DP	A7a	FEF	A46
V1	-	72	86	-	72	129	-	103	-	-	-	-	-	-	-	-	-	-	-	-	-	64	45	-	-	-	-	-	-
V2	-	-	5	5	3	13	1	2	8	12	8	16	-	-	-	-	-	-	-	-	-	5	0	6	-	-	-	-	-
V3	-	3	-	-	3	13	-	2	8	13	-	17	-	-	-	-	-	-	-	-	27	5	0	14	6	-	-	-	-
Vp	-	-	-	-	-	-	0	-	-	5	-	5	-	-	-	-	-	-	-	-	15	-	-	0	-	-	-	-	-
V3a	-	3	5	5	-	13	-	8	8	12	8	16	-	-	-	-	-	-	-	-	-	5	-	-	9	-	-	-	-
V4	-	3	5	5	3	-	-	2	8	-	17	11	6	10	10	-	7	-	-	-	27	26	0	14	-	9	-	-	31
VOt	-	-	-	0	-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
V4t	-	-	-	-	-	-	-	-	4	4	-	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MT	-	3	5	5	3	13	-	2	14	10	18	-	-	-	-	-	-	-	-	-	-	0	14	6	-	-	-	24	34
MSTd	-	-	-	0	-	-	-	0	-	-	-	5	0	-	-	-	-	-	-	-	-	-	0	0	0	15	10	-	-
MSTl	-	-	-	-	-	-	-	0	-	-	-	4	-	-	-	-	-	-	-	-	-	-	0	0	-	-	-	8	-
FST	-	-	-	-	-	-	-	0	5	5	-	5	0	-	-	-	-	-	-	-	15	-	-	0	0	-	15	10	-
PtD	-	-	-	-	-	-	-	-	-	-	-	-	-	1	16	0	-	-	-	-	-	-	-	-	-	-	-	2	3
PtV	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	-	-	-	2	3
CtD	-	-	-	-	-	-	-	-	-	-	-	-	0	-	9	0	-	-	-	-	-	-	-	-	-	-	-	2	3
CtV	-	-	-	-	-	-	-	-	-	-	-	-	0	-	14	0	-	-	-	-	-	-	-	-	-	-	-	2	3
AItD	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
AItV	-	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-	-	-	-	-	3	3	-	-	-	-	-	-	-
STPa	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
STPp	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
STPa	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TF	-	-	-	0	-	-	-	-	2	2	-	2	-	0	28	0	30	32	-	-	-	-	-	-	-	-	6	-	6
TH	-	-	-	-	-	-	-	-	-	-	-	-	0	1	9	0	10	11	-	-	-	-	-	-	-	3	-	3	-
PO	-	-	-	-	-	-	-	8	12	8	-	-	-	-	-	-	-	-	-	-	-	-	14	6	9	27	-	-	-
PIP	-	5	5	0	-	13	-	8	-	-	-	-	-	-	-	-	-	-	-	-	-	5	-	9	-	9	-	-	-
LIP	-	-	-	-	-	-	-	-	4	4	4	4	0	-	-	-	-	-	-	-	12	-	-	0	0	12	8	12	-
VIP	-	-	-	-	-	-	-	-	4	4	4	4	-	-	-	-	-	-	-	-	-	-	0	-	-	-	12	-	-
DP	-	-	-	-	-	-	-	-	4	4	4	4	-	-	-	-	-	-	-	-	-	-	0	-	-	-	12	-	-
A7a	-	-	-	-	-	-	-	-	1	1	-	-	-	-	32	-	33	34	3	-	-	-	-	0	-	0	-	2	3
FEF	-	-	-	-	-	-	-	-	1	1	1	1	-	-	31	-	32	-	-	-	-	-	-	0	-	0	3	-	3
A46	-	-	-	-	-	-	-	-	-	-	-	-	-	-	28	-	30	32	6	-	-	-	-	-	0	6	4	-	-

Tabelle B.4: Hier ist für jede Kante im visuellen System (nach Young [You92]) ausgehend vom Startknoten V1 die Anzahl der Wege, die um 1 länger als kürzeste Wege sind, angegeben die über diese Kante laufen. Diese Wege laufen nur über Kanten, die Knoten in gleicher oder ansteigender Entfernung von V1 verbinden. Andere Kanten wurden hier nicht berücksichtigt.

	V1	V2	V3	Vp	V3a	V4	VOT	V4t	MT	MSTd	MSTl	FST	PITd	PITv	CITd	CITv	AITd	AITv	STPp	STPa	TF	TH	PO	PIP	LIP	VIP	DP	A7a	FEF	A46				
V1	-	75	91	-	75	142	-	-	111	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
V2	-	-	5	5	3	13	1	2	8	13	9	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
V3	-	3	-	-	3	13	-	2	8	14	-	18	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
Vp	-	-	-	-	-	0	-	-	5	-	-	5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
V3a	-	3	5	5	-	13	-	-	8	13	9	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
V4	-	3	5	5	3	-	-	2	8	-	-	18	12	6	11	11	-	7	-	-	-	-	30	29	-	0	14	-	9	-	34			
VOT	-	-	-	0	-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-			
V4t	-	-	-	-	-	-	-	-	-	4	4	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-			
MT	-	3	5	5	3	13	-	2	-	15	11	19	-	-	-	-	-	-	-	-	-	-	0	14	6	-	-	-	-	-	26	37		
MSTd	-	-	0	-	-	-	-	0	-	-	5	5	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	15	10	-	-			
MSTl	-	-	-	-	-	-	-	-	-	-	4	4	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	8	-	-	-			
FST	-	-	-	-	-	-	-	0	-	5	5	5	0	-	-	-	-	-	-	-	-	15	-	-	0	0	-	15	10	-	-			
PITd	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	17	0	-	-	-	-	-	-	-	3		
PITv	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	3		
CITd	-	-	-	-	-	-	-	-	-	-	-	-	-	0	-	-	0	0	-	-	-	8	0	-	-	-	-	-	-	-	-	2	3	
CITv	-	-	-	-	-	-	-	-	-	-	-	-	-	0	-	-	15	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	
AITd	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	
AITv	-	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-	1	-	-	-	-	3	3	-	-	-	-	-	-	-	-	-	-	
STPp	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	-	-	-	-	-	-	-	-	-	-	-	
STPa	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	6	-	-	-	6	21	-	-	-	-	-	-	-	-	-	-	-	
TF	-	-	-	0	-	-	-	-	2	-	-	2	0	0	2	30	0	32	34	-	-	-	-	-	-	-	-	-	-	-	-	-	6	
TH	-	-	-	-	-	-	-	-	-	-	-	-	0	0	1	1	10	0	11	12	-	-	-	-	-	-	-	-	-	-	-	-	3	
PO	-	-	5	5	-	13	-	-	8	13	9	-	-	-	-	-	-	-	-	-	-	-	-	5	-	-	-	-	-	-	-	-	3	
PIP	-	-	-	0	-	-	-	-	4	4	4	4	-	0	-	-	-	-	-	-	-	12	-	-	-	0	0	12	8	-	-	-	12	
LIP	-	-	-	-	-	-	-	-	4	4	4	4	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-	-	-	-	-	-	12	
VIP	-	-	-	-	-	-	-	-	4	4	4	4	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-	-	-	-	-	-	12	
DP	-	-	-	-	-	-	-	-	4	4	4	4	-	-	-	-	-	-	-	-	-	-	-	0	-	-	-	-	-	-	-	-	12	
A7a	-	-	-	-	-	-	-	-	1	1	1	1	-	-	-	-	-	-	-	-	-	33	-	34	35	3	3	-	-	-	-	-	2	3
FEF	-	-	-	-	-	-	-	-	1	1	1	1	-	-	-	-	-	-	-	-	-	32	-	33	-	0	0	0	3	3	-	-	3	3
A46	-	-	-	-	-	-	-	-	-	-	-	-	-	2	2	2	30	-	32	34	6	6	6	-	-	-	0	6	4	-	-	-	4	-

Tabelle B.5: Hier ist für jede Kante im visuellen System (nach Young [You92]) ausgehend vom Startknoten V1 die Anzahl der kürzesten Wege und der Wege, die um 1 länger als kürzeste Wege sind, angegeben die über diese Kante laufen. Diese Wege laufen nur über Kanten, die Knoten in gleicher oder ansteigender Entfernung von V1 verbinden. Andere Kanten wurden hier nicht berücksichtigt.

Tabelle B.6: Die in Kapitel 7 verwendeten Arealbeziehungen. Die in CoCoMac verzeichneten Beziehungen werden angegeben mit D für „Disjoint“, I für „Identiy“, O für „Overlap“ und S für „Subarea“. r_1 (r_2) bezeichnet die relative Größe des Schnittes in Areal 1 (Areal 2) und r'_1 (r'_2) die relative Größe des gewichteten Schnittes in Areal 1 (Areal 2). Die Daten sind so geordnet, daß $r_1 \geq r_2$ für die Beziehungen D, I und O aus CoCoMac gilt. Die S-Beziehungen, für die eine solche Anordnung nicht möglich war, sind mit einem Plus (+) markiert. In der Regel ist auch $r'_1 \geq r'_2$. Die Beziehungen bei denen dieses nicht der Fall ist, wurden mit einem Stern (*) markiert.

Areal 1	Areal 2	Beziehung in CoCoMac	r_1	r_2	r'_1	r'_2	
LV00-V2v	BB47-TE	D	0,001	0,001	0	0	
PHT00-MT(V5)	LV00-V4	D	0,001	0,001	0	0	
PHT00-V3V	LV00-V4	D	0,003	0,002	0,001	0,001	
PHT00-3a	LV00-2	D	0,003	0,002	0,002	0,001	
PHT00-3b	LV00-2	D	0,003	0,002	0,001	0,001	
LV00-V2d	BB47-OC	D	0,004	0,001	0,001	0	
UD86-TF	BB47-TE	D	0,005	0,001	0,002	0	
LV00-MDP	PHT00-MT(V5)	D	0,006	0,001	0,003	0	
LV00-V2v	FV91-V1	D	0,006	0,003	0,001	0,001	
LV00-V2v	B05-17	D	0,006	0,003	0,001	0	
UD86-MT	FV91-V4	D	0,007	0,001	0,003	0	
LV00-V3A	PHT00-MT(V5)	D	0,007	0,001	0,005	0	
FV91-V3A	PHT00-V2	D	0,008	0,001	0,012	0	
LV00-V4	UD86-V2	D	0,008	0,003	0,002	0,001	
LV00-LIPd	FV91-VIP	D	0,008	0,005	0,005	0,002	
LV00-V2d	PHT00-V1	D	0,009	0,002	0,004	0	
FV91-3b	LV00-3a	D	0,01	0,008	0,008	0,003	
LV00-V2d	FV91-V1	D	0,011	0,003	0,003	0,001	
LV00-V2v	UD86-V1	D	0,014	0,005	0,003	0,001	
FV91-V3A	LV00-V4	D	0,016	0,004	0,007	0,001	
LV00-TF	FV91-V4	D	0,016	0,005	0,007	0,002	
LV00-V3	FV91-V2	D	0,017	0,002	0,007	0	
LV00-1	FV91-2	D	0,017	0,01	0,011	0,005	
B05-17	FV91-V2	D	0,018	0,015	0,003	0,005	*
PHT00-MT(V5)	LV00-V1	D	0,023	0,003	0,008	0,003	
LV00-V2d	B05-17	D	0,024	0,008	0,009	0,001	
LV00-V4	PHT00-V2	D	0,025	0,008	0,01	0,003	
LV00-MT	FV91-V4	D	0,027	0,006	0,015	0,002	
B05-18	UD86-V1	D	0,028	0,024	0,007	0,005	
LV00-1	PHT00-4(F1)	D	0,029	0,011	0,019	0,003	
PHT00-V3V	FV91-V4	D	0,029	0,013	0,024	0,004	
UD86-V4	FV91-V2	D	0,034	0,013	0,011	0,005	
LV00-1	FV91-4	D	0,038	0,007	0,024	0,002	
LV00-1	PHT00-3b	D	0,038	0,025	0,035	0,013	
UD86-TF	BB47-OC	D	0,039	0,004	0,024	0,001	
UD86-V4	PHT00-V2	D	0,039	0,021	0,016	0,008	
LV00-V3	UD86-V4	D	0,04	0,01	0,02	0,003	
LV00-V3A	UD86-V4	D	0,041	0,004	0,02	0,001	

Fortsetzung auf der nächsten Seite

Anhang B. Berechnete Daten

Areal 1	Areal 2	Beziehung in CoCoMac	r_1	r_2	r'_1	r'_2	
FV91-V3A	UD86-V4	D	0,041	0,006	0,019	0,002	
BB47-OB	UD86-V1	D	0,042	0,029	0,017	0,007	
LV00-V2v	BB47-OC	D	0,045	0,022	0,012	0,004	
LV00-1	B05-2	D	0,046	0,01	0,029	0,004	
UD86-V2	B05-17	D	0,046	0,037	0,018	0,008	
UD86-TF	BB47-OB	D	0,052	0,006	0,044	0,002	
BB47-OC	FV91-V2	D	0,053	0,047	0,022	0,024	*
FV91-7b	LV00-7a	D	0,053	0,049	0,026	0,022	
LV00-1	PHT00-3a	D	0,059	0,044	0,045	0,032	
UD86-V4	LV00-V2v	D	0,064	0,054	0,038	0,024	
BB47-OC	B05-18	D	0,064	0,059	0,015	0,018	*
BB47-OB	B05-17	D	0,064	0,06	0,037	0,02	
B05-18	LV00-V1	D	0,069	0,056	0,033	0,023	
FV91-V1	B05-18	D	0,069	0,061	0,029	0,026	
UD86-V2	BB47-OC	D	0,072	0,054	0,034	0,014	
LV00-V2v	PHT00-V1	D	0,073	0,03	0,032	0,012	
UD86-TF	PHT00-V2	D	0,079	0,01	0,074	0,004	
BB47-OB	PHT00-V1	D	0,08	0,058	0,044	0,023	
UD86-TEO	LV00-V4	D	0,082	0,062	0,043	0,027	
UD86-V2	FV91-V1	D	0,085	0,066	0,035	0,019	
B05-18	PHT00-V1	D	0,086	0,079	0,055	0,042	
UD86-V2	B05-19	D	0,088	0,083	0,061	0,049	
PHT00-3a	LV00-4	D	0,089	0,043	0,076	0,016	
LV00-TF	B05-19	D	0,091	0,013	0,052	0,005	
BB47-OB	FV91-V1	D	0,091	0,082	0,044	0,031	
PHT00-MT(V5)	UD86-V4	D	0,092	0,04	0,058	0,027	
UD86-V2	LV00-V1	D	0,097	0,054	0,045	0,015	
BB47-OB	LV00-V1	D	0,098	0,064	0,051	0,023	
LV00-3b	PHT00-4(F1)	D	0,101	0,051	0,057	0,028	
UD86-V2	PHT00-V1	D	0,102	0,064	0,069	0,027	
FV91-4	B05-3	D	0,104	0,059	0,06	0,065	*
UD86-TEO	FV91-V4	D	0,108	0,064	0,079	0,036	
LV00-V4	B05-19	D	0,122	0,041	0,102	0,016	
FV91-V3A	UD86-V2	D	0,13	0,011	0,082	0,005	
LV00-V3	PHT00-V2	D	0,139	0,018	0,088	0,008	
PHT00-V2	UD86-V1	D	0,152	0,093	0,115	0,042	
PHT00-MT(V5)	FV91-V4	D	0,156	0,085	0,118	0,055	
LV00-V3	BB47-OB	D	0,161	0,019	0,099	0,007	
PHT00-V2	BB47-OC	D	0,172	0,133	0,131	0,05	
LV00-4	B05-3	D	0,181	0,049	0,105	0,053	
PHT00-V2	FV91-V1	D	0,182	0,148	0,143	0,075	
PHT00-V2	B05-17	D	0,183	0,153	0,153	0,063	
PHT00-V2	LV00-V1	D	0,19	0,11	0,151	0,051	
B05-19	BB47-OB	D	0,194	0,175	0,154	0,182	*
B05-19	FV91-V2	D	0,219	0,154	0,212	0,251	*
LV00-V3	FV91-V4	D	0,223	0,067	0,233	0,048	
FV91-V4	B05-19	D	0,231	0,098	0,145	0,076	
PHT00-4(F1)	B05-3	D	0,279	0,075	0,323	0,089	
FV91-TH	LV00-TF	D	0,292	0,091	0,242	0,09	
LV00-TF	PHT00-V4V	D	0,325	0,229	0,342	0,189	

Fortsetzung auf der nächsten Seite

Areal 1	Areal 2	Beziehung in CoCoMac	r_1	r_2	r'_1	r'_2	
FV91-V2	LV00-V1	D	0,327	0,271	0,1	0,042	
FV91-V2	UD86-V1	D	0,33	0,288	0,101	0,047	
FV91-V2	PHT00-V1	D	0,354	0,333	0,128	0,067	
PHT00-3b	LV00-3a	D	0,362	0,356	0,349	0,375	*
PHT00-1	LV00-2	D	0,405	0,234	0,508	0,247	
LV00-V3	UD86-V2	D	0,464	0,063	0,53	0,053	
LV00-LIPv	FV91-VIP	D	0,722	0,425	0,776	0,477	
FV91-7a	LV00-7a	D	0,817	0,737	0,875	0,866	
PG91-6Vb	LV00-6Vb	I	0,036	0,008	0,025	0,003	
PG91-8Ac	LV00-8Ac	I	0,069	0,036	0,049	0,019	
BB47-OA	B05-19	I	0,341	0,296	0,21	0,293	*
UD86-FST	PHT00-FST	I	0,38	0,183	0,347	0,146	
UD86-MT	LV00-MT	I	0,429	0,405	0,469	0,387	
LV00-1	FV91-1	I	0,471	0,463	0,521	0,527	*
PG91-6Ds	LV00-6Ds	I	0,518	0,324	0,546	0,387	
FV91-TH	BB47-TH	I	0,526	0,343	0,516	0,308	
FV91-V3A	UD86-V3A	I	0,528	0,48	0,616	0,546	
BB47-FB	B05-6	I	0,532	0,415	0,553	0,43	
BB47-TF	LV00-TF	I	0,556	0,359	0,576	0,355	
PHT00-3a	LV00-3a	I	0,559	0,482	0,644	0,544	
PHT00-MST	UD86-MST	I	0,569	0,335	0,658	0,333	
LV00-3a	FV91-3a	I	0,578	0,495	0,632	0,522	
PG91-46v	LV00-46v	I	0,581	0,281	0,598	0,329	
FV91-TF	LV00-TF	I	0,627	0,582	0,652	0,721	*
LV00-31	PHT00-31	I	0,635	0,431	0,711	0,527	
PHT00-V2	B05-18	I	0,66	0,472	0,711	0,396	
BB47-OB	B05-18	I	0,703	0,563	0,744	0,505	
UD86-V3v	PHT00-V3V	I	0,71	0,386	0,791	0,462	
B05-4	FV91-4	I	0,717	0,706	0,814	0,81	
B05-4	BB47-FA	I	0,731	0,312	0,779	0,611	
BB47-OB	FV91-V2	I	0,738	0,577	0,842	0,844	*
LV00-PIP	FV91-PIP	I	0,74	0,479	0,778	0,563	
PHT00-V2	FV91-V2	I	0,743	0,519	0,811	0,754	
PHT00-4(F1)	LV00-4	I	0,751	0,748	0,834	0,855	*
BB47-OC	PHT00-V1	I	0,759	0,641	0,867	0,921	*
PHT00-V2	BB47-OB	I	0,766	0,685	0,837	0,758	
UD86-V2	PHT00-V2	I	0,768	0,735	0,813	0,826	*
UD86-V3d	PHT00-V3D	I	0,77	0,25	0,766	0,275	
UD86-V2	FV91-V2	I	0,78	0,522	0,851	0,764	
BB47-TF	FV91-TF	I	0,785	0,547	0,859	0,655	
B05-17	PHT00-V1	I	0,793	0,62	0,911	0,901	
LV00-MT	PHT00-MT(V5)	I	0,794	0,316	0,79	0,505	
UD86-VIP	FV91-VIP	I	0,807	0,215	0,831	0,227	
FV91-V1	PHT00-V1	I	0,808	0,653	0,903	0,911	*
LV00-4	B05-4	I	0,813	0,392	0,89	0,483	
LV00-V3A	FV91-V3A	I	0,814	0,48	0,905	0,612	
FV91-V4	UD86-V4	I	0,814	0,656	0,897	0,756	
FV91-7a	LV00-7a	I	0,817	0,737	0,875	0,866	
FV91-MT	LV00-MT	I	0,848	0,478	0,897	0,523	
LV00-V4	FV91-V4	I	0,87	0,687	0,925	0,807	

Fortsetzung auf der nächsten Seite

Anhang B. Berechnete Daten

Areal 1	Areal 2	Beziehung in CoCoMac	r_1	r_2	r'_1	r'_2	
BB47-OC	UD86-V1	I	0,88	0,693	0,946	0,948	*
BB47-OC	LV00-V1	I	0,897	0,67	0,952	0,935	
FV91-V1	BB47-OC	I	0,915	0,874	0,968	0,929	
LV00-V4	UD86-V4	I	0,919	0,585	0,964	0,701	
FV91-V1	UD86-V1	I	0,926	0,697	0,978	0,932	
PHT00-V1	UD86-V1	I	0,934	0,871	0,972	0,937	
B05-17	BB47-OC	I	0,94	0,871	0,979	0,949	
B05-17	UD86-V1	I	0,949	0,692	0,989	0,948	
B05-17	FV91-V1	I	0,953	0,924	0,976	0,967	
LV00-4	FV91-4	I	0,957	0,455	0,982	0,565	
PHT00-V1	LV00-V1	I	0,967	0,856	0,988	0,928	
B05-17	LV00-V1	I	0,975	0,675	0,995	0,933	
FV91-V1	LV00-V1	I	0,99	0,707	0,998	0,936	
UD86-V1	LV00-V1	I	0,99	0,94	0,998	0,983	
B05-27	BB47-TF	O	0,135	0,081	0,088	0,052	
BB47-TG	B05-20	O	0,215	0,18	0,169	0,147	
BB47-TG	B05-22	O	0,306	0,109	0,237	0,083	
BB47-TF	B05-20	O	0,327	0,094	0,281	0,079	
BB47-TG	B05-21	O	0,472	0,332	0,591	0,314	
B05-18	FV91-V2	O	0,503	0,49	0,41	0,635	*
FV91-MSTd	BB47-PG	S	0,013	0,003	0,004	0,001	
PHT00-TFL	BB47-TF	S	0,047	0,007	0,036	0,003	
FV91-FST	BB47-TA	S	0,061	0,005	0,026	0,001	
BB47-IB	B05-13	S	0,067	0,089	0,034	0,052	+
B05-24	BB47-LA	S	0,072	0,33	0,027	0,26	+
FV91-MSTd	B05-7	S	0,104	0,027	0,076	0,014	
BB47-IA	B05-16	S	0,116	0,233	0,12	0,203	+
B05-24	BB47-FL	S	0,124	0,166	0,073	0,123	+
PHT00-THO	BB47-TH	S	0,158	0,071	0,116	0,044	
BB47-TH	B05-20	S	0,171	0,036	0,164	0,017	
BB47-IA	B05-15	S	0,185	0,767	0,257	0,797	+
FV91-V3	BB47-OA	S	0,272	0,029	0,218	0,013	
FV91-7a	B05-7	S	0,369	0,116	0,388	0,13	
FV91-LIP	B05-7	S	0,417	0,035	0,41	0,032	
PHT00-TFM	BB47-TF	S	0,48	0,042	0,462	0,018	
FV91-MIP	B05-7	S	0,482	0,033	0,451	0,032	
FV91-STPa	BB47-TA	S	0,537	0,054	0,48	0,049	
PHT00-TFO	BB47-TF	S	0,615	0,169	0,682	0,154	
B05-21	BB47-TE	S	0,619	0,321	0,664	0,442	
B05-20	BB47-TE	S	0,621	0,271	0,69	0,257	
FV91-46	B05-9	S	0,64	0,171	0,714	0,199	
FV91-V4	BB47-OA	S	0,658	0,322	0,776	0,417	
BB47-TA	B05-22	S	0,658	0,616	0,726	0,752	*
FV91-VIP	B05-7	S	0,687	0,119	0,715	0,146	
FV91-LIP	BB47-PG	S	0,689	0,062	0,68	0,072	
FV91-7a	BB47-PG	S	0,714	0,24	0,79	0,281	
FV91-MDP	B05-7	S	0,724	0,078	0,803	0,064	
FV91-VP	BB47-OA	S	0,774	0,08	0,772	0,055	
B05-14	BB47-IB	S	0,836	0,685	0,91	0,815	
FV91-STPp	BB47-TA	S	0,998	0,257	0,999	0,307	

Fortsetzung auf der nächsten Seite

Areal 1	Areal 2	Beziehung in CoCoMac	r_1	r_2	r'_1	r'_2
UD86-VIP	BB47-PG	S	1	0,049	1	0,056
FV91-V3A	BB47-OA	S	1	0,09	1	0,128

