
Bioinformatische Anwendung von *Graphlets* zur Analyse von Proteinstrukturtopologien

Ben Haladik

25. April 2016

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	<i>State of the art</i>	2
1.3	Ziele	3
1.4	Aufbau der Arbeit	3
2	Materialien und Methoden	4
2.1	<i>Protein Topology Graph Library</i>	4
2.2	PLCC	4
2.3	Der <i>Graphlet</i> -Algorithmus	5
2.3.1	Beschreibung des Algorithmus	5
2.4	graphletAnalyser	6
2.5	Ähnlichkeitsmaß	7
2.5.1	Relative <i>Graphlet</i> -Häufigkeiten-Distanz	7
2.6	<i>PDBeFold</i>	8
2.7	Datensätze	8
2.7.1	Fallstudie	8
2.7.2	Der PDBTop500-Datensatz	9
3	Ergebnisse	15
3.1	Der modifizierte Jaccard-Index	15
3.2	Der <i>Graphlet</i> -Worte-Algorithmus	15
3.3	Erweiterung von graphletAnalyser	17
3.4	Fallstudie	18
3.5	PDBTop500-Datensatz	19
4	Diskussion und Ausblick	23
4.1	Diskussion	23
4.1.1	Fallstudie	23
4.1.2	PDBTop500-Datensatz und Aldolasen	27
4.2	Ausblick	27
4.2.1	Optimierung der Laufzeit von graphletAnalyser	27
4.2.2	Neue Bewertungsschemata	28
4.2.3	Weiterentwicklung des <i>Graphlet</i> -Worte-Algorithmus	28
4.2.4	Zusammenhängende Graphen	28

4.2.5	Schluss	28
5	Anhang	30

Abbildungsverzeichnis

3.1	<i>Graphlet</i> -Verteilungen für Aminosäuregraphen	20
3.2	<i>Graphlet</i> -Verteilungen für Proteingraphen	21
3.3	<i>Graphlet</i> -Verteilungen für Komplexgraphen	22
4.1	Proteingraph von 3GF9 <i>Schäfer et al.</i>	24
4.2	Proteingraph von 2id9 <i>Schäfer et al.</i>	25
4.3	Proteingraph von 3i42 <i>Schäfer et al.</i>	26
4.4	Proteingraph von 2w0i <i>Schäfer et al.</i>	26
5.1	Graphlets der Größe 3 (<i>Shervashidze et al.</i>)	30
5.2	Graphlets der Größe 4 (<i>Shervashidze et al.</i>)	30
5.3	Graphlets der Größe 5 (<i>Shervashidze et al.</i>)	31

Tabellenverzeichnis

2.1	α -Proteine der Fallstudie	10
2.2	β -Proteine der Fallstudie	11
2.3	α/β -Proteine der Fallstudie	12
2.4	Aldolasen	13
2.5	Aldolasen - Lyasen	14
4.1	Korrelationen der RGF mit den Jaccard-Indizes für die verschiedenen Graphtypen .	24
4.2	Alignmenttabelle aus <i>PDBeFold</i>	25
5.2	RGF-Distanzen der Proteingraphen	32
5.1	RGF-Distanzen der Aminosäuregraphen	32
5.3	RGF-Distanzen der Komplexgraphen	33
5.4	Jaccard-Indizes der Aminosäuregraphen	33
5.5	Jaccard-Indizes der Proteingraphen	34
5.6	Jaccard-Indizes der Komplexgraphen	34

Abkürzungsverzeichnis

GEF *Guanine Nucleotide Exchange Factor*

GML *Graph Modelling Language / Graph Meta Language*

GTP Guanosintriphosphat

PDB *Protein Data Bank*

PTGL *Protein Topology Graph Library*

RGF Relative *Graphlet* Häufigkeiten Distanz

SSE Sekundärstrukturelement

Kapitel 1

Einleitung

1.1 Motivation

Proteine sind die essentiellen Bausteine zellulären Lebens. Jede Aktivität eines Organismus kann durch die Aktivität der beteiligten Proteine beschrieben werden. Dementsprechend ist ein tiefes Verständnis von Proteinen zentral für das Verständnis von zellulärem Leben. Die Struktur eines Proteins bestimmt seine Funktion und die vergleichende Analyse dieser Strukturen liefert Einblicke in entfernte evolutionäre Verwandtschaftsbeziehungen, die durch reine Sequenzanalysen nicht mehr nachvollziehbar sind.

Hierzu betrachten wir Proteinstrukturtopologien. Die Topologie eines Proteins ist als die Anordnung seiner Sekundärstrukturelemente (SSEs) zueinander definiert. Die Betrachtung von SSEs hat den Vorteil, dass diese auch über große evolutionäre Distanzen stark konserviert sind. Es ist also naheliegend, Proteine entsprechend ihrer Topologien einzuordnen.

Da die Anzahl der bekannten Proteinstrukturen in den letzten Jahren rasant angestiegen sind, ist es nicht mehr möglich diese Einordnung und die hierfür nötigen Vergleiche nur manuell vorzunehmen. Es werden algorithmische Vergleichsmethoden und entsprechende Datenstrukturen benötigt.

Zur Darstellung dieser Topologien werden hier Graphen verwendet. Graphen gehören zu den am stärksten untersuchten mathematischen Strukturen, da sich mit ihnen viele verschiedene komplexe Zusammenhänge darstellen lassen. Ihre Nutzung ist überall dort angebracht, wo Daten nicht als Zahlen oder Vektoren darstellbar sind, weil sie eine Menge von Objekten und ihre Beziehungen untereinander repräsentieren.

Graphen finden sich in der Erforschung von sozialen Netzwerken wieder und werden in der Chemie zur Darstellung von Molekülen genutzt. Auch in der Bioinformatik sind Graphen das zentrale Mittel zur Darstellung komplexer Beziehungen. Interaktionen von Proteinen werden genauso als Graphen modelliert, wie Signalwege in Zellen oder eben Proteine. Die Analyse von biologischen Netzwerken ist ein zentrales Mittel, um biologische Vorgänge besser zu verstehen [21].

Der Vergleich von Graphen ist jedoch keine triviale Aufgabe. Die Frage: „Ist dieser Graph in diesem anderen Graphen enthalten?“ befriedigend schnell beantworten zu können, ist nicht möglich, denn das zugrunde liegende Entscheidungsproblem ist NP-vollständig [22].

Um dieses Problem zu umgehen, werden approximative Methoden angewandt. Hierzu gehört das Berechnen von *Graphlets* [28]. *Graphlets* sind kleine, induzierte Teilgraphen, die für große Graphen immer noch schnell abzählbar sind. Mit dieser Technik lassen sich in polynomieller Zeit Teilstruktu-

ren eines Graphen ermitteln. Dabei erhält man Vektoren, deren Vergleich einfacher durchzuführen ist, als ein direkter Vergleich der Graphen selbst. So lassen sich *Graphlets* zur vergleichenden Analyse von Proteinstrukturtopologien verwenden.

1.2 *State of the art*

Mit der wachsenden Anzahl von Struktureinträgen in der *Protein Data Bank* (PDB) ist eine Vielzahl von Methoden entstanden, um diese Strukturen zu vergleichen.

Der wohl bekannteste Algorithmus zum Vergleich von dreidimensionalen Proteinstrukturen ist DALI von *Holm* und *Sander* [20]. Er führt ein globales Alignment durch, indem Distanzmatrizen verglichen werden. In diesen Matrizen sind die intramolekularen Distanzen der C_α -Atome der jeweiligen Proteine eingetragen.

In den 23 Jahren, die seit der Veröffentlichung von DALI vergangen sind, sind aber noch viele weitere Methoden mit unterschiedlichen Ansätzen entwickelt worden. Der Algorithmus von *Shindyalov* und *Bourne* [34] berechnet ein Alignment von Proteinstrukturen, indem er zunächst kleine Paare von Substrukturen aligniert und dann versucht, dieses Alignment auf einen optimalen Pfad auszuweiten.

Der FATCAT-Algorithmus von *Ye* und *Godzik* [39] verwendet eine ähnliche Idee und versucht zusätzlich, die Substrukturen flexibel zu alignieren, um auch gleiche Proteine mit veränderter Konformation erkennen zu können.

TM-align von *Zhang* und *Skolnick* [40] berechnet eine Rotationsmatrix und nutzt Dynamische Programmierung. Der Algorithmus benutzt als Bewertungsschema den sogenannten *TM-Score*, der besonders gut geeignet ist, um lokale Ähnlichkeiten zu erkennen.

Der SSM-Algorithmus von *Krissinel* und *Hendrick* [23] verwendet eine graphenbasierte Darstellung von Proteinen für ein erstes Alignment und verfeinert dieses dann durch die Berechnung der Distanzen äquivalenter α -C-Atome. Er wird im *PDBeFold-Web-Server* implementiert. Als einziger hier beschriebener Algorithmus ermöglicht er schnelle multiple Strukturvergleiche über einen *Web-Service*.

Die meisten dieser Methoden führen einen *Template*-basierten Vergleich durch. Das heißt, dass der Ähnlichkeitswert, der einem Paar zugewiesen wird, zum einen von der Reihenfolge der Eingabe abhängt und zum anderen von den Proteinen abhängt, die Teil des multiplen Vergleichs sind, aber nicht zu dem gerade betrachteten Paar gehören. Als Ähnlichkeitswerte werden paarweise durchschnittliche Abstände der Strukturen zueinander ausgegeben.

Im Gegensatz zu diesen rein algorithmischen Methoden verwenden Datenbanken wie CATH [35] und SCOPe [15] eine Mischung aus algorithmischen Methoden und manueller Einordnung durch Experten, um Proteine in strukturelle Klassen einzuordnen.

Graphlets wurden zuerst von *Pržulj et al.* auf biologische Daten angewandt [28], [29]. Sie nutzten den *Graphlet*-Algorithmus, um Ähnlichkeiten von Protein-Protein-Interaktionsnetzwerken zu berechnen.

N. Shervashidze war die erste, die *Graphlets* zur Analyse von Proteinstrukturen anwandte [33]. Sie nutzte *Support Vector Machines* mit *Graphlet*-Vektoren, um für Proteingraphen zu entscheiden, ob diese Enzyme darstellen oder nicht.

Auch von *Tatiana Bakirova* wurden *Graphlets* verwendet, um Proteinstrukturen zu analysieren [3]. Sie hat das Programm **graphletAnalyser** verfasst, das in dieser Arbeit genutzt und weiterentwickelt wurde.

1.3 Ziele

Ziel dieser Arbeit war zunächst die Erweiterung der Funktionalität von **graphletAnalyser**. Hierzu gehört zunächst eine funktionierende Datenbankanbindung, um die berechneten Daten abzuspeichern. Die Ausgabe sollte so erweitert werden, dass das Berechnen der Verteilungen der relativen Häufigkeiten der *Graphlets* möglich wird. Das Berechnen von markierten *Graphlets* sollte so implementiert werden, dass es auf Graphen mit beliebigen Markierungen angewandt werden kann. Deshalb wurde ein Algorithmus entwickelt und implementiert, der aus einem Alphabet von Knotenmarkierungen alle Worte berechnet, die markierte 2- und 3-*Graphlets* repräsentieren – der *Graphlet*-Worte-Algorithmus. Die Suche nach diesen markierten *Graphlets* in einem Graphen wurde ebenfalls implementiert.

In Fallstudien wird überprüft, ob und inwiefern sich *Graphlets* eignen, um die Ähnlichkeit von Proteinstrukturtopologien zu untersuchen. Dies wurde mit unterschiedlichen Metriken getestet. Die hierbei errechneten Ähnlichkeitswerte wurden mit den Ergebnissen des Strukturalignment-Programms *PDBeFold* [23] verglichen.

Weiterhin wurde untersucht, wie die relativen Häufigkeiten der *Graphlets* in einem großen Faltungsraum verteilt sind. Um zu überprüfen, ob sich Superfamilien anhand dieser Verteilungen voneinander abgrenzen lassen, wurden diese Häufigkeiten für den PDBTop500-Datensatz mit einem Datensatz aus Aldolasen verglichen.

1.4 Aufbau der Arbeit

Zunächst wird im Kapitel *Materialien und Methoden* die *Protein Topology Graph Library* (PTGL) [25] vorgestellt, deren Idee die Grundlage für diese Arbeit liefert.

Es folgt eine Kurzbeschreibung von PLCC, der *Software*, die die Graphen der PTGL erstellt und diese verwaltet, sowie eine Beschreibung des *Graphlet*-Algorithmus.

Weiterhin wird das Programm **graphletAnalyser** vorgestellt, welches den *Graphlet*-Algorithmus implementiert. Anschließend wird die erste Metrik vorgestellt, mit denen die erhaltenen *Graphlet*-Vektoren verglichen werden. Es folgt eine Beschreibung der verwendeten Datensätze und von *PDBeFold*, dessen Ergebnisse mit denen von **graphletAnalyser** verglichen werden.

Im Ergebnisteil wird zunächst der modifizierte Jaccard-Index präsentiert, der sich mit leichten Änderungen am Tanimoto-Koeffizienten orientiert. Dann folgen Beschreibungen des neuen *Graphlet*-Worte-Algorithmus und der in den Fallstudien erhaltenen Ergebnisse.

Im abschließenden Teil *Diskussion und Ausblick* wird versucht, die Frage zu klären, ob sich *Graphlets* für den multiplen Proteinstrukturvergleich und ähnliche Anwendungen eignen. Weiterhin wird untersucht, ob Modifikationen der *Graphlet*-Vektoren oder der Metriken nötig sein könnten, um bessere Ergebnisse zu erhalten.

Kapitel 2

Materialien und Methoden

2.1 *Protein Topology Graph Library*

Die *Protein Topology Graph Library* entstand aus einer Idee von *Patrick May* und *Ina Koch* [25]. Ausgehend von der Tatsache, dass sich Proteinstrukturtopologien als räumliche Beziehungen von SSEen untereinander definieren lassen, verwendet die PTGL *Graphen*, um Proteinstrukturtopologien darzustellen. Hierbei stellen die Knoten des Graphen die SSEe eines Proteins dar. Sie werden dem jeweiligen SSE entsprechend markiert. Knoten, die α -Helices repräsentieren, werden mit einem H markiert, β -Faltblätter mit einem E. Weiterhin ermöglicht die PTGL die Darstellung von Liganden, [31] denen mit L markierte Knoten zugeordnet werden. Um die räumliche Nachbarschaft von SSEs und Liganden mit- und untereinander darstellen zu können, werden ungerichtete Kanten zwischen Knoten gezogen, wenn die entsprechenden Elemente benachbart sind. Jede Polypeptidkette eines Proteins wird dann als *Proteingraph* dargestellt. Die Zusammenhangskomponenten eines Proteingraphen werden als Faltungsgraphen bezeichnet, weil sie typischerweise eine unabhängige Faltungseinheit darstellen.

Durch diese abstrahierte Darstellung können zentrale Charakteristika eines Proteins wie Motive und Domänen einfach visualisiert werden.

2.2 PLCC

PLCC ist die *Software*, die die Daten der PTGL generiert und verwaltet. Sie wurde von *Tim Schäfer* geschrieben und wird von ihm verwaltet.

Die Berechnung der Graphen der PTGL erfolgt unter Verwendung der entsprechenden PDB- und DSSP-Dateien. Um den Graphen für eine Polypeptidkette zu berechnen, werden aus der DSSP-Datei die SSEe des Proteins ausgelesen. Für jedes Paar von SSEen wird die Anzahl der räumlichen Kontakte der zugehörigen Aminosäurereste in der PDB-Datei berechnet. Wenn die Anzahl dieser Kontakte einen gewissen Grenzwert überschreitet, wird angenommen, dass diese SSEe räumlich benachbart sind und die jeweiligen Knoten werden durch eine Kante verbunden. So wird für jede Polypeptidkette ein Graph erstellt.

Komplexgraphen werden ebenfalls in dieser Arbeit untersucht. Ihre Berechnung erfolgt analog zur Berechnung der Proteingraphen. Der Unterschied besteht darin, dass ein Komplexgraph mehrere Polypeptidketten beschreibt.

Aminosäuregraphen werden analog zu Proteingraphen und Komplexgraphen berechnet. Der Unterschied zu den anderen Graphformaten ist, dass keine SSEs betrachtet werden. Stattdessen repräsentiert jeder Knoten eine Aminosäure eines Proteins. Die Knoten werden entsprechend der chemischen Eigenschaften der Aminosäuren markiert. Knoten, die saure oder basische Aminosäurereste darstellen, werden mit einem c markiert. Ein p markiert Knoten als polare Reste, die weder sauer noch basisch sind. Für unpolare Aminosäuren wird ein h verwendet. Auch Liganden können in Aminosäuregraphen dargestellt werden. Ihre Knoten werden durch ein ? markiert. Aminosäuregraphen können Proteinkomplexe und einzelne Polypeptidketten darstellen.

Weiterhin ermöglicht PLCC den Vergleich von *Graphlet*-Vektoren, die im folgenden Teil vorgestellt werden.

2.3 Der *Graphlet*-Algorithmus

Um diese Graphen vergleichen zu können, werden *Graphlets* verwendet. Diese sind im Gegensatz zu Graph-Isomorphismen in polynomieller Zeit berechenbar [33]. *Graphlets* sind kleine, induzierte Teilgraphen mit bis zu fünf Knoten. Die Abbildungen 5.1, 5.2 und 5.3 zeigen die *Graphlets* mit drei, vier und fünf Knoten. Wir betrachten hierbei ausschließlich zusammenhängende *Graphlets*. Sie werden durch den Algorithmus gezählt; ihre jeweilige relative Häufigkeit wird in einen Vektor geschrieben, den wir als *Graphlet*-Vektor bezeichnen.

2.3.1 Beschreibung des Algorithmus

Um alle *Graphlets* der Größe $k \in \{3, 4, 5\}$ zu zählen, werden alle Euler-Wege der Länge $k - 1$ in dem gegebenen Graphen gesucht. Für jeden gefundenen Euler-Weg werden alle inzidenten Kanten aller Knoten des Weges überprüft. Wird hierbei ein *Graphlet* gefunden, wird der Zähler für das entsprechende *Graphlet* im *Graphlet*-Vektor um den Wert an der entsprechenden Stelle der unten stehenden Gewichtungsvektoren w_k erhöht.

Für *Graphlets* mit $k \geq 4$ existieren zusätzlich die sogenannten Stern-*Graphlets*. Das sind g_4 in der Abbildung 5.2, sowie g_{19}, g_{20} und g_{21} in der Abbildung 5.3. Diese enthalten keinen Euler-Weg der Länge $k - 1$.

Graphlet-Gewichtungsvektoren

$$w_2 := \left(\frac{1}{2}\right)$$

$$w_3 := \left(\frac{1}{6}, \frac{1}{2}\right)$$

$$w_4 := \left(\frac{1}{24}, \frac{1}{12}, \frac{1}{4}, 1, \frac{1}{8}, \frac{1}{2}\right)$$

$$w_5 := \left(\frac{1}{120}, \frac{1}{72}, \frac{1}{48}, \frac{1}{36}, \frac{1}{28}, \frac{1}{20}, \frac{1}{14}, \frac{1}{10}, \frac{1}{12}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, \frac{1}{12}, \frac{1}{12}, \frac{1}{4}, \frac{1}{4}, \frac{1}{2}, 1, \frac{1}{2}, 1\right)$$

Jede Stelle eines Gewichtungsvektors w_k ist mit einem *Graphlet* assoziiert. Die Nenner der Brüche in den Vektoren entsprechen den Anzahlen der Euler-Wege der Länge $k - 1$ in den entsprechenden *Graphlets*. Wenn alle Euler-Wege eines *Graphlets* besucht wurden und der zugehörige induzierte Teilgraph ihm entspricht, wird sein Zähler im *Graphlet*-Vektor ganzzahlig, und das *Graphlet* gilt als gefunden. Hierbei bilden die Stern-*Graphlets*, die keine Euler-Wege der Länge $k - 1$ enthalten, die Ausnahme.

Da wir 29 verschiedene *Graphlets* betrachten, hat ein *Graphlet*-Vektor 29 Stellen. An jeder Stelle steht die relative Häufigkeit eines *Graphlets*. Damit ist ein Vergleich in konstanter Zeit möglich. Anstelle von Graphen oder dreidimensionalen Körpern werden mit diesem approximativen Verfahren Vektoren im Vektorraum \mathbb{R}^{29} verglichen.

2.4 graphletAnalyser

In seiner ursprünglichen Version wurde das Programm bereits 2013 von *Tatiana Bakirova* geschrieben. Im Sommer 2015 wurde es um einige Funktionen erweitert.

graphletAnalyser ist in C++ geschrieben. Zur internen Darstellung der Graphen wird die *Boost-Graph-Library* verwendet.

Die Anbindung an die *Postgresql*-Datenbank wird durch die Bibliothek *pqxx* realisiert.

Als Input erhält das Programm eine oder mehrere GML-Dateien, die Graphen darstellen. Aus jeder GML-Datei wird mit der *Boost Graph Library* intern ein Graph erstellt, der für die Berechnungen verwendet wird. Für den eingelesenen Graphen berechnet das Programm alle *Graphlets* mit drei, vier und fünf Knoten.

Die Implementierung entspricht der *Matlab*-Implementierung von *N. Shervashidze* [33]. Zusätzlich berechnet es markierte *Graphlets* der Größen 2 und 3, wobei die Markierungen den Zuordnungen der SSEe durch die PTGL entsprechen. Es gibt 35 verschiedene markierte *Graphlets* mit 2 oder 3 Knoten. Unter Berücksichtigung dieser Markierungen erhöht sich die Anzahl der Stellen der *Graphlet*-Vektoren auf 64. Die Knotengradverteilungen können ebenfalls berechnet werden.

Der Output des Programms umfasst zum einen die Ausgabe der *Graphlet*-Vektoren mit oder ohne markierte *Graphlets* in den Formaten csv, matlab und nova. Zusätzlich können die *Graphlet*-Vektoren in einer Datenbank, die zuvor mit PLCC erstellt wurde, gespeichert werden. Die

Knotengradverteilung kann zusammen mit anderen Daten eines Graphen als csv-Datei ausgegeben werden.

2.5 Ähnlichkeitsmaß

Durch die Verwendung der *Graphlets* wird der direkte Vergleich von Graphen durch einen Vergleich von *Graphlet*-Vektoren ersetzt. Typischerweise wird eine Abstandsmessung durchgeführt, um Vektoren zu vergleichen. Hier verwenden wir eine Metrik von *Pržulj et al.*.

2.5.1 Relative *Graphlet*-Häufigkeiten-Distanz

Pržulj et al. haben *Graphlets* bereits auf Protein-Protein-Interaktionsnetzwerke [28] angewandt. Als Maß für die Ähnlichkeit von Netzwerken nutzen sie die Relative *Graphlet* Häufigkeiten Distanz (RGF). Diese Metrik berechnet den Abstand $D(G, H)$ zwischen zwei Graphen G und H als logarithmierte Differenz der normalisierten Anzahl der *Graphlets* in G und H . Sie ist folgendermaßen definiert:

Sei $N_i(G)$ die Anzahl der *Graphlets* vom Typ $i \in 1, \dots, 29$ und $T(G) = \sum_{i=1}^{29} N_i(G)$ die Anzahl der *Graphlets* in G beziehungsweise H .

Dann ist $D(G, H)$ für zwei Graphen G und H definiert als:

$$D(G, H) := \sum_{i=1}^{29} |F_i(G) - F_i(H)|$$

mit $F_i(G) := -\log\left(\frac{N_i(G)}{T(G)}\right)$.

Diese Metrik lässt sich analog zur euklidischen Distanz auffassen, denn sie berechnet den Abstand für zwei Vektoren x, y als Differenz ihrer Einträge $x_i - y_i$. Sie verwendet die normalisierten *Graphlet*-Vektoren unter der Annahme, dass die Ähnlichkeit zweier Netzwerke sich aus der Ähnlichkeit lokaler Substrukturen ableiten lässt [28]. Somit können Netzwerke, die ähnliche Substrukturen haben, sich aber in ihrer Größe stark unterscheiden, immer noch als ähnlich erkannt werden. Zusätzlich soll durch diese Normalisierung ausgeschlossen werden, dass einzelne, besonders häufige *Graphlets* in einem Graphen die Messung verzerren.

Alle Einträge der Vektoren stammen aus dem Intervall $[0, 1] \in \mathbb{R}$. Diese werden logarithmiert. Damit erhält man Abstände, die aus deutlich größeren Intervallen stammen. Die Ergebnisse können so leichter interpretiert werden und der Einfluss von Rundungsfehlern verkleinert sich. Dies sorgt auch dafür, dass seltenere *Graphlets* einen stärkeren Einfluss auf die Messung haben, als häufige *Graphlets*.

Weiterhin wurde gezeigt, [28] dass diese Metrik auch bei verrauschten Daten noch sehr gut funktioniert. Hierbei ist jedoch zu beachten, dass sie bisher vor allem für sehr große Netzwerke mit mehreren Tausend Knoten und Kanten verwendet wurde. Diese Größe kann von Aminosäuregraphen erreicht werden, wenn sie große Proteinkomplexe modellieren. Proteingraphen und Komplexgraphen sind aber deutlich kleiner.

2.6 *PDBeFold*

Im Rahmen der Fallstudie wird **graphletAnalyser** mit *PDBeFold* [23] verglichen. *PDBeFold* führt ebenfalls einen graphenbasierten Strukturvergleich durch. Analog zur PTGL modellieren Knoten SSEs. Zusätzlich werden diese mit der Anzahl der Aminosäuren des jeweiligen SSEs markiert. Im Gegensatz zur PTGL verbindet *PDBeFold* jeden Knoten durch eine Kante mit jedem anderen Knoten. Die Kanten werden hierbei mit Vektoren markiert, in denen unter anderem die Distanzen und Winkel zwischen den SSEs eingetragen sind.

Die Proteine werden somit als vollständige, ungerichtete Graphen modelliert, in denen jede Kante alle Informationen zur räumlichen Orientierung ihrer inzidenten Knoten enthält. Diese Graphen werden vorberechnet.

Zum Vergleich von Strukturen wird für diese Graphen zunächst ein Alignment durchgeführt, um eine Abschätzung der strukturellen Ähnlichkeit zu erhalten und die äquivalenten Residuen zu ermitteln. Im Anschluss führt

PDBeFold ein Alignment der C_α -Atome durch und berechnet die Ähnlichkeit der Proteine als durchschnittlichen paarweisen Abstand äquivalenter Residuen. Als Abstandsmaß verwendet es die *Root Mean Square Deviation*.

Da **graphletAnalyser** Vektoren berechnet, die verglichen werden, ist ein direkter Vergleich der Abstände aus beiden Programmen nicht möglich. Der Vergleich kann nur durch einen Vergleich relativer Abstände innerhalb der Datensätze erfolgen.

2.7 Datensätze

Im folgenden Teil sind die verwendeten Datensätze beschrieben. Es wurde eine Fallstudie mit Abstandsmessungen durchgeführt, um diese zu testen.

Zusätzlich wurden *Graphlet*-Vektoren für einen Datensatz aus knapp 500 nicht-redundanten PDB-Einträgen berechnet. Für diesen Datensatz wurden keine Abstandsmessungen durchgeführt, stattdessen wurden die *Graphlet*-Vektoren selbst statistisch analysiert.

2.7.1 Fallstudie

Dieser Datensatz enthält 15 verschiedene Proteine mit bekannten strukturellen Ähnlichkeiten. Die Proteine wurden so gewählt, dass ihre Einordnungen in strukturelle Klassen in den Datenbanken CATH [35] und SCOPe [15] äquivalent zueinander sind. Der Datensatz wurde also so zusammengestellt, dass bezüglich der strukturellen Einteilung der Proteine der größtmögliche Konsens besteht. Je fünf Proteine dieses Datensatzes befinden sich in der gleichen Klasse. Von diesen fünf besitzen je vier die gleiche Topologie und von diesen entstammen wiederum drei der gleichen homologen Superfamilie. Von diesen ist ein Paar direkt homolog mit einer Sequenzidentität von mehr als 95%. Das dritte Protein ist zu den anderen beiden entfernter homolog mit einer Sequenzidentität von weniger als 65%.

Es sind Proteine der Klassen α , β und α/β vertreten. Proteine mit wenigen SSEs wurden nicht betrachtet, da für diese die Proteingraphen und die Komplexgraphen in den meisten Fällen keine Kanten besitzen. Dadurch können keine *Graphlets* in diesen Graphen gefunden werden.

Weiterhin wurde darauf geachtet, dass jedes Protein dieses Datensatzes aus genau einer Polypeptidkette mit genau einer Domäne besteht. Dies hat mehrere Gründe. Zum ersten ermöglicht diese

Auswahl die beste Vergleichbarkeit der Messungen mit Proteingraphen, Komplexgraphen und Aminosäuregraphen, da Proteingraphen im Gegensatz zu Komplexgraphen und Aminosäuregraphen nur eine Polypeptidkette darstellen können. Durch die Beschränkung auf Proteine mit genau einer Polypeptidkette wird ein direkter Vergleich durch die Korrelation der paarweisen Distanzen in den unterschiedlichen Graphentypen möglich. Hier werden Liganden in den Komplexgraphen einbezogen und in den Proteingraphen ignoriert.

Ein weiterer Grund für die Beschränkung auf solche Proteine ist, dass man ein „klareres Signal“ erhält. Die Struktur von Proteinen wird hauptsächlich anhand ihrer Domänen beschrieben. Wenn *Graphlets* sich wirklich für den Vergleich von Topologien eignen, dann müssten gleiche Domänen zu gleichen oder zumindest ähnlichen *Graphlet*-Vektoren führen. Da diese Vektoren globale Struktureigenschaften des Proteins beschreiben, erhielte man für ein Multidomänenprotein mit unterschiedlichen Domänen einen Vektor, der die Struktureigenschaften beider Domänen „vermischt“ und das Signal verrauscht.

In den Tabellen 2.1, 2.2 und 2.3 sind die Proteine der Fallstudie dargestellt.

Für diesen Datensatz wurden alle *Graphlet*-Vektoren berechnet. Die paarweisen Distanzen dieser Vektoren wurden mit der RGF und dem modifizierten Jaccard-Index berechnet. Für die berechneten Distanzmatrizen wurden die Korrelationen der Distanzen in den verschiedenen Formaten berechnet und sie wurden mit den von PDBFold berechneten Distanzen verglichen.

2.7.2 Der PDBTop500-Datensatz

Der PDBTop500-Datensatz wurde von Lovell *et al.* [24] vorgestellt. Diese nutzten ihn zur Validierung von PDB-Strukturvorhersagen unter Verwendung der C_α -Geometrie. Der Datensatz enthält 500 nicht redundante PDB-Einträge mit einer Auflösung von 1,8Å oder besser.

Er wurde ausgewählt, weil die *Graphlet*-Vektoren für ihn schnell berechenbar waren. Im Gegensatz zu anderen Datensätzen mit nicht redundanten PDB-Einträgen wie dem FATCAT- oder dem ASTRAL-Datensatz enthält dieser keine riesigen Proteinkomplexe, deren *Graphlets* auf durchschnittlichen Computern nicht in weniger als 24 Stunden berechnbar sind. Er ist aber immer noch groß genug, um Strukturvergleichsmethoden validieren zu können. Zhang und Skolnick [40] verwendeten zur Validierung von *TM-align* einen Datensatz aus 200 PDB-Einträgen. Da ein Vergleich der paarweisen Distanzen auf einem solchen Datensatz zu aufwändig war, wurde dieser Datensatz für eine statistische Analyse der *Graphlet*-Vektoren genutzt. Für die relativen Häufigkeiten der einzelnen *Graphlets* wurden Minima, Maxima, Varianzen und Durchschnittswerte berechnet. Hierbei war das Ziel festzustellen, ob wirklich alle *Graphlets* für Strukturvergleiche notwendig sind.

Weiterhin wurde ein Datensatz aus Aldolasen zusammengestellt, um festzustellen, ob sich die Verteilungen der *Graphlet*-Häufigkeiten, zwischen dem großen Datensatz und den Aldolasen unterscheiden. Wenn der Unterschied der Verteilungen zwischen diesen beiden Datensätzen so groß ist, dass eine klare Unterscheidung möglich ist, dann sollte auch eine maschinelle Klassifizierung der *Graphlets* für Aldolasen möglich sein. Aldolasen wurden gewählt, weil diese Superfamilie strukturell besonders divers ist, wie Das *et al.* [9] zeigten. Dabei beträgt die Sequenzidentität der Proteine dieser Klasse immer mindestens 35%. Wenn die *Graphlet*-Verteilungen der Aldolasen und des PDBTop500-Datensatzes klar unterscheidbar sind, dann sollten sie auch für Superfamilien, die strukturell einheitlicher sind, unterscheidbar sein. Die Aldolasen, die hier betrachtet sind in den Tabellen 2.4 und 2.5 aufgeführt.

Tabelle 2.1: α -Proteine der Fallstudie

Alle Proteine dieser Tabelle gehören zur Architektur der *Up Down Bundles*. Die Bilder der 3D-Strukturen und die Beschreibung der Einträge stammen aus der PDB und aus UniProt [8]. Die Einordnung der Topologie und der Superfamilie stammt aus CATH.

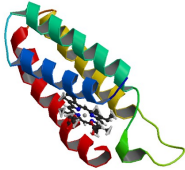
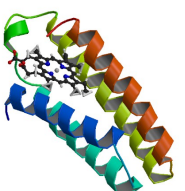
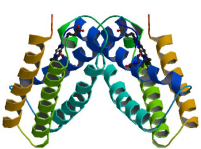
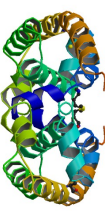
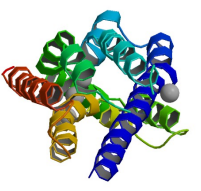
PDB-ID	3D Bild	Struktur	Beschreibung
1QPU		Topologie: <i>Four Helix Bundle</i> Superfamilie: CATH: 1.20.120.10	Der Eintrag beschreibt die Struktur des oxidierten Cytochrom B562 aus <i>E. coli</i> [2]. Es ist am Elektronentransport beteiligt.
1QQ3		Topologie: <i>Four Helix Bundle</i> Superfamilie: CATH: 1.20.120.10	Die Hem-bindende Variante des oxidierten Cytochrom B562 aus <i>E. coli</i> wird durch diesen Eintrag beschrieben [1]. Auch diese Variante des Proteins ist für Elektronentransport zuständig.
1CGN		Topologie: <i>Four Helix Bundle</i> Superfamilie: CATH: 1.20.120.10	Dieser Eintrag beschreibt die Struktur von Cytochrom C' [11] aus <i>Achromobacter xylosoxidans</i> . Es ist ebenfalls für Elektronentransport zuständig.
1HE9		Topologie: <i>Four Helix Bundle</i> Superfamilie: <i>Bacterial GAP Domain</i>	Der Eintrag beschreibt die GTPase aktivierende Domäne des Toxins Exoenzym S aus <i>Pseudomonas aeruginosa</i> [38].
3GF9		Topologie: <i>Dbl homology Domain</i> Superfamilie: DBL Homology Domain	Die RhoGEF-Domäne des Proteins Intersectin 2 aus <i>Homo sapiens</i> wird durch diesen Eintrag beschrieben [32]. Dieses Protein ist für Endocytose zuständig.

Tabelle 2.2: β -Proteine der Fallstudie

Die Proteine in dieser Tabelle gehören zur Architektur der β -Barrels. Die Bilder der 3D-Strukturen und die Beschreibung der Einträge stammen aus der PDB und aus UniProt. Die Einordnung der Topologie und der Superfamilie stammt aus CATH.


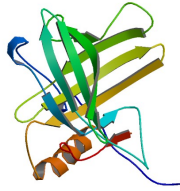
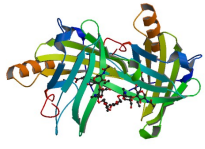
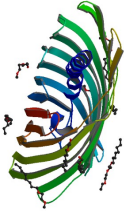
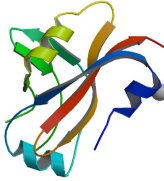
PDB-ID	3D Bild	Struktur	Beschreibung
1EXS		Topologie: <i>Lipocalin</i> Superfamilie: CATH: 2.40.128.20	Der Eintrag beschreibt die Struktur des β -Lactoglobulin aus <i>Sus scrofa</i> [19]. Es ist ein lipidbindendes Protein.
1NGL		Topologie: <i>Lipocalin</i> Superfamilie: CATH: 2.40.128.20	Das neutrophile Gelatinase assoziierte Lipocalin des <i>Homo sapiens</i> wird durch diesen Eintrag beschrieben [7]. Es ist ein Transportprotein.
1QQS		Topologie: <i>Lipocalin</i> Superfamilie: CATH: 2.40.128.20	Die Struktur des neutrophilen Gelatinase assoziierten Lipocalin Homodimers wird durch den Eintrag beschrieben [16]. Es ist ein zuckerbindendes Protein.
3SLO		Topologie: <i>Lipocalin</i> Superfamilie: Autotransporter Esterase	Der Eintrag beschreibt die Struktur der N1023D Mutante des Autotransporters EspP [4] aus <i>E. coli</i> . Dieses Protein ist für den Transport von Proteinen zuständig.
1WJX		Topologie: <i>Small Protein B</i> Superfamilie: 2.40.280.10	Dieser Eintrag beschreibt das Protein TT0801 aus <i>Thermus thermophilus</i> [6].

Tabelle 2.3: α/β -Proteine der Fallstudie

Die hier dargestellten Proteine haben eine *3-Layer-Sandwich*-Architektur. Die Bilder und Beschreibungen entstammen UniProt und der PDB. Die Beschreibung der Struktur stammt aus

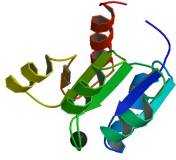
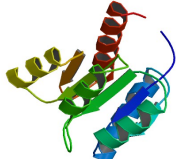
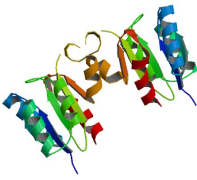

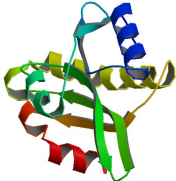
	PDB-ID	3D Bild	Struktur	Beschreibung
CATH.	5CHY		Topologie: <i>Rossman Fold</i> Superfamilie: CATH: 3.40.50.2300	Der Eintrag beschreibt die Struktur einer Mutante des Chemotaxisproteins des Gens <i>cheY</i> aus <i>E. coli</i> [41]. Es ist ein Signaltransduktionsproteinprotein.
	2ID9		Topologie: <i>Rossman Fold</i> Superfamilie: CATH: 3.40.50.2300	Der Eintrag beschreibt das synthetische Protein T87I phosphono-CheY [26]. Es ist ein stabiles Analogon zu dem oben beschriebenen Chemotaxisprotein.
	3I42		Topologie: <i>Rossman Fold</i> Superfamilie: CATH: 3.40.50.2300	Der Eintrag beschreibt eine Empfängerdomäne aus <i>Methylobacillus flagellatus</i> , die regulatorische Signale empfängt und ähnlich zu <i>cheY</i> ist [30].
	1D4O		Topologie: <i>Rossman Fold</i> Superfamilie: <i>TPP-binding domain</i>	Die Struktur der Transhydrogenase Domäne II aus <i>Bos taurus</i> [27] wird durch diesen Eintrag beschreiben. Sie wird in der PDB als Oxidoreductase klassifiziert.
	2W0I		Topologie: Severin Superfamilie: Severin	Der Eintrag beschreibt eine Domäne des Proteins TWINFLIN-2 aus <i>Homo sapiens</i> [12]. Dieses Protein inhibiert die Polymerisation von Actin.

Tabelle 2.4: Aldolasen

1LBM, 1NSJ und 1V5X sind in die Synthese von L-Tryptophan eingebunden [17], [18], [36]. 3CEU ist eine Transferase [14] und 7TIM nimmt an der Gluconeogenese teil [10]. Die Bilder und Beschreibungen stammen aus der PDB und aus UniProt.

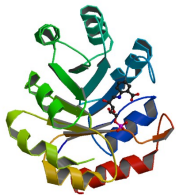
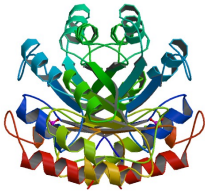
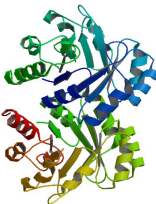
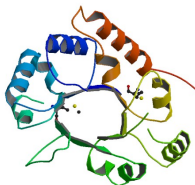
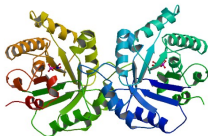
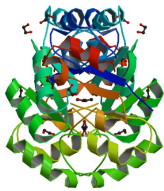
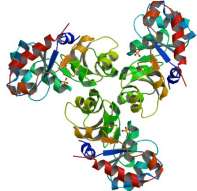
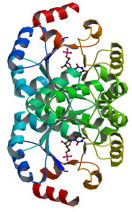
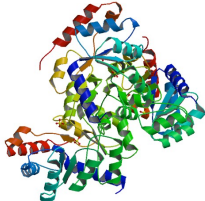
PDB-ID	3D Bild	Struktur	Beschreibung
1LBM		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Der Eintrag beschreibt die Phosphoribosyl Anthranilat Isomerase im Komplex mit einem Liganden aus <i>Thermotoga maritima</i> .
1NSJ		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Dieser Eintrag beschreibt das gleiche Protein wie der Eintrag 1LBM mit veränderter Konformation ohne Liganden.
1V5X		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Auch dieser Eintrag beschreibt die Struktur der Phosphoribosyl Anthranilat Isomerase aus <i>Thermotoga maritima</i> mit veränderter Konformation.
3CEU		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Hier wird die Struktur der Thiamin Phosphat Posophorylase aus <i>E.coli</i> beschrieben.
7TIM		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Dieser Eintrag beschreibt einen Komplex aus der Triosephosphat Isoermase und Phophoglycolhydroxamat aus <i>Saccharomyces cerevisiae</i> .

Tabelle 2.5: Aldolasen - Lyasen

Alle Proteine in der untenstehenden Tabelle werden von der PDB als Lyasen klassifiziert. Die Beschreibungen stammen aus UniProt, die Bilder stammen aus der PDB.

PDB-ID	3D Bild	Struktur	Beschreibung
1VQT		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Dieser Eintrag beschreibt die Struktur der Oritidin 5'-Phosphatase Decarboxylase aus <i>Thermotoga maritima</i> [13]. Sie ist in die Synthese von UMP eingebunden.
1WAU		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Der Eintrag beschreibt das synthetische Protein T87I phosphono-CheY [26]. Es ist ein stabiles Analogon zu dem oben beschriebenen Protein.
1LOR		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Die Struktur der Orotidin 5'-Monophosphatase aus <i>Methanothermobacter thermautotrophicus</i> im Komplex mit einem Liganden wird durch diesen Eintrag beschrieben [37]. Das Protein ist an der Decarboxylierung von Oritidin beteiligt.
1MXS		Topologie: TIM <i>Barrel</i> Superfamilie: Aldolasen - Klasse I	Dieser Eintrag beschreibt die 2-Keto-3-deoxy-6-Phosphogluconat Aldolase aus <i>Pseudomonas putida</i> [5]. Genau wie die obigen Proteine wird dieses durch die PDB als Lyase klassifiziert.

Kapitel 3

Ergebnisse

3.1 Der modifizierte Jaccard-Index

Der Jaccard-Index ist ein Maß, um die Ähnlichkeit von gleich mächtigen Mengen zu bewerten. Für zwei Mengen A, B berechnet sich der Jaccard-Index $D_{Jac}(A, B)$ folgendermaßen:

$$D_{Jac}(A, B) := \frac{\sum_{x \in A \cap B} 1}{\sum_{x \in A \cup B} 1}.$$

Dementsprechend sind zwei Mengen A, B gleich, wenn $D_{Jac} = 1$ gilt und disjunkt, wenn $D_{Jac} = 0$ gilt. Mit ihm wird die relative Anzahl der Elemente beider Mengen berechnet. Um dieses Maß in sinnvoller Weise auf *Graphlet*-Vektoren zu übertragen, wurde ein zusätzlicher Faktor $k \in \mathbb{R}$ mit $k \in [0, 1]$ eingeführt, der als Präzisionsfaktor zu verstehen ist. Die Definition des modifizierten Jaccard-Index $D_{JacM}(v, w)$ für zwei Vektoren v, w lautet also:

$$D_{JacM}(v, w) := \frac{\sum_{i=1}^n x_i}{\sum_{x \in A \cup B} 1}$$

mit

$$x_i = \begin{cases} 1 & \text{if } v_i \geq w_i \times k \wedge w_i \geq v_i \times k \\ 0 & \text{else} \end{cases}.$$

In dieser modifizierten Variante werden zwei Vektoren v, w als gleich angesehen, wenn sich v_i, w_i für alle i höchstens um den Faktor k unterscheiden. Dies steht im Gegensatz zur RGF, die – analog zum euklidischen Abstand – die Abstände zwischen zwei *Graphlet*-Vektoren misst. Für alle weiteren Messungen in dieser Arbeit wurde $k = 0,9$ gewählt. Dies hatte den Grund, dass weder niedrigere noch höhere Werte zu sinnvolleren, interpretierbaren Ergebnissen führten.

3.2 Der *Graphlet*-Worte-Algorithmus

In der letzten Version von `graphletAnalyser` war es bereits möglich, markierte *Graphlets* mit 2 und 3 Knoten in Proteingraphen zu zählen. Diese Funktionalität wurde im Rahmen dieser Arbeit

verallgemeinert, so dass der Nutzer beliebige Alphabete angeben kann. Der Algorithmus erhält das Alphabet $\Sigma = \{\sigma_i : i \in \mathbb{N}\}$ der Knotenmarkierungen. Aus diesem Alphabet berechnet er Worte w , die zur Repräsentation der markierten *Graphlets* genutzt werden. Hierbei können verschiedene Worte das gleiche *Graphlet* repräsentieren. Im Falle von 2-*Graphlets* repräsentieren die zwei Worte (σ_i, σ_j) und (σ_j, σ_i) das gleiche markierte *Graphlet* mit den Knotenmarkierungen σ_i, σ_j . Worte, die das gleiche *Graphlet* repräsentieren, werden im Folgenden als *äquivalente Graphlet-Worte* bezeichnet.

Die Berechnung der äquivalenten *Graphlet*-Worte der Länge 2 ist trivial. Aus dem Alphabet Σ werden alle Worte $w = (\sigma_i, \sigma_j)$ berechnet, wobei Spiegelungen nicht mit ausgegeben werden, da zwei Worte (σ_j, σ_i) und (σ_j, σ_i) äquivalente *Graphlet*-Worte sind.

Die Berechnung aller äquivalenten *Graphlet*-Worte der Länge 3 ist komplizierter, da sie für zwei verschiedene *Graphlets* berechnet werden müssen. Für das *Graphlet* g_1 sind alle Worte äquivalent zueinander, die zyklische Vertauschungen voneinander sind. Für das *Graphlet* g_2 sind Worte äquivalent zueinander, die Spiegelungen voneinander sind (siehe Abbildung 5.1).

```

input   : Ein Alphabet  $\Sigma = \{\sigma_1, \dots, \sigma_n\}$ 
output : Zwei Listen Wortliste-3-Weg, Wortliste-3-Kreis, die die repräsentierenden
           Worte für den Kreis bzw. Weg aus 3 Knoten beschreiben

1 for  $i \leftarrow 1$  to  $n$  do
2   Wortliste-3-Weg.add( $\sigma_i \sigma_i \sigma_i$ )
3   Wortliste-3-Weg.add( $\sigma_i \sigma_i \sigma_i$ )
4   for  $k \leftarrow i + 1$  to  $n$  do
5     Wortliste-3-Weg.add( $\sigma_i \sigma_k \sigma_i$ )
6     Wortliste-3-Weg.add( $\sigma_i \sigma_i \sigma_k$ )
7     Wortliste-3-Weg.add( $\sigma_i \sigma_k \sigma_k$ )
8     Wortliste-3-Weg.add( $\sigma_k \sigma_i \sigma_k$ )
9     Wortliste-3-Kreis.add( $\sigma_i \sigma_k \sigma_i$ )
10    Wortliste-3-Kreis.add( $\sigma_i \sigma_k \sigma_k$ )
11    for  $m \leftarrow k + 1$  to  $n$  do
12      Wortliste-3-Weg.add( $\sigma_i \sigma_k \sigma_m$ )
13      Wortliste-3-Weg.add( $\sigma_m \sigma_i \sigma_k$ )
14      Wortliste-3-Weg.add( $\sigma_k \sigma_m \sigma_i$ )
15      Wortliste-3-Kreis.add( $\sigma_i \sigma_k \sigma_m$ )
16      Wortliste-3-Kreis.add( $\sigma_m \sigma_k \sigma_i$ )
17    end
18  end
19 end

```

Algorithmus 1 : *Graphlet*-Worte-Algorithmus

Der Algorithmus besteht aus drei *for*-Schleifen, die über das Alphabet iterieren. In jeder dieser Schleifen werden alle äquivalenten Worte hinzugefügt, die aus den aktuell betrachteten Buchstaben erzeugt werden können.

Für das Alphabet der Protein- und Komplexgraphen $\Sigma_{SSE} := \{H, E, L\}$ und das Alphabet der Aminosäuregraphen $\Sigma_{AA} := \{h, p, c, ?\}$ gibt der oben beschriebene Algorithmus die folgenden

Listen aus:

$$\begin{aligned} p_2 &:= (HH, HE, HL, EE, EL, LL) \\ p_{3-Weg} &:= (HHH, HEH, HHE, HEE, EHE, \\ &\quad HEL, LHE, ELH, HLH, HHL, \\ &\quad HLL, LHL, EEE, ELE, EEL, \\ &\quad ELL, LEL, LLL) \\ p_{3-Kreis} &:= (HHH, HEH, HEE, HEL, LEH, \\ &\quad HLH, HLL, EEE, ELE, ELL, LLL) \end{aligned}$$

Die Vektoren a_2 , a_{3-Weg} und $a_{3-Kreis}$ beschreiben die Worte für *Graphlets* in AA-Graphen

$$\begin{aligned} a_2 &:= (hh, hp, hc, h?, pp, pc, p?, cc, c?, ??) \\ a_{3-Weg} &:= (hhh, hph, hhp, hpp, php, hpc, chp, pch, hp?, ?hp, \\ &\quad p?h, hch, hhc, hcc, chc, hc?, ?hc, c?h, h?h, hh?, \\ &\quad h??, ?h?, ppp, pcp, ppc, pcc, cpc, pc?, ?pc, c?p, \\ &\quad p?p, pp?, p??, ?p?, ccc, c?c, cc?, c??, ?c?, ???) \\ a_{3-Kreis} &:= (hhh, hph, hpp, hpc, cph, hp?, ?ph, hch, hcc, hc?, \\ &\quad ?ch, h?h, h??, ppp, pcp, pcc, pc?, ?cp, p?p, p??, \\ &\quad ccc, c?c, c??, ???) \end{aligned}$$

Um die Korrektheit des Algorithmus zu überprüfen, wurde die Ausgabe für Alphabete mit drei und vier Buchstaben händisch überprüft.

3.3 Erweiterung von graphletAnalyser

Das Einlesen von Komplexgraphen und Aminosäuregraphen ist implementiert worden. Die entsprechenden Alphabete sind im Programmcode vordefiniert und können vom Nutzer über Parameter ausgewählt werden oder in der Konfigurationsdatei festgelegt werden.

Nutzerdefinierte Knotenmarkierungen können nun in der Konfigurationsdatei angegeben werden. Der Nutzer kann ein Alphabet von Knotenmarkierungen und ein *Label*, unter dem diese Knotenmarkierungen in den GML-Dateien gespeichert sind, angeben. Für dieses Alphabet werden alle äquivalenten *Graphlet*-Worte durch den *Graphlet*-Worte-Algorithmus berechnet. Diese werden dann bei der Berechnung der markierten *Graphlets* im Graphen unter dem vorgegebenen *Label* gesucht und gezählt. Es können also beliebige Alphabete und *Labels* angegeben werden, so lange die Markierungen der Knoten nicht mehr als einen Buchstaben enthalten.

Die Datenbankankündigung wurde um Funktionen zum Speichern von Aminosäuregraphen und Komplexgraphen erweitert. Das Speichern von Vektoren markierter *Graphlets* wurde implementiert.

Wenn die Option `--useDatabase` ausgewählt wird, prüft das Programm, ob der entsprechende Graph bereits in der Datenbank vorhanden ist. Falls der Graph gefunden wurde, wird der *Graphlet*-Vektor für den entsprechenden Graphen in die Datenbank eingetragen.

3.4 Fallstudie

Die paarweisen RGF-Distanzen der Proteine befinden sich in den Tabellen 5.1, 5.2 und 5.6. Die paarweisen Jaccard-Indizes befinden sich in den Tabellen 5.4, 5.5 und 5.6. Hierbei wurden die Zellen, die die vier besten Bewertungen für das Protein der entsprechenden Zeile enthalten, grün eingefärbt. Das Grün ist umso dunkler, je stärker die Ähnlichkeit bewertet wird.

Der Vergleich mittels RGF zeigt für die Aminosäuregraphen die stärksten Ähnlichkeiten immer innerhalb der entsprechenden CATH-Klassen. Sowohl die Proteine aus der α -Klasse als auch die Proteine der β -Klasse haben die besten Ähnlichkeitswerte mit Proteinen der gleichen Klasse. Innerhalb der Klasse der α/β -Proteine, gibt es mit 2ID9 und 1D4O zwei Proteine, denen eine größere Ähnlichkeit zu Proteinen der *mainly-alpha*-Klasse attestiert wurde. Bei 1D4O fällt auf, dass der niedrigste Wert mit 7,091 deutlich höher ist, als die besten Werte aller anderen Proteine. 2ID9 hat laut RGF-Distanz die größte Ähnlichkeit zu 1HE9. Bis auf diese beiden Ausnahmen lässt sich jedoch eine starke Korrelation mit den CATH-Klassen erkennen. Alle anderen Proteine haben mindestens die zwei kleinsten RGF-Distanzen zu Vertretern aus der selben CATH- und SCOPe-Klasse.

Dies gilt für die Aminosäuregraphen. Die RGF-Distanzen der Proteingraphen zueinander zeigen ein weniger klares Bild. Für die α -Klasse und die der β -Klasse befinden sich die Proteine mit den kürzesten Distanzen immer noch in der selben Klasse. Dies lässt sich für die Proteine der α/β -Klasse aber nicht mehr behaupten. Hier haben 2ID9, 3I42 und 2W0I die kürzesten Distanzen zu Proteinen anderer Klassen.

Bei den Komplexgraphen ist die Korrelation zwischen der RGF-Distanz und der Zugehörigkeit zur strukturellen Klasse noch geringer. Die Tabelle 5.3 zeigt nur für die Proteine 1QQ3, 1HE9, 1EXS und 1QQS die kürzeste Distanz zu einem Vertreter der gleichen Klasse. Es fällt jedoch auf, dass besonders häufig die Proteine der *alpha-beta*-Klasse 5CHY, 2ID9 und 3I42 als ähnlich zu anderen bewertet werden.

Der Vergleich der Jaccard-Indizes zeigt ein ähnliches Bild, wie der Vergleich der RGF-Distanzen. Bei den Aminosäuregraphen zeigt sich, dass innerhalb der α -Klasse wieder die paarweisen Ähnlichkeiten der α -Proteine am größten sind. Dies gilt bis auf eine Ausnahme auch für die β -Proteine. Das Protein mit der PDB-ID 1NGL wird als strukturell ähnlichstes Protein zu 2W0I bewertet. In der Klasse der α/β -Proteine gibt es mit 2ID9 und 1D4O wieder zwei Ausreißer, die die größten paarweisen Ähnlichkeiten nicht zu Vertretern der eigenen Klasse haben. Für 2ID9 wird 1HE9 als ähnlichstes Protein angegeben und 1D4O wird 1QQS zugeordnet.

Die paarweisen modifizierten Jaccard-Indizes der Proteingraphen zeigen - wie schon bei den RGF-Distanzen - eine geringere Korrelation mit der Zugehörigkeit zu den CATH-Klassen als die Koeffizienten der Aminosäuregraphen. Es haben zwar wieder mindestens drei Vertreter jeder Klasse ihren nächsten Nachbarn in der gleichen Klasse, aber es gibt auch einige Proteine, die ihren nächsten Nachbarn außerhalb der eigenen Klasse haben. Hierzu gehören die α/β -Proteine 1D4O und 2W0I und das α -Protein 3GF9. Es fällt auf, dass wieder die Proteine mit den PDB-DIs 2ID9 und 3I42 besonders häufig als strukturell ähnlich zu vielen anderen Proteinen bewertet werden.

Für die Komplexgraphen zeigt die Tabelle wieder eine hohe Ähnlichkeit unter den ersten drei Proteinen 1QPU, 1QQ3 und 1CGN. Auch innerhalb der Klasse α/β sind drei Proteine mit der höchsten paarweisen Ähnlichkeit bewertet worden. Die geringe Anzahl von stark bewerteten Ähnlichkeiten innerhalb der β -Klasse ist sehr auffällig. 1QQS und 3SLO sind das einzige Paar mit β -Topologie, dessen Ähnlichkeit als groß bewertet wurde.

3.5 PDBTop500-Datensatz

Für den PDBTop500-Datensatz wurden *Graphlet*-Vektoren der drei verschiedenen Graphenformate berechnet. Diese Vektoren wurden für jedes Format in einer .csv-Datei zusammengefasst, und für jede dieser Dateien wurden *Boxplots* erstellt. In diese *Boxplots* wurden zusätzlich die relativen Häufigkeiten der bereits beschriebenen Aldolasen eingetragen. Hiermit kann festgestellt werden, ob die Superfamilie der Aldolasen anhand ihrer *Graphlets* von anderen Superfamilien unterscheidbar ist. Der *Boxplot* in Abbildung 3.1 stellt diese Daten für die Aminosäuregraphen dar. Da die relativen Häufigkeiten der *Graphlets* in den Aminosäuren deutlich kleiner sind als bei Proteingraphen und Komplexgraphen, weil es für sie mehr *Graphlet*-Worte gibt, wurde die Skala der y-Achse logarithmiert.

Unter den *Graphlets* ohne Markierungen in Abbildung 3.1 gibt es nur zwei, bei denen die relativen Häufigkeiten der Aldolasen nicht mit denen des Datensatzes überlappen. Dies sind die *Graphlets* g_8 und g_{15} in Abbildung 5.3, die in den Aldolasen nicht gefunden werden.

Unter den markierten *Graphlets* gibt es deutlich mehr Unterschiede. Sieben verschiedene *Graphlet*-Verteilungen überlappen sich nicht. Die ersten drei hiervon entsprechen den *Graphlet*-Worten $h?$, $c?$ und $??$. Diese stehen für Kanten zwischen unpolaren Residuen und Liganden, Kanten zwischen sauren oder basischen Residuen und Liganden und Kanten zwischen Liganden. Die nächsten drei Worte, für die keine Überlappungen gefunden wurden, sind die Worte hhh , hph und hhp . Diese repräsentieren Wege der Länge 2 zwischen drei unpolaren Aminosäuren, zwei Aminosäuren mit einer polaren Aminosäure als inneren Knoten und zwei unpolaren Aminosäuren mit einer polaren Aminosäure als Endknoten.

Bei den Proteingraphen in Abbildung 3.2 sind die *Graphlet*-Verteilungen ein wenig stärker voneinander getrennt. Dies gilt für das 4-*Graphlet* g_6 und das 5-*Graphlet* g_{13} . Unter den markierten *Graphlets* ist die Trennung für die am stärksten, die durch die Worte HL und HLL markiert werden. Ersteres repräsentiert den Kontakt einer α -Helix mit einem Liganden. Letzteres markiert einen Kreis, in dem zwei Liganden mit einer Helix in Kontakt stehen.

Für die Komplexgraphen in Abbildung 3.3 finden wir die stärksten Unterschiede zwischen den Verteilungen der Aldolasen und denen des Referenzdatensatzes. Dies gilt für die unmarkierten *Graphlets* g_1 und g_2 in Abbildung 5.1, g_2 , g_3 und g_6 in Abbildung 5.2 sowie g_6 , g_{10} , g_{12} , g_{13} , g_{16} , g_{18} , g_{20} und g_{21} in Abbildung 5.3. Die stärksten Abweichungen der Verteilungen finden wir bei den markierten *Graphlets* für jene, die durch die Worte HL, LLL und EEE repräsentiert sind. Für das Wort LLL finden wir diese Abweichungen sowohl für das markierte *Graphlet*, das g_1 in 5.1 entspricht, als auch für jenes, das g_2 repräsentiert. Damit sind die Unterschiede der *Graphlet*-Verteilungen zwischen dem Referenzdatensatz und den Aldolasen in den Komplexgraphen am größten.

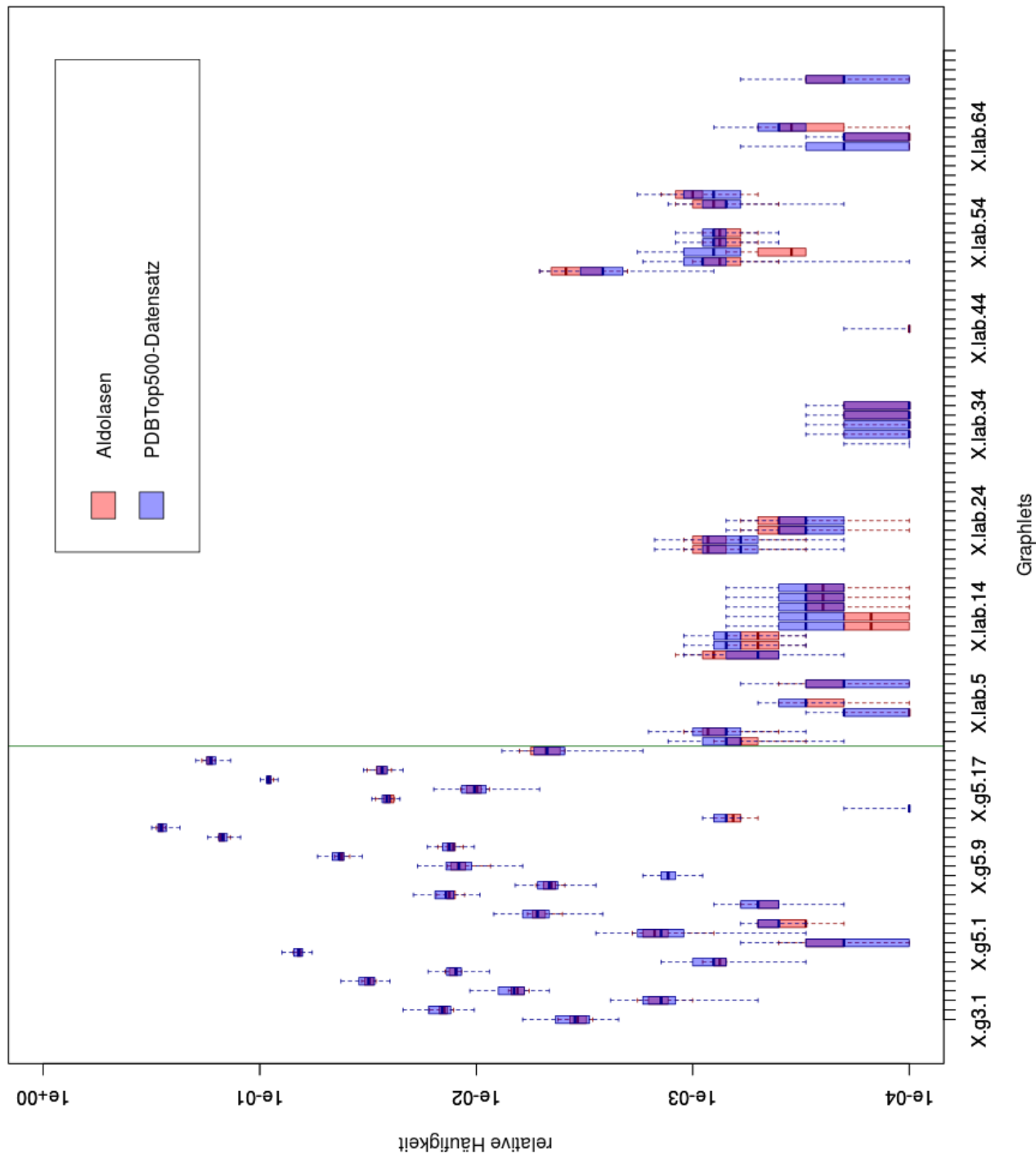


Abbildung 3.1: *Graphlet*-Verteilungen für Aminosäuregraphen

Jeder Punkt auf der x-Achse entspricht einem *Graphlet*. Die y-Achse ist logarithmiert und beschreibt die relative Häufigkeiten. Die Häufigkeiten der Proteine des PDBTop500-Datensatzes sind in blau, die der Aldolasen in rot angegeben. Links von der vertikalen Linie sind *Graphlets* ohne Markierungen, rechts davon sind *Graphlets* mit Markierungen. Ausreißer sind nicht eingetragen.

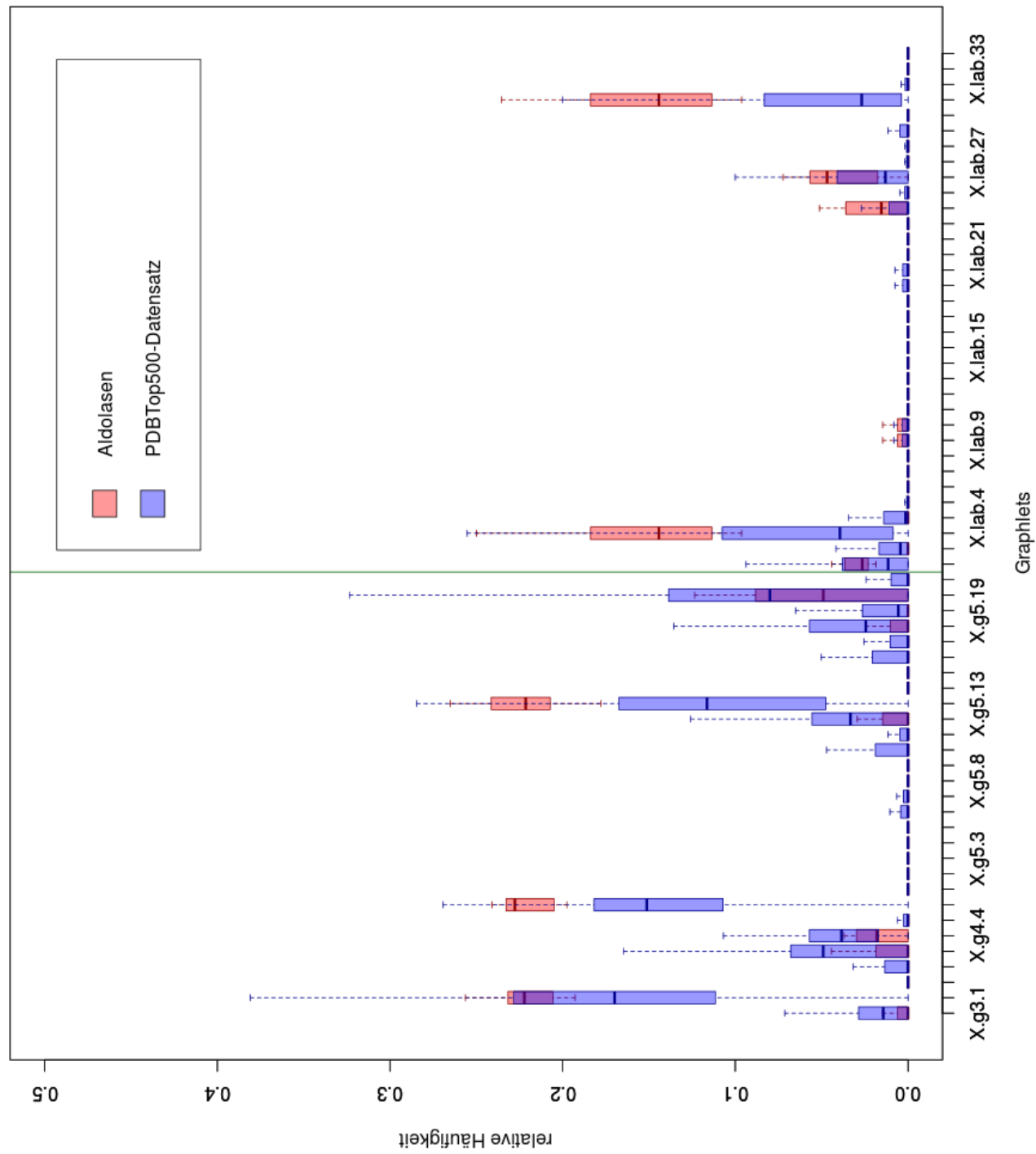


Abbildung 3.2: *Graphlet*-Verteilungen für Proteingraphen

Hier entspricht jeder Punkt auf der x-Achse einem *Graphlet* und die y-Achse zeigt die relativen Häufigkeiten an. Wieder trennt die dunkelgrüne Linie markierte und nicht markierte *Graphlets*. Solche ohne Markierungen sind links, mit Markierungen Versehene befinden sich rechts.

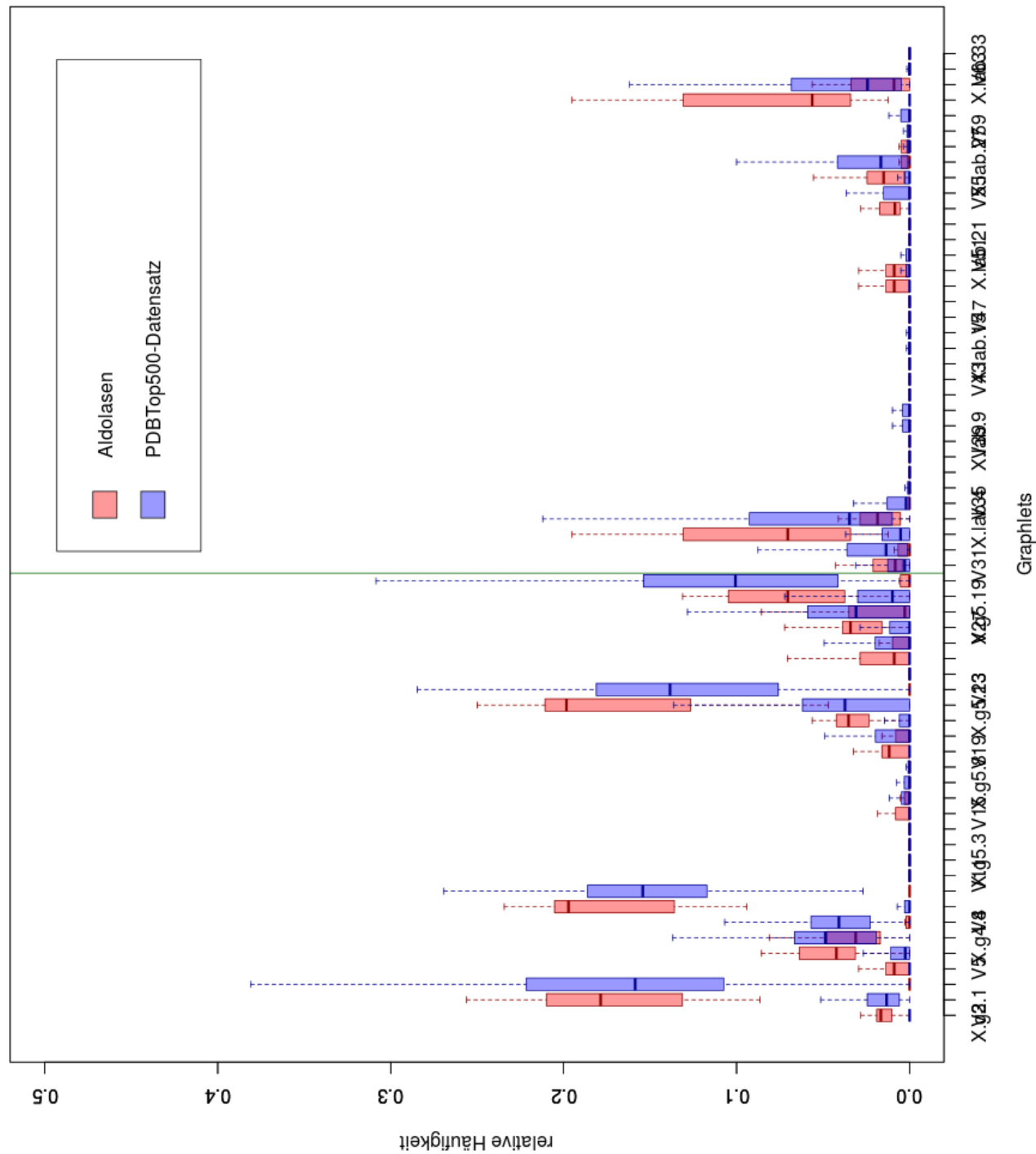


Abbildung 3.3: *Graphlet*-Verteilungen für Komplexgraphen

Auch hier entspricht jeder Punkt auf der x-Achse einem *Graphlet* und die y-Achse zeigt die relativen Häufigkeiten an. Wieder trennt die dunkelgrüne Linie markierte und nicht markierte *Graphlets*. Solche ohne Markierungen sind links, mit Markierungen verschene befinden sich rechts.

Kapitel 4

Diskussion und Ausblick

4.1 Diskussion

Die folgende Diskussion widmet sich vor allem der Frage, wieso die Ergebnisse der Ähnlichkeitsvergleiche sich so stark zwischen den jeweiligen Graphendarstellungen unterscheiden. Des weiteren wird der Zusammenhang zwischen dem modifizierten Jaccard-Index und der RGF untersucht.

4.1.1 Fallstudie

Vergleich der Graphtypen

Wie schon im Ergebnisteil dargestellt, zeigen die Vergleiche der Aminosäuregraphen den höchsten Konsens mit der Einteilung der Strukturen durch CATH und SCOPe. Eine mögliche Erklärung hierfür ist die Gestalt der Graphen. Bisher wurden *Graphlets* zur Analyse von zusammenhängenden Graphen verwendet [33], [29]. Proteingraphen und Komplexgraphen sind jedoch nicht immer zusammenhängend. Es kommt häufig vor, dass einzelne Knoten keine Verbindungen zum Rest des Graphen aufweisen. Dies kommt daher, dass *Coils* nicht modelliert werden. Die Abbildung 4.1 zeigt ein Beispiel.

Dadurch, dass dies in den zusammenhängenden *Graphlets* nicht berücksichtigt werden kann, geht Information verloren. Unabhängig von der Wahl des Ähnlichkeitsmaßes würde dieser Graph mit einem anderen Graphen, dem die beiden Helix-Knoten mit einem Grad von 0 fehlen, als gleich bewertet werden, obwohl dieser zwei SSEs weniger aufweist. Diese SSEs können jedoch biologisch von zentraler Bedeutung sein. Im Gegensatz hierzu sind die Aminosäuregraphen dieser Fallstudie zusammenhängend, weil sie alle Aminosäuren in der Polypeptidkette modellieren. Dies erklärt die höhere Genauigkeit.

Vergleich der Distanzmaße

Wirft man einen Blick in die Tabellen, sieht es zunächst so aus, als würden sich Jaccard-Index und RGF ähnlich gut eignen, um die Ähnlichkeit der *Graphlet*-Vektoren zu bewerten. Die RGF stellt eine *Distanz* zwischen zwei Vektoren dar. Dementsprechend steht ein RGF-Wert von 0 für die Gleichheit zweier Vektoren, je höher der Wert ist, desto höher ist der Abstand zwischen den beiden Vektoren. Der modifizierte Jaccard-Index, der hier Verwendung findet, zählt Elemente, die

The albe protein graph of PDB entry 3gf9, chain A [V=10, E=10].

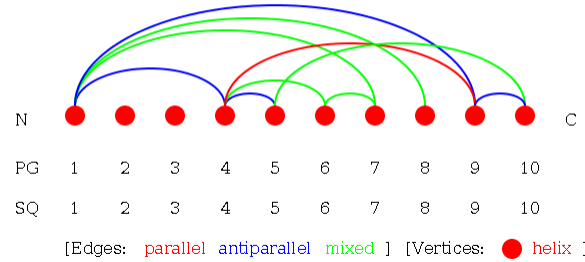


Abbildung 4.1: Proteingraph von 3GF9 Schäfer *et al.*

Tabelle 4.1: Korrelationen der RGF mit den Jaccard-Indizes für die verschiedenen Graphtypen

Datensatz	Korrelation
Aminosäuregraphen	0.6052
Proteingraphen	0.3102
Komplexgraphen	0.1256

sich um höchstens einen Faktor k unterscheiden. Ein RGF-Wert von 1 bedeutete, dass alle Elemente beider Vektoren sich höchstens um den Faktor k unterscheiden. Ein Wert von 0 bedeutet, dass alle Elemente sich um mehr als den Faktor k unterscheiden. Dementsprechend würde man erwarten, dass die *Pearson*-Korrelation von RGF und Jaccard-Index negativ ist. Die Tabelle 4.1 zeigt jedoch, dass diese für die Messungen der Fallstudie positiv ausfällt.

Eine mögliche Erklärung hierfür ist, dass die *Graphlet*-Häufigkeiten bei der RGF logarithmiert werden, dadurch werden die größten Werte zu den kleinsten und andersrum. So tragen Werte in ähnlichen Größenordnungen immer zu einer kürzeren Distanz bei, obwohl sich die tatsächlichen absoluten Häufigkeiten stark unterscheiden. Damit werden selten auftretende *Graphlets* stärker für die Distanz gewichtet als solche, die häufig auftreten.

Beim Betrachten der relativen *Graphlet*-Häufigkeiten haben wir gesehen, dass für die *Graphlets*, die besonders häufig sind, auch die Varianz dieser Häufigkeiten besonders groß ist, während gleichzeitig einige *Graphlets* fast nie auftauchen.

Diese seltenen *Graphlets* sorgen dafür, dass der Jaccard-Index wahrscheinlich nie für ein Paar aus zwei beliebigen Proteinen den Wert Null annimmt. Damit wird bereits eine gewisse Mindestähnlichkeit vorausgesetzt. Gleichzeitig kann der Jaccard-Index aber *Graphlets*, die selten auftreten nicht stärker gewichten, denn wenn ein seltenes *Graphlet* in einem Graphen einmal auftaucht und in einem anderen zweimal ist dieser Unterschied zu groß, um in eine positive Bewertung einzufließen, obwohl dies eine starke Ähnlichkeit bedeuten könnte. Diese Form von Ähnlichkeit wird in der RGF berücksichtigt.

Tabelle 4.2: Alignmenttabelle aus *PDBeFold*

In den Zellen sind die *Root-mean-square deviations* für die einzelnen Paare eingetragen.

PDB	1cgn	2id9	3i42	2w0i	1qq3
1cgn		1.357	1.497	3.554	1.757
2id9	1.357		0.405	2.982	1.651
3i42	1.497	0.405		3.066	1.866
2w0i	3.554	2.982	3.066		3.109
1qq3	1.757	1.651	1.866	3.109	

The albe protein graph of PDB entry 2id9, chain A [V=10, E=5].

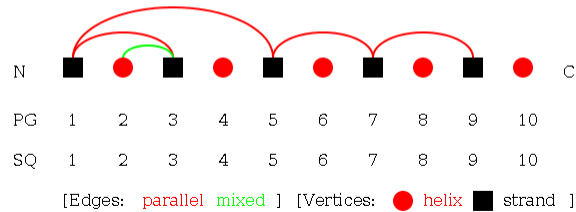


Abbildung 4.2: Proteingraph von 2id9 Schäfer et al.

Vergleich mit *PDBeFold* - die Ausreißer

Da *PDBeFold* keine 15 Proteine auf einmal alignieren kann und ein *Template*-basiertes Alignment durchführt, wurde für den Vergleich mit den *Graphlet*-Abständen eine Teilmenge der Proteine ausgewählt.

Beim Vergleich der Komplexgraphen in Tabelle 5.3 fällt auf, dass die Proteine 2ID9, 3I42 und 2W0I die geringsten Distanzen zu 1CGN haben, obwohl 1QQ3 sein Ähnlichstes sein sollte. Ein multipler Vergleich dieser Proteine mit *PDBeFold* ist in der Tabelle 4.2 zu sehen.

Sie zeigt, dass *PDBeFold* die Ähnlichkeit des α -Proteins 1CGN zu den α/β -Proteinen 2ID9 und 3I42 höher bewertet als die Ähnlichkeit zu dem α -Protein 1QQ3, obwohl dieses in der gleichen Superfamilie einzuordnen ist. Somit stimmen die Bewertungen der *Graphlet*-Vergleiche teilweise mit denen durch *PDBeFold* überein. Der einzige Unterschied in diesem Alignment ist, dass das Paar 1QQ3, 1CGN als drittähnlichstes bewertet wurde, während es beim *Graphlet*-Vergleich als viertähnlichstes eingeordnet wurde.

Die starke Ähnlichkeit der β -Proteine zu den α/β -Proteinen lässt sich erklären, wenn man die entsprechenden Graphen aus der PTGL betrachtet. Diese sind in den Abbildungen 4.2, 4.3 und 4.4 dargestellt:

In all diesen Graphen haben die meisten Knoten, die α -Helices repräsentieren, keine adjazenten Knoten. Damit fließen für keinen dieser Graphen mehr als zwei α -Helices in die *Graphlet*-Vektoren ein. Stattdessen finden sich nur die zusammenhängenden Teilgraphen dieser Graphen in den *Gra*-

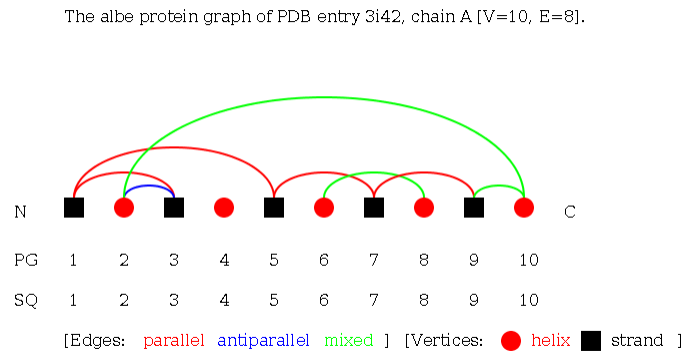


Abbildung 4.3: Proteingraph von 3i42 Schäfer *et al.*

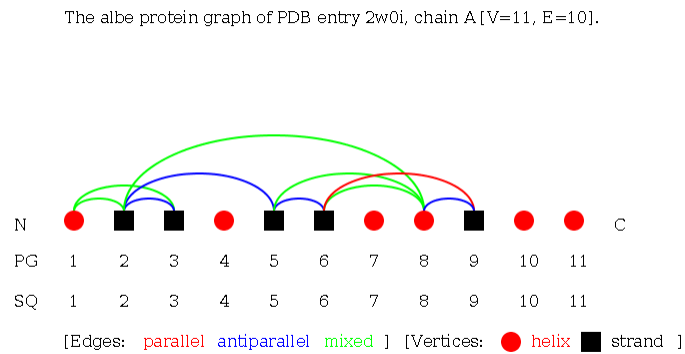


Abbildung 4.4: Proteingraph von 2w0i Schäfer *et al.*

phlet-Vektoren wieder, und diese Teilgraphen bestehen fast nur aus Knoten, die β -Faltblätter repräsentieren.

4.1.2 PDBTop500-Datensatz und Aldolasen

Der Vergleich der Proteine aus dem PDBTop500-Datensatz mit den Aldolasen hat gezeigt, dass sich die Verteilungen der *Graphlets* für einige *Graphlets* so stark unterscheiden, dass eine klare Unterscheidung der beiden anhand ihrer Verteilungen möglich zu sein scheint.

Bei den Aminosäuregraphen scheint diese Unterscheidung dabei am schlechtesten möglich zu sein, da sich die Verteilungen für nicht markierte *Graphlets* durch die Abwesenheit der *Graphlets* g_8 und g_{15} in den Aldolasen unterscheiden. Es ist nicht klar, ob diese *Graphlets* wirklich in den meisten anderen Proteinen vorkommen, oder diese Unterscheidung nur wegen der geringen Anzahl der Strukturen, die im Datensatz vertreten sind, zustande kommt. Bei den markierten *Graphlets* unterscheiden sich die Verteilungen am stärksten für jene, die die räumliche Nähe von SSEs und Liganden bzw. Liganden untereinander repräsentieren. Auch hier ist es fraglich, ob dieses Merkmal zur Unterscheidung von Aldolasen von allen anderen Superfamilien wirklich geeignet ist, oder ob Liganden in dem Datensatz unterrepräsentiert sind.

Es ist interessant zu sehen, dass die Unterschiede in den *Graphlet*-Verteilungen für Proteingraphen und Komplexgraphen am stärksten sind, obwohl diese im multiplen Vergleich der Fallstudie die geringeren Zusammenhänge mit der strukturellen Einteilung durch CATH zeigten. Die Ursache hierfür können die verwendeten Metriken sein. Die hohen Unterschiede der *Graphlet*-Häufigkeiten der Komplexgraphen im Gegensatz zu den Proteingraphen sind wahrscheinlich auf die Kontakte der Polypeptidketten untereinander zurückzuführen.

Die *Graphlet*-Vektoren scheinen sich stark zu verändern, wenn die Kontakte von Polypeptidketten miteinander berücksichtigt werden. Dies wirft die Frage auf, ob *Graphlet*-Vektoren geeignet sind, um eine Domäne in einem Komplex zu finden.

4.2 Ausblick

In der Diskussion wurde gezeigt, dass sich *Graphlets* zumindest bedingt eignen, um die Ähnlichkeit von Proteinstrukturtopologien zu bewerten. Die folgenden Abschnitte widmen sich der Frage, wie die Berechnung der *Graphlets* beschleunigt werden könnte. Weiterhin wird beschrieben, inwiefern sich die Analyse mittels *Graphlets* verbessern ließe. Dies kann entweder durch Veränderungen der *Graphlet*-Vektoren oder durch neue Bewertungsschemata entstehen.

4.2.1 Optimierung der Laufzeit von graphletAnalyser

Vor allem bei der Berechnung der *Graphlets* auf Aminosäuregraphen großer Proteine besteht Verbesserungspotential. Aktuell zählt das Programm bei einem Aufruf automatisch alle *Graphlets* mit drei, vier und fünf Knoten jeweils getrennt voneinander.

Da für die Ähnlichkeitsberechnung in ihrer aktuellen Form alle *Graphlets* einbezogen werden, ist es sinnvoll diese Berechnungen in einem Algorithmus zusammenzufassen. Für Graphen mit n Knoten und einem maximalen Knotengrad d ließe sich die Laufzeit damit von $O(nd^4 + nd^3 + nd^2)$ auf $O(nd^4)$ reduzieren. Asymptotisch ist dieser Unterschied vernachlässigbar, aber in der Praxis können so redundante Berechnungen gespart werden.

Diese Zusammenfassung ließe sich dadurch bewerkstelligen, dass man die Funktionen zum Überprüfen der *Graphlets* aus den Algorithmen zum Zählen der 3- und 4-*Graphlets* in den Algorithmus zum Zählen der 5-*Graphlets* in die *for*-Schleifen der entsprechenden Tiefen einfügt.

4.2.2 Neue Bewertungsschemata

Da die *Graphlet*-Verteilungen beim Vergleich des PDBTop500-Datensatzes mit den Aldolasen nur für einige wenige *Graphlets* stark genug waren, um beide anhand der Verteilungen unterscheiden zu können, ist es ratsam ein Bewertungsschema einzuführen, das die verschiedenen *Graphlets* unterschiedlich stark gewichtet. Um die entsprechenden *Graphlets* zu identifizieren, könnte man eine Faktorenanalyse durchführen. Durch diese erhält man Gewichtungen der einzelnen *Graphlets*, die man in das neue Bewertungsschema einfließen lassen kann.

Eine Alternative wäre, die entsprechenden Gewichte anhand von bereits vorhandenen strukturellen Einteilungen zu berechnen – also *Graphlets*, die in einer Topologie oder Superfamilie besonders stark vertreten sind, stärker zu gewichten. Damit wäre allerdings nur ein Klassifikationsschema möglich, das unterscheiden kann, ob ein Eingabeprotein zu dieser strukturellen Klasse gehört oder nicht. Hierfür könnte man *Support Vector Machines* verwenden, wie sie von *Shervashidze et al.* [33] bereits zur Klassifizierung von Enzymen verwendet worden sind.

4.2.3 Weiterentwicklung des *Graphlet*-Worte-Algorithmus

Der *Graphlet*-Worte-Algorithmus ist so erweiterbar, dass Worte hinzugefügt werden können, die größere *Graphlets* repräsentieren können. Allerdings gilt dies zunächst nicht für die Stern-*Graphlets*. Da in ihnen keine Euler-Wege, die mehr als 2 Knoten enthalten, vorkommen, kann man sie nicht durch Worte repräsentieren.

Auch für die anderen *Graphlets* mit vier und fünf Knoten ist das Finden von repräsentierenden Worten nicht trivial, da es für sie deutlich mehr Euler-Wege gibt als für die *Graphlets* der Größe 3. Dementsprechend wäre eine schnelle Implementierung nur für die *Graphlets* möglich, die einfache Wege oder Kreise sind.

4.2.4 Zusammenhängende Graphen

Viele Probleme bei der Ähnlichkeitsbewertung durch *Graphlets* scheinen daher zu rühren, dass Proteingraphen und Komplexgraphen nicht immer zusammenhängend sind. Um dieses Problem in Zukunft umgehen zu können, gibt es zwei Möglichkeiten. Erstens kann man die Graphenberechnung durch PLCC so anpassen, dass ungeordnete Regionen der Polypeptidkette einbezogen werden. Dadurch vergrößert man aber wieder die Darstellung und verschlechtert deren intuitive Verständlichkeit. Die zweite Möglichkeit ist den *Graphlet*-Algorithmus so anzupassen, dass auch *Graphlets* erkannt werden, die nicht zusammenhängend sind. Dies ist zweifelsfrei möglich. Es ist jedoch fraglich, ob die zusätzlichen Dimensionen, die der *Graphlet*-Vektor dadurch erhält, wirklich für eine präzisere Unterscheidung sorgen können. Möglicherweise verschlechtert sich dadurch auch das Signal.

4.2.5 Schluss

In der Fallstudie konnten durch den Vergleich von *Graphlet*-Vektoren die meisten ähnlichen Proteine als solche erkannt werden. Die Betrachtung der Verteilungen der *Graphlets* zeigt, dass diese sich

zwischen einer strukturell diversen Superfamilie und einem Referenzdatensatz unterscheiden.

Damit scheinen *Graphlets* gut für ähnlichkeitsbasierte Anwendungen geeignet zu sein. Es gibt aber noch Verbesserungspotential bei der Genauigkeit und der Geschwindigkeit des Programms.

Es ist allerdings nicht klar, wie die Genauigkeit genau verbessert werden könnte. Da *Graphlets* nicht einfach visualisiert werden können, sind die berechneten Ähnlichkeitswerte nur verständlich, wenn man die entsprechenden Graphen betrachten kann.

Für eine genauere Einteilung von Proteinstrukturtopologien mit *Graphlet*-Vektoren wird es nötig sein, ein Ähnlichkeitsmaß zu finden, das biologisch motiviert ist, oder beim Vergleich zusätzliche Informationen zu berücksichtigen. Die Untersuchung der *Graphlet*-Verteilungen hat gezeigt, dass es einige wenige *Graphlets* sich besser zur Unterscheidung eignen als die meisten anderen.

Hierbei ist zu berücksichtigen, dass die Ähnlichkeit von Proteinen immer im Kontext der jeweiligen Fragestellung zu verstehen ist. Die Spezifität der Bindung hängt häufig von den Positionen einiger weniger Aminosäuren in der Bindungstasche eines Proteins ab. Im Gegensatz dazu ist die konservierte Domänenstruktur für Erkenntnisse über evolutionäre Beziehungen wichtiger.

Kapitel 5

Anhang

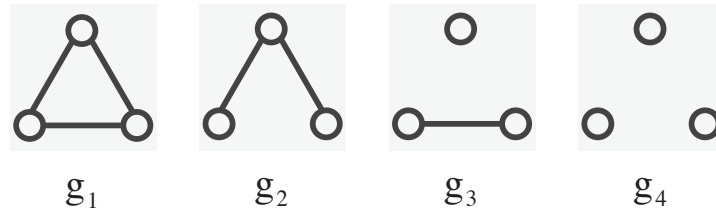


Abbildung 5.1: Graphlets der Größe 3 (*Shervashidze et al.*)

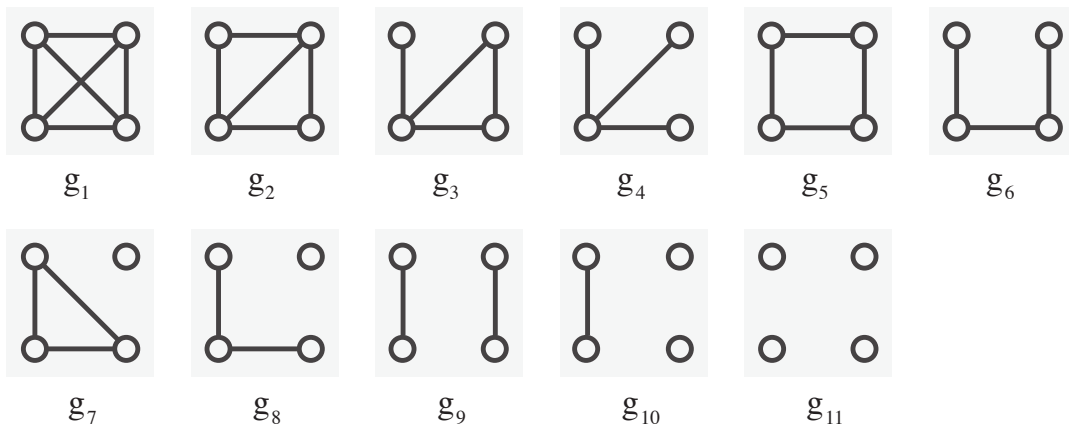


Abbildung 5.2: Graphlets der Größe 4 (*Shervashidze et al.*)

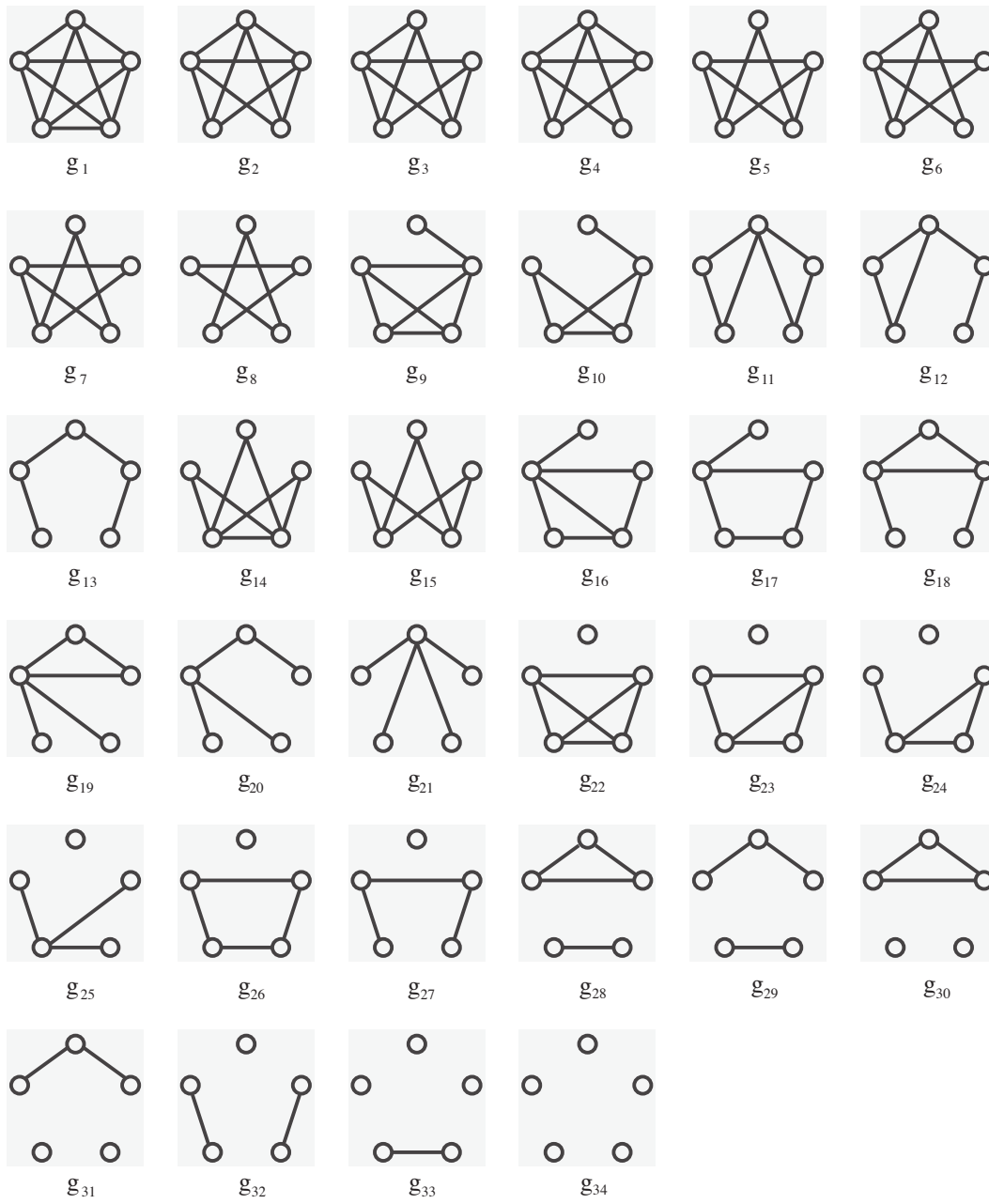


Abbildung 5.3: Graphlets der Größe 5 (*Shervashidze et al.*)

Tabelle 5.2: RGF-Distanzen der Proteingraphen

In den Zellen der Tabelle stehen die RGF-Distanzen für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 niedrigsten Distanzen grün unterlegt. Je dunkler das grün ist, desto kürzer ist die Distanz.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	0.806	0.606	0.697	3.131	1.159	2.195	2.197	1.226	1.049	1.014	0.811	1.098	0.628	0.972
1QQ3	0.806	X	0.405	1.504	1.567	1.703	0.883	0.883	1.396	1.270	0.869	0.405	0.693	0.988	1.779
1CGN	0.606	0.405	X	0.810	1.432	1.633	1.193	1.193	1.513	1.339	1.060	0.811	1.098	1.011	1.516
1HE9	0.697	1.504	0.810	X	4.492	3.253	1.386	1.386	4.357	1.291	1.135	1.099	1.386	3.599	2.506
3GF9	3.131	1.567	1.432	4.492	X	2.523	2.815	2.817	1.596	2.056	1.816	1.917	2.034	3.122	4.130
1EXS	1.159	1.703	1.633	3.253	2.523	X	1.324	1.324	2.968	0.696	0.885	1.492	1.610	3.102	3.820
1NGL	2.195	0.883	1.193	1.386	2.815	1.324	X	0.002	0.787	2.629	2.279	0.788	0.286	0.980	2.722
1QQS	2.197	0.883	1.193	1.386	2.817	1.324	0.002	X	0.787	2.631	2.281	0.788	0.285	0.980	2.722
3SLO	1.226	1.396	1.513	4.357	1.596	2.968	0.787	0.787	X	0.257	0.705	1.024	1.073	3.137	6.154
1WJX	1.049	1.270	1.339	1.291	2.056	0.696	2.629	2.631	0.257	X	1.043	0.933	0.914	0.644	2.093
5CHY	1.014	0.869	1.060	1.135	1.816	0.885	2.279	2.281	0.705	1.043	X	0.607	0.800	0.275	2.282
2ID9	0.811	0.405	0.811	1.099	1.917	1.492	0.788	0.788	1.024	0.933	0.607	X	0.692	0.783	2.890
3I42	1.098	0.693	1.098	1.386	2.034	1.610	0.286	0.285	1.073	0.914	0.800	0.692	X	1.076	3.008
1D4O	0.628	0.988	1.011	3.599	3.122	3.102	0.980	0.980	3.137	0.644	0.275	0.783	1.076	X	4.406
2W0I	0.972	1.779	1.516	2.506	4.130	3.820	2.722	2.722	6.154	2.093	2.282	2.890	3.008	4.406	X

Tabelle 5.1: RGF-Distanzen der Aminosäuregraphen

In den Zellen der Tabelle stehen die RGF-Distanzen für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 niedrigsten Distanzen grün unterlegt. Je dunkler das grün ist, desto kürzer ist die Distanz.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	1.139	1.296	6.159	5.656	10.52	11.00	12.35	12.45	11.66	7.517	6.128	6.625	7.326	8.572
1QQ3	1.139	X	1.310	5.980	5.645	9.964	10.38	11.61	11.84	11.10	6.967	5.776	6.039	7.491	8.059
1CGN	1.296	1.310	X	6.093	5.737	9.987	10.44	11.55	11.81	11.16	6.801	5.663	5.929	7.091	7.918
1HE9	6.159	5.980	6.093	X	2.289	8.032	6.813	11.03	10.87	8.405	3.974	2.445	3.452	10.93	4.807
3GF9	5.656	5.645	5.737	2.289	X	9.544	8.479	12.36	12.19	10.03	5.135	3.370	4.359	12.08	6.018
1EXS	10.52	9.964	9.987	8.032	9.544	X	4.463	4.041	3.138	2.802	4.773	7.086	5.541	7.374	4.530
1NGL	11.00	10.38	10.44	6.813	8.479	4.463	X	7.716	6.865	3.764	4.718	6.054	5.065	9.755	4.031
1QQS	12.35	11.61	11.55	11.03	12.36	4.041	7.716	X	1.674	5.340	8.148	10.20	8.612	7.226	7.404
3SLO	12.45	11.84	11.81	10.87	12.19	3.138	6.865	1.674	X	4.750	7.488	9.994	8.600	7.680	6.997
1WJX	11.66	11.10	11.16	8.405	10.03	2.802	3.764	5.340	4.750	X	5.264	7.522	5.699	8.450	4.384
5CHY	7.517	6.967	6.801	3.974	5.135	4.773	4.718	8.148	7.488	5.264	X	2.600	2.817	8.667	1.497
2ID9	6.128	5.776	5.663	2.445	3.370	7.086	6.054	10.20	9.994	7.522	2.600	X	2.447	10.24	3.657
3I42	6.625	6.039	5.929	3.452	4.359	5.541	5.065	8.612	8.600	5.699	2.817	2.447	X	9.544	2.817
1D4O	7.326	7.491	7.091	10.93	12.08	7.374	9.755	7.226	7.680	8.450	8.667	10.24	9.544	X	8.970
2W0I	8.572	8.059	7.918	4.807	6.018	4.530	4.031	7.404	6.997	4.384	1.497	3.657	2.817	8.970	X

Tabelle 5.3: RGF-Distanzen der Komplexgraphen

In den Zellen der Tabelle stehen die RGF-Distanzen für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 niedrigsten Distanzen grün unterlegt. Je dunkler das grün ist, desto kürzer ist die Distanz.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	4.405	5.056	2.271	10.75	3.567	2.309	13.66	14.10	2.128	1.933	2.165	1.877	8.791	4.422
1QQ3	4.405	X	1.906	1.714	10.03	3.860	3.457	6.851	12.70	2.587	2.431	2.557	2.269	7.907	3.002
1CGN	5.056	1.906	X	0.985	9.469	1.712	1.979	7.333	12.20	1.954	1.120	0.249	0.037	6.501	0.980
1HE9	2.271	1.714	0.985	X	5.881	3.456	1.386	4.069	5.432	1.466	1.291	1.099	1.386	9.950	2.506
3GF9	10.75	10.03	9.469	5.881	X	5.473	4.230	14.86	15.99	2.516	1.971	2.575	2.693	8.811	11.61
1EXS	3.567	3.860	1.712	3.456	5.473	X	1.290	3.217	4.391	0.611	0.662	1.459	1.576	9.191	4.226
1NGL	2.309	3.457	1.979	1.386	4.230	1.290	X	3.637	2.620	2.697	2.644	0.788	0.286	4.373	2.722
1QQS	13.66	6.851	7.333	4.069	14.86	3.217	3.637	X	12.90	5.645	4.202	3.394	3.512	16.22	4.907
3SLO	14.10	12.70	12.20	5.432	15.99	4.391	2.620	12.90	X	4.974	3.863	2.351	2.412	14.09	7.116
1WJX	2.128	2.587	1.954	1.466	2.516	0.611	2.697	5.645	4.974	X	0.360	0.961	0.965	2.225	2.042
5CHY	1.933	2.431	1.120	1.291	1.971	0.662	2.644	4.202	3.863	0.360	X	0.797	0.914	1.728	2.093
2ID9	2.165	2.557	0.249	1.099	2.575	1.459	0.788	3.394	2.351	0.961	0.797	X	0.692	1.921	2.890
3I42	1.877	2.269	0.037	1.386	2.693	1.576	0.286	3.512	2.412	0.965	0.914	0.692	X	2.039	3.008
1D4O	8.791	7.907	6.501	9.950	8.811	9.191	4.373	16.22	14.09	2.225	1.728	1.921	2.039	X	10.76
2W0I	4.422	3.002	0.980	2.506	11.61	4.226	2.722	4.907	7.116	2.042	2.093	2.890	3.008	10.76	X

Tabelle 5.4: Jaccard-Indizes der Aminosäuregraphen

In den Zellen der Tabelle stehen die Jaccard-Indizes für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 größten Indizes grün unterlegt. Je dunkler das grün ist, desto größer der Index.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	1.0	1.0	0.621	0.666	0.276	0.25	0.25	0.224	0.25	0.538	0.714	0.5	0.363	0.333
1QQ3	1.0	X	1.0	0.621	0.621	0.304	0.276	0.304	0.224	0.25	0.578	0.666	0.666	0.428	0.463
1CGN	1.0	1.0	X	0.621	0.714	0.333	0.224	0.224	0.224	0.276	0.5	0.666	0.578	0.333	0.428
1HE9	0.621	0.621	0.621	X	0.935	0.463	0.428	0.176	0.276	0.304	0.621	0.875	0.764	0.333	0.463
3GF9	0.666	0.621	0.714	0.935	X	0.276	0.333	0.2	0.2	0.224	0.621	0.764	0.621	0.25	0.463
1EXS	0.276	0.304	0.333	0.463	0.276	X	0.621	0.621	0.764	0.875	0.666	0.5	0.578	0.463	0.818
1NGL	0.25	0.276	0.224	0.428	0.333	0.621	X	0.276	0.428	0.666	0.621	0.538	0.578	0.463	0.666
1QQS	0.25	0.304	0.224	0.176	0.2	0.621	0.276	X	1.0	0.621	0.5	0.276	0.395	0.463	0.463
3SLO	0.224	0.224	0.224	0.276	0.2	0.764	0.428	1.0	X	0.538	0.463	0.333	0.395	0.463	0.5
1WJX	0.25	0.25	0.276	0.304	0.224	0.875	0.666	0.621	0.538	X	0.538	0.363	0.5	0.395	0.621
5CHY	0.538	0.578	0.5	0.621	0.621	0.666	0.621	0.5	0.463	0.538	X	0.818	0.764	0.428	1.0
2ID9	0.714	0.666	0.666	0.875	0.764	0.5	0.538	0.276	0.333	0.363	0.818	X	0.818	0.304	0.621
3I42	0.5	0.666	0.578	0.764	0.621	0.578	0.578	0.395	0.395	0.5	0.764	0.818	X	0.333	0.714
1D4O	0.363	0.428	0.333	0.333	0.25	0.463	0.463	0.463	0.463	0.395	0.428	0.304	0.333	X	0.333
2W0I	0.333	0.463	0.428	0.463	0.463	0.818	0.666	0.463	0.5	0.621	1.0	0.621	0.714	0.333	X

Tabelle 5.5: Jaccard-Indizes der Proteingraphen

In den Zellen der Tabelle stehen die Jaccard-Indizes für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 größten Indizes grün unterlegt. Je dunkler das grün ist, desto größer der Index.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	0.764	0.764	0.578	0.463	0.463	0.621	0.621	0.428	0.578	0.666	0.666	0.666	0.538	0.395
1QQ3	0.764	X	0.935	0.764	0.463	0.621	0.714	0.714	0.538	0.714	0.764	0.875	0.818	0.666	0.5
1CGN	0.764	0.935	X	0.764	0.428	0.578	0.714	0.714	0.538	0.666	0.714	0.875	0.818	0.621	0.463
1HE9	0.578	0.764	0.764	X	0.463	0.578	0.621	0.621	0.578	0.538	0.621	0.714	0.714	0.666	0.538
3GF9	0.463	0.463	0.428	0.463	X	0.428	0.463	0.463	0.463	0.463	0.538	0.428	0.463	0.5	0.333
1EXS	0.463	0.621	0.578	0.578	0.428	X	0.5	0.5	0.714	0.5	0.578	0.666	0.578	0.714	0.621
1NGL	0.621	0.714	0.714	0.621	0.463	0.5	X	1.0	0.538	0.818	0.818	0.818	0.875	0.578	0.395
1QQS	0.621	0.714	0.714	0.621	0.463	0.5	1.0	X	0.538	0.818	0.818	0.818	0.875	0.578	0.395
3SLO	0.428	0.538	0.538	0.578	0.463	0.714	0.538	0.538	X	0.538	0.5	0.578	0.621	0.578	0.463
1WJX	0.578	0.714	0.666	0.538	0.463	0.5	0.818	0.818	0.538	X	0.764	0.714	0.818	0.538	0.395
5CHY	0.666	0.764	0.714	0.621	0.538	0.578	0.818	0.818	0.5	0.764	X	0.875	0.818	0.621	0.395
2ID9	0.666	0.875	0.875	0.714	0.428	0.666	0.818	0.818	0.578	0.714	0.875	X	0.935	0.714	0.463
3I42	0.666	0.818	0.818	0.714	0.463	0.578	0.875	0.875	0.621	0.818	0.818	0.935	X	0.666	0.463
1D4O	0.538	0.666	0.621	0.666	0.5	0.714	0.578	0.578	0.578	0.538	0.621	0.714	0.666	X	0.5
2W0I	0.395	0.5	0.463	0.538	0.333	0.621	0.395	0.395	0.463	0.395	0.395	0.463	0.463	0.5	X

Tabelle 5.6: Jaccard-Indizes der Komplexgraphen

In den Zellen der Tabelle stehen die Jaccard-Indizes für die entsprechenden PDB-Dateien. Für jede Zeile (jedes Protein) sind die 4 größten Indizes grün unterlegt. Je dunkler das grün ist, desto größer der Index.

PDB-ID	1QPU	1QQ3	1CGN	1HE9	3GF9	1EXS	1NGL	1QQS	3SLO	1WJX	5CHY	2ID9	3I42	1D4O	2W0I
1QPU	X	0.428	0.395	0.428	0.153	0.304	0.395	0.132	0.111	0.363	0.395	0.428	0.428	0.276	0.276
1QQ3	0.428	X	0.666	0.621	0.2	0.395	0.5	0.2	0.090	0.5	0.5	0.538	0.538	0.333	0.428
1CGN	0.395	0.666	X	0.621	0.224	0.428	0.5	0.224	0.111	0.463	0.5	0.538	0.538	0.363	0.363
1HE9	0.428	0.621	0.621	X	0.25	0.578	0.621	0.153	0.090	0.538	0.621	0.714	0.714	0.428	0.538
3GF9	0.153	0.2	0.224	0.25	X	0.276	0.224	0.153	0.153	0.276	0.304	0.25	0.224	0.276	0.224
1EXS	0.304	0.395	0.428	0.578	0.276	X	0.5	0.2	0.153	0.578	0.578	0.666	0.578	0.5	0.621
1NGL	0.395	0.5	0.5	0.621	0.224	0.5	X	0.132	0.132	0.818	0.818	0.818	0.875	0.428	0.395
1QQS	0.132	0.2	0.224	0.153	0.153	0.2	0.132	X	0.224	0.132	0.132	0.132	0.132	0.132	0.2
3SLO	0.111	0.090	0.111	0.090	0.153	0.153	0.132	0.224	X	0.111	0.090	0.090	0.111	0.111	0.153
1WJX	0.363	0.5	0.463	0.538	0.276	0.578	0.818	0.132	0.111	X	0.935	0.714	0.818	0.5	0.395
5CHY	0.395	0.5	0.5	0.621	0.304	0.578	0.818	0.132	0.090	0.935	X	0.875	0.818	0.5	0.395
2ID9	0.428	0.538	0.538	0.714	0.25	0.666	0.818	0.132	0.090	0.714	0.875	X	0.935	0.5	0.463
3I42	0.428	0.538	0.538	0.714	0.224	0.578	0.875	0.132	0.111	0.818	0.818	0.935	X	0.428	0.463
1D4O	0.276	0.333	0.363	0.428	0.276	0.5	0.428	0.132	0.111	0.5	0.5	0.5	0.428	X	0.333
2W0I	0.276	0.428	0.363	0.538	0.224	0.621	0.395	0.2	0.153	0.395	0.395	0.463	0.463	0.333	X

Literaturverzeichnis

- [1] Fabio Arnesano, Lucia Banci, Ivano Bertini, Simone Ciofi-Baffoni, Thierry de Lumley Woodyear, Christopher M Johnson, and Paul D Barker. Structural consequences of b-to c-type heme conversion in oxidized escherichia coli cytochrome b 562. *Biochemistry*, 39(6):1499–1514, 2000.
- [2] Fabio Arnesano, Lucia Banci, Ivano Bertini, Jasmin Faraone-Mennella, Antonio Rosato, Paul D Barker, and Alan R Fersht. The solution structure of oxidized escherichia coli cytochrome b 562. *Biochemistry*, 38(27):8657–8670, 1999.
- [3] Tatiana Bakirova, Tim Schäfer, and Ina Koch. Comparison of protein topology graphs using graphlet-based methods. *GCB 2013 Göttingen-Poster Abstracts*.
- [4] Travis J Barnard, James Gumbart, Janine H Peterson, Nicholas Noinaj, Nicole C Easley, Nathalie Dautin, Adam J Kuszak, Emad Tajkhorshid, Harris D Bernstein, and Susan K Buchanan. Molecular basis for the activation of a catalytic asparagine residue in a self-cleaving bacterial autotransporter. *Journal of molecular biology*, 415(1):128–142, 2012.
- [5] BJ Bell, L Watanabe, JL Rios-Steiner, A Tulinsky, L Lebioda, and RK Arni. Structure of 2-keto-3-deoxy-6-phosphogluconate (kdpg) aldolase from pseudomonas putida. *Acta Crystallographica Section D: Biological Crystallography*, 59(8):1454–1458, 2003.
- [6] Yoshitaka Bessho, Rie Shibata, Shun-ichi Sekine, Kazutaka Murayama, Kyoko Higashijima, Chie Hori-Takemoto, Mikako Shirouzu, Seiki Kuramitsu, and Shigeyuki Yokoyama. Structural basis for functional mimicry of long-variable-arm trna by transfer-messenger rna. *Proceedings of the National Academy of Sciences*, 104(20):8293–8298, 2007.
- [7] Murray Coles, Tammo Diercks, Bernd Muehlenweg, Stefan Bartsch, Volker Zölzer, Harald Tschesche, and Horst Kessler. The solution structure and dynamics of human neutrophil gelatinase-associated lipocalin. *Journal of molecular biology*, 289(1):139–157, 1999.
- [8] UniProt Consortium et al. Uniprot: a hub for protein information. *Nucleic acids research*, page gku989, 2014.
- [9] Sayoni Das, Natalie L Dawson, and Christine A Orengo. Diversity in protein domain superfamilies. *Current opinion in genetics & development*, 35:40–49, 2015.
- [10] Robert C Davenport, Paul A Bash, Barbara A Seaton, Martin Karplus, Gregory A Petsko, and Dagmar Ringe. Structure of the triosephosphate isomerase-phosphoglycolohydroxamate

- hr/>
-
- complex: an analog of the intermediate on the reaction pathway. *Biochemistry*, 30(24):5821–5826, 1991.
- [11] AARON J Dobbs, BRYAN F Anderson, H RICK Faber, and EDWARD N Baker. Three-dimensional structure of cytochrome c’ from two alcaligenes species and the implications for four-helix bundle structures. *Acta Crystallographica Section D: Biological Crystallography*, 52(2):356–368, 1996.
- [12] J.M. Elkins, Pike A.C.W., E. Salah, P. Savitsky, J.W. Murray, A. Roos, F. Von Delft, A. Edwards, C.H. Arrowsmith, M. Wilkstrom, C. Bountra, and S. Knapp. Structure of c-terminal actin depolymerizing factor homology (adf-h) domain of human twinfilin-2. <http://www.rcsb.org/pdb/explore/explore.do?structureId=2w0i>, 2008. [Stand: 21.04.2016].
- [13] Joint Center for Structural genomics. Crystal structure of orotidine 5’-phosphate decarboxylase (tm0332) from thermotoga maritima at 2.00 Å resolution. <http://www.rcsb.org/pdb/explore/explore.do?structureId=1vqt>, 2005. [stand: 21.04.2016].
- [14] F. Forouhar, Y. Chen, J. Seetharaman, H. Janjua, L. Mao, R. Xiao, E.L. Foote, M. Maglaqui, C. Ciccocanti, L. Zhao, M.C. Baran, T.B. Acton, G.T. Montelione, J.F. Hunt, L. Tong, and Northeast Structural Genomics Consortium. Crystal structure of thiamine phosphate pyrophosphorylase (bt_0647) from bacteroides thetaiotaomicron. northeast structural genomics consortium target btr268. <http://www.rcsb.org/pdb/explore/explore.do?structureId=3ceu>, 2008. [Stand: 21.04.2016].
- [15] Naomi K Fox, Steven E Brenner, and John-Marc Chandonia. Scope: Structural classification of proteinextended, integrating scop and astral data and classification of new structures. *Nucleic acids research*, 42(D1):D304–D309, 2014.
- [16] David H Goetz, Sirkku T Willie, Roger S Armen, Tomas Bratt, Niels Borregaard, and Roland K Strong. Ligand preference inferred from the structure of neutrophil gelatinase associated lipocalin. *Biochemistry*, 39(8):1935–1941, 2000.
- [17] Martina Henn-Sax, Ralf Thoma, Steffen Schmidt, Michael Hennig, Kasper Kirschner, and Reinhard Sterner. Two ($\beta\alpha$)₀ 8-barrel enzymes of histidine and tryptophan biosynthesis have similar reaction mechanisms and common strategies for protecting their labile substrates. *Biochemistry*, 41(40):12032–12042, 2002.
- [18] Michael Hennig, Reinhard Sterner, Kasper Kirschner, and Johan N Jansonius. Crystal structure at 2.0 Å resolution of phosphoribosyl anthranilate isomerase from the hyperthermophile thermotoga maritima: Possible determinants of protein stability, #. *Biochemistry*, 36(20):6009–6016, 1997.
- [19] Flip J Hoedemaeker, Ronald W Visschers, Arno C Alting, ME Kuil, JP Abrahams, et al. A novel ph-dependent dimerization motif in β -lactoglobulin from pig (sus scrofa). *Acta Crystallographica Section D: Biological Crystallography*, 58(3):480–486, 2002.
- [20] Liisa Holm and Chris Sander. Protein structure comparison by alignment of distance matrices. *Journal of molecular biology*, 233(1):123–138, 1993.
- [21] Björn H Junker and Falk Schreiber. *Analysis of biological networks*, volume 2. John Wiley & Sons, 2011.

-
-
- [22] Richard M Karp. *Reducibility among combinatorial problems*. Springer, 1972.
- [23] E Krissinel and K Henrick. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. *Acta Crystallographica Section D: Biological Crystallography*, 60(12):2256–2268, 2004.
- [24] Simon C Lovell, Ian W Davis, W Bryan Arendall, Paul IW de Bakker, J Michael Word, Michael G Prisant, Jane S Richardson, and David C Richardson. Structure validation by C α geometry: ϕ , ψ and $c\beta$ deviation. *Proteins: Structure, Function, and Bioinformatics*, 50(3):437–450, 2003.
- [25] Patrick May, Annika Kreuchwig, Thomas Steinke, and Ina Koch. PTGL: a database for secondary structure-based protein topologies. *Nucleic acids research*, 38(suppl 1):D326–D330, 2010.
- [26] Kenneth McAdams, Eric S Casper, R Matthew Haas, Bernard D Santarsiero, Aimee L Egger, Andrew Mesecar, and Christopher J Halkides. The structures of t87i phosphono-chey and t87i/y106w phosphono-chey help to explain their binding affinities to the flm and cheZ peptides. *Archives of biochemistry and biophysics*, 479(2):105–113, 2008.
- [27] G Sridhar Prasad, Vandana Sridhar, Mutsuo Yamaguchi, Youssef Hatefi, and C David Stout. Crystal structure of transhydrogenase domain iii at 1.2 Å resolution. *Nature Structural & Molecular Biology*, 6(12):1126–1131, 1999.
- [28] Natasa Pržulj, Derek G Corneil, and Igor Jurisica. Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, 2004.
- [29] Natasa Pržulj, Derek G Corneil, and Igor Jurisica. Efficient estimation of graphlet frequency distributions in protein–protein interaction networks. *Bioinformatics*, 22(8):974–980, 2006.
- [30] U.A. Ramagopal, C. Morano, Burley S.K., and Almo S.C. Structure of the response regulator receiver domain (cheY-like) from *Methylobacillus flagellatus*. <http://www.rcsb.org/pdb/explore/explore.do?structureId=3i42>, 2009. [Stand: 21.04.2016].
- [31] Tim Schäfer, Patrick May, and Ina Koch. Computation and Visualization of Protein Topology Graphs Including Ligand Information. In Sebastian Böcker, Franziska Hufsky, Kerstin Scheubert, Jana Schleicher, and Stefan Schuster, editors, *German Conference on Bioinformatics 2012*, volume 26 of *OpenAccess Series in Informatics (OASIs)*, pages 108–118, Dagstuhl, Germany, 2012. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [32] Y Shen, Y Tong, W Tempel, Y Li, C.H. Arrowsmith, A.M. Edwards, C Bountra, J Weigelt, A Bochkarev, and H Park. Crystal structure of human intersectin 2 rhogef domain. <http://www.rcsb.org/pdb/explore/explore.do?structureId=3gf9>, 2009. [Stand: 21.04.2016].
- [33] Nino Shervashidze, Tobias Petri, Kurt Mehlhorn, Karsten M Borgwardt, and Svn Vishwanathan. Efficient graphlet kernels for large graph comparison. In *International conference on artificial intelligence and statistics*, pages 488–495, 2009.
- [34] Ilya N Shindyalov and Philip E Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein engineering*, 11(9):739–747, 1998.

-
-
- [35] Ian Sillitoe, Tony E Lewis, Alison Cuff, Sayoni Das, Paul Ashford, Natalie L Dawson, Nicholas Furnham, Roman A Laskowski, David Lee, Jonathan G Lees, et al. CATH: comprehensive structural and functional annotations for genome sequences. *Nucleic acids research*, page gku947, 2014.
 - [36] Junichiro Taka, Kyoko Ogasahara, Jeyaraman Jeyakanthan, Naoki Kunishima, Chizu Kuroishi, Mitsuaki Sugahara, Shigeyuki Yokoyama, and Katsuhide Yutani. Stabilization due to dimer formation of phosphoribosyl anthranilate isomerase from thermus thermophilus hb8: X-ray analysis and dsc experiments. *Journal of biochemistry*, 137(5):569–578, 2005.
 - [37] Ning Wu and Emil F Pai. Crystal structures of inhibitor complexes reveal an alternate binding mode in orotidine-5-monophosphate decarboxylase. *Journal of Biological Chemistry*, 277(31):28080–28087, 2002.
 - [38] Martin Würtele, Louis Renault, Joseph T Barbieri, Alfred Wittinghofer, and Eva Wolf. Structure of the exos gtpase activating domain. *FEBS letters*, 491(1):26–29, 2001.
 - [39] Yuzhen Ye and Adam Godzik. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics*, 19(suppl 2):ii246–ii255, 2003.
 - [40] Yang Zhang and Jeffrey Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic acids research*, 33(7):2302–2309, 2005.
 - [41] Xiangyang Zhu, Joseph Rebello, Philip Matsumura, and Karl Volz. Crystal structures of chey mutants y106w and t87i/y106w chey activation correlates with movement of residue 106. *Journal of Biological Chemistry*, 272(8):5000–5006, 1997.