Ein Alpha-Shape-basiertes Protein-Docking-Verfahren

Inaugural-Dissertation
zur
Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Universität zu Köln

vorgelegt von
Niels Lange
aus Leverkusen

Köln

2009

Berichterstatter:	Prof. Dr. R. Schrader Prof. Dr. U. Lang
Tag der mündlichen Prüfung:	14.01.2010

Abstract

Proteins are the building block of life. Therefore the understanding of protein interactions is of great interest. Because experimental methods to examine protein complexes are often quiet time-consuming there is a desire for computer-aided methods for complex prediction. Calculating the complex that is formed by two structures is called *protein-protein docking*. Input data are the experimentally determined atom coordinates of the single structures.

There are several docking methods that can predict acceptable complexes for some cases. But the problem is not finally solved. That is the reason for developing a new approach to protein-docking in this thesis. Contrary to most common methods no grid-based representation of the proteins is used. Grids provide easy mathematics, but also tie the problem to a rigid structure. That makes handling of protein flexibility, which occur upon complex formation, more difficult.

The new method uses *alpha shapes* to represent the protein structures. They extend the concept of the convex hull and define a surface which resolution can be adjusted by a scale parameter (α). By incrementally increasing the level-of-detail pockets on the protein surface can be detected. This is used in the algorithm to generate geometrically fitting complexes. How many are produced can be varied by choosing (α). Furthermore it was shown that a scoring-function can be realized within the alpha shape, which allows a fast check of the geometric correlation of generated complexes. The extensibility of the approach was demonstrated by implementing an amino acid specific score.

The quality of the generated complexes was tested by running the algorithm against a dataset of test cases. The results lie in the range of other methods. They can be improved by further optimizing the parameters of the algorithm.

The runtime for generating and scoring a few hundred thousand complexes is less than five minutes.

Kurzzusammenfassung

Proteine sind die Bausteine des Lebens. Das Verständnis von Interaktionen zwischen Proteinen ist daher von großem Interesse. Da die experimentelle Untersuchung von Proteinkomplexen äußerst zeitintensiv ist, besteht der Wunsch nach rechnergestützten Verfahren zur Vorhersage solcher Anordnungen. Die Berechnung des Komplexes aus zwei Proteinen wird in der Bioinformatik als *Protein-Protein-Docking* bezeichnet. Eingabedaten sind die experimentell bestimmten Atomkoordinaten der Einzelstrukturen.

Es existieren mehrere Docking-Verfahren, die bereits passable Vorhersagen für eine Reihe von Komplexen erzielen können. Abschließend gelöst ist die Aufgabenstellung aber noch nicht. Aus diesem Grund ist im Rahmen dieser Arbeit ein neuer Ansatz zur Lösung des Docking-Problems entwickelt worden. Im Gegensatz zu den meisten gängigen Verfahren wird dabei nicht auf eine Gitterdarstellung der Proteine zurückgegriffen. Diese erlaubt zwar eine mathematisch einfache Handhabung des Problems, fasst dieses aber gleichzeitig in ein starres Raster. Die Behandlung von Protein-Verformungen, die bei der Komplex-Bildung auftreten, wird dadurch erschwert.

Die in dem neuen Verfahren verwendete Repräsentation der Protein-Strukturen basiert auf *Alpha-Shapes*. Diese erweitern den Begriff der konvexen Hülle und definieren eine Oberfläche, deren Auflösung mittels eines Skalenparameters (α) eingestellt werden kann. Durch eine schrittweise Erhöhung des Detaillierungsgrades lassen sich Taschen auf der Proteinoberfläche detektieren. Dies wird in dem hier entwickelten Algorithmus dazu verwendet, um gezielt Proteinkomplexe zu generieren, die lokal geometrisch zueinander passen. Die Zahl der erzeugten Komplexe lässt sich dabei durch α steuern. Des Weiteren wurde gezeigt, dass sich mit Hilfe der Alpha-Shape eine Bewertungsfunktion realisieren lässt, die eine zügige

Beurteilung von künstlich erzeugten Komplexen bezüglich ihrer geometrischen Korrelation erlaubt. Die Flexibilität des Ansatzes wurde beispielhaft durch die Erweiterung der Bewertung um Aminosäuren-spezifische Gewichte unter Beweis gestellt.

Zur Beurteilung der Qualität der produzierten Lösungen wurde der Algorithmus mit einem Testdatensatz überprüft. Die Ergebnisse liegen in der Größenordnung anderer Verfahren, lassen sich aber noch deutlich steigern, da die verwendeten Parameter noch nicht vollständig optimiert sind. Die Laufzeiten zur Generierung und Beurteilung von einigen hunderttausend Komplexen liegt im Bereich weniger Minuten.

Inhaltsverzeichnis

Kapitel 1

Einleitung

„Jeder dumme Junge kann einen Käfer zertreten. Aber alle Professoren der Welt können keinen herstellen.“
Arthur Schopenhauer

Biologisches Leben auf unserem Planeten basiert auf dem Zusammenspiel von Proteinen. Diese bilden nicht nur die Struktur der Zellen, aus denen sich die Lebewesen zusammensetzen. Sie sind auch für deren Funktion verantwortlich, in dem sie z.B. Stoffe transportieren, chemische Reaktionen katalysieren oder Signalstoffe verarbeiten. Das Verständnis des Zusammenspiels verschiedener Proteine untereinander trägt somit unmittelbar zum Verständnis der Funktionsweise lebender Organismen bei.

Die Wichtigkeit von Proteinen für das Leben spiegelt sich auch in der Namesgebung selbst wieder. Der Begriff *Protein* wurde 1838 von Jöns Jakob Berzelius von dem griechischen Wort *proteuo* („ich nehme den ersten Platz ein“) abgeleitet.

Durch die Verbesserung der experimentellen Methoden zur Bestimmung von Proteinstrukturen existiert inzwischen eine breite Datenbasis, die es erlaubt computergestützte Analysen dieser Strukturen durchzuführen. Interessanter als die Untersuchung einzelner Strukturen ist aber die Berechnung von möglichen Komplexen aus mehreren Proteinen. Das ist insbesondere bei der Entwicklung von Medikamenten relevant. Das Vorhersagen von Proteinkomplexen kann aufwendige Experimente überflüssig machen, bzw. Hinweise darauf liefern, bei welchen Substanzen es sich eventuell lohnt, in einem Experiment genauer hinzuschauen.

Krankheiten werden in den meisten Fällen durch körperfremde Erregern oder dem falschen Zusammenspiel körpereigener Moleküle ausgelöst. Dabei kann man im Wesentlichen zwei Fälle unterscheiden. Entweder spielt ein körpereigenes Molekül im Krankheitsprozess eine negative Rolle, und sollte daher von einem Medikament in seiner Funktion gehemmt werden, oder das Gegenteil ist der Fall, d.h. eine Erhöhung der Aktivität eines Moleküls kann eine Krankheit lindern. In jedem Fall ist man daran interessiert, herauszufinden an welcher Stelle eines Proteins ein Wirkstoff ansetzen könnte und wie er dafür beschaffen sein müsste.

Die Angriffspunkte der meisten Arzneimittel sind Enzyme oder Rezeptoren. Enzyme führen praktisch alle chemischen Reaktionen durch, die im Körper ablaufen. Ohne sie könnte unser Körper nichts verdauen, keine neuen Substanzen aufbauen, umformen oder wieder abbauen. Jedes Enzym ist dabei auf eine ganz bestimmte Reaktionen spezialisiert. Das Leberenzym CSE wirkt beispielsweise mit einer Umwandlungsreaktion an der Bildung von Cholesterin mit. Es ist das Ziel für bestimmte Medikamente, die den Cholesterinspiegel im Blut senken können, wenn er zu hoch ist [97].

Rezeptoren sind die Empfangsantennen der Zellen für Hormone und ähnliche Botenstoffe, die von anderen Zellen zu ihnen gelangen. Die Beta-Rezeptoren sind beispielsweise Empfangsantennen für Adrenalinmoleküle, die den Herzschlag beschleunigen. Sie sind zugleich das Ziel für Betablocker, die dafür sorgen können, dass sich ein schwaches Herz nicht überanstrengt, indem sie es vor den Adrenalinmolekülen abschirmen [39].

1.1 Protein-Docking

Die Komplexbildung von einem Protein mit einem anderen Molekül wird als *Docking* bezeichnet. Sind beide Bindungspartner Proteine einer gewissen Komplexität, spricht man vom *Protein-Protein-Docking*. Handelt es sich bei einem der beteiligten Moleküle um ein kleines, das auch nicht zwangsläufig ein Protein sein muss, so bezeichnet man den Vorgang auch als *Protein-Ligand-Docking*.

In dieser Arbeit geht es um die Entwicklung eines neuen Protein-Protein Docking-Algorithmus, d.h. die Entwicklung einer Software, die anhand von gegebenen

dreidimensionalen Strukturdaten mögliche Proteinkomplexe berechnet. Zur Veranschaulichung der Problemstellung kann folgendes Bild herangezogen werden.

Protein-Docking ist ein dreidimensionales Puzzle, welches durch zwei Dinge erschwert wird. Zum einen herrschen Kräfte zwischen den Puzzleteilen, d.h. sie können sich gegenseitig anziehen oder abstoßen. Zum anderen müssen die Puzzelteile noch mehr oder weniger verformt werden, damit sie ineinander passen. Die Teile weisen also eine gewisse Flexibilität auf, die dafür sorgt, dass es keine eindeutige Lösung des Puzzles gibt. Stattdessen gibt es viele mögliche Strukturen, die aus den Teilen gebildet werden können, und es muss entschieden werden, welche davon der tatsächlich in der Natur vorkommenden, welche als *nativ* bezeichnet wird, am nächsten kommt. Das Kernstück eines Docking-Algorithmus bildet daher eine *Bewertungsfunktion*, die den verschiedenen Lösungsvorschlägen einen Wert zuordnet und so eine Rangfolge der Lösungen bestimmt. Ziel dabei ist es, dass nahe native Strukturen eine hohe Bewertung erhalten.

Aus dem bisher Gesagten ergibt sich die Notwendigkeit für eine weitere wichtige Komponente einer Docking-Software: es braucht einen Generator möglicher Proteinkomplexe, die dann bewertet werden können. Der Generator kann prinzipiell auf zwei Arten realisiert werden. Entweder man erzeugt weitestgehend alle möglichen Lösungen, d.h. alle Translationen und Rotationen des einen Proteins in Bezug zu dem anderen. Oder man beschränkt sich auf die Erzeugung von Strukturen, die gewissen Randbedingungen genügen, um so die Zahl der Lösungen, die mittels der Bewertungsfunktion beurteilt werden müssen, zu verringern. Die geringere Anzahl zu bewertender Lösungen wird durch einen erhöhten Rechenaufwand bei der Überprüfung der Randbedingungen erkauft. Zudem besteht die Gefahr, dass manche nahe nativen Strukturen gar nicht erzeugt werden, weil sie nicht in das Raster des Generators passen. Diese Strukturen können somit auch nicht vom Docking-Algorithmus als Lösungen gefunden werden, weil sie im Bewertungsschritt nicht berücksichtigt werden. Die Kunst besteht also darin, möglichst wenige Strukturen zu erzeugen, aber gleichzeitig viele, die der nativen ähnlich sind.

Eine weitere Unterteilung computergestützter Docking-Verfahren erfolgt bezüglich des Grades der Automation. Man unterscheidet dabei vollständig automatische Ansätze von denen, die in einem oder mehreren Schritten eine manuelle Einga-

be erfordern. Es gibt beispielsweise Programme, die eine Vorgabe benötigen, in welcher Region auf der Proteinoberfläche das Interface zu suchen ist, oder welche Bereiche gegeneinander beweglich sind [83, 84]. Im weiteren Verlauf werden ausschließlich vollautomatische Algorithmen behandelt.

Existierende Algorithmen

Im Jahr 2004 betitelten Vajda und Camacho einen Artikel zum Stand des Protein-Protein-Dockings mit der Frage *„Ist das Glas halbvoll oder halbleer?“* [94] und fassten ihre Analyse wie folgt zusammen. *Enzyme-Inhibitor* Komplexe können mit annehmbarer Genauigkeit bestimmt werden, *Antibody-Antigen* Paare sind schon weniger zuverlässig zu ermitteln und kleine *Signal-Komplexe* im Allgemeinen schwer vorherzusagen. Komplexe mit großen Interface-Regionen und signifikanten Konformationsänderungen beim Docking gingen zu dieser Zeit über die Möglichkeiten der verfügbaren Algorithmenen hinaus. Auch fünf Jahre später schließen Vajda und Kozakov einen Artikel mit der Feststellung, dass die Behandlung größerer Verformungen des Backbones bei der Komplexbildung ein ungelöstes Problem des Protein-Dockings darstellt [95].

Nichtsdestoweniger kann eine stetige Verbesserung der Docking-Verfahren festgestellt werden, wie David W. Ritchie in [82] betont. Einen wesentlich Anteil daran hat die Kombination verschiedener Ansätze innerhalb eines Docking-Prozesses. Beispielsweise werden neben geometrischen Merkmalen auch physiko-chemische Faktoren in die Berechnung einbezogen und darüber hinaus statistische Informationen verwendet, die anhand von Komplex-Datenbanken ermittelt wurden. Neben einer Verbesserung der benutzten Algorithmen ermöglicht die gesteigerte vorhandene Rechenleistung auch eine Verfeinerung der darin verwendeten Modelle. So wird es zum Beispiel möglich, das Lösungsmittel in dem sich die Proteine befinden und das den Docking-Prozess beeinflusst, explizit zu modellieren. Aktuelle Entwicklungen im Hardware-Bereich, wie Many-Core CPUs und programmierbare Grafikprozessoren, in Kombination mit verbesserten Techniken zur Programmierung paralleler Berechnungen werden in naher Zukunft noch weitere Möglichkeiten bieten.

Eine Aufstellung aktuell im Einsatz befindlicher Programme findet sich in [95] und [101]. Gemeinsam haben alle Verfahren, dass sie in mehreren Stufen ablau-

fen. Dabei wird zunächst eine große Anzahl von Komplexen erzeugt und bewertet. Anschließend erfolgt eine genauere Untersuchung der Strukturen, die gemäß der ersten Bewertungsfunktion am besten abgeschnitten haben. Die Verfahren unterscheiden sich vor allem in den Faktoren, die bei der weiteren Bewertung benutzt werden. Auffällig ist, dass ein überwiegender Teil der Programme die anfängliche geometrische Bewertung mit Hilfe einer *Fast Fourier Transformation* (FFT) der Gitterdarstellung der Proteine bewerkstelligen. Dieses Vorgehen wird im zweiten Kapitel etwas genauer vorgestellt, bildet aber nicht die Grundlage des hier entwickelten Verfahrens.

1.2 Ansatz dieser Arbeit

Zu Beginn der Arbeit stand die Überlegung, welche Voraussetzungen von einer Protein-Docking-Software erfüllt werden müssen, damit darin leicht die während der Komplexbildung auftretenden Verformungen modelliert werden können. Daraus folgte die Entscheidung für die Verwendung einer auf der Triangulation der Proteine basierenden Repräsentation und gegen die Benutzung einer Gitterdarstellung bei den Berechnungen. Die Suche nach einer Möglichkeit zur Bestimmung der Proteinoberflächen führte zu den von Edelsbrunner eingeführten Alpha-Shapes [30]. Diese erwiesen sich als besonders vielseitig und es zeigte sich, dass man darauf aufsetzend ein komplettes Protein-Docking-Verfahren entwickeln kann.

Eine weitere Grundsatzentscheidung betraf die Vorgehensweise bei der Erzeugung potentieller Proteinanordnungen. Damit die Anzahl der zu bewertenden Komplexe in einem beherrschbaren Rahmen bleibt, werden diese gezielt nach bestimmten Kriterien erzeugt und nicht, wie es z.B. bei den gitterbasierten Ansätzen üblich ist, einfach möglichst viele Anordnungen produziert. Die Idee dabei ist, dass auf diese Weise mehr Rechenzeit in die Bewertung der einzelnen Komplexe investiert werden kann und damit bessere Lösungen gefunden werden können.

Ziel der Arbeit war es, die generelle Nutzbarkeit von Alpha-Shapes innerhalb eines Protein-Protein-Docking-Verfahrens zu untersuchen. Dazu wurde eine Vielzahl der dabei vorhandenen Möglichkeiten in einem Programm namens *Alpha-Dock* realisiert und anhand eines Testdatensatzes [69] die grundsätzliche Eignung

des Ansatzes überprüft. In der bisherigen Version des Programms werden die Proteine als starre Körper behandelt *(rigid body docking)*. Der verwendete Ansatz erlaubt es aber, in der weiteren Entwicklung die explizite Berechnung von Proteinverformungen zu berücksichtigen.

Die Implementierung erfolgte ohne existierende Codebasis. Da ein besonderer Wert auf die Erweiterbarkeit und die Anpassungsfähigkeit der entstandenen Software gelegt wurde, sind nur frei zugängliche und lizenzfrei zu verwendende Bibliotheken bei der Programmierung zum Einsatz gekommen. Auch der fertige Quelltext von *AlphaDock* wird zukünftig unter der Open Source Lizenz GPL [4] zur Verfügung gestellt werden.

1.3 Gliederung der Ausarbeitung

Im nächsten Kapitel werden zunächst die biologischen Grundlagen der Proteinsynthese und -klassifikation zusammengefasst und, ebenfalls in aller Kürze, die Verfahren zur Bestimmung von Proteinstrukturdaten erläutert. Die bereits angesprochenen Schritte eines Docking-Verfahrens werden etwas ausführlicher beleuchtet und die Hintergründe der in dieser Arbeit verwendeten Alpha-Shapes vorgestellt.

Das dritte Kapitel behandelt das entwickelte Verfahren im Detail. Die einzelnen Schritte, angefangen von der internen Repräsentation der Proteine mittels Alpha-Shapes, über die Extraktion charakteristischer Merkmale auf deren Oberflächen, bis hin zur Erzeugung und Bewertung von Komplexen, werden eingehend untersucht.

Im vierten Kapitel werden die mit *AlphaDock* berechneten Ergebnisse aufgeführt. Begonnen wird mit den im Vorfeld getätigten Analysen der korrekt gedockten Komplexe des Testdatensatzes. Daraus werden die zu verwendenden Parameter abgeleitet und erste Schätzungen für deren Wertebereiche bestimmt. Die weiteren Resultate sind unter Verwendung dieser von Hand optimierten Werte entstanden.

Kapitel fünf fasst die Arbeit zusammen und gibt einen Ausblick auf zukünftige Entwicklungsmöglichkeiten des Alpha-Shape-basierten Protein-Dockings. Im An-

hang finden sich noch einige Details zum Programmaufbau und den darin vorkommenden Parametern. Auch die Bedienung der Software wird dort kurz erklärt.

Kapitel 2

Grundlagen

2.1 Biologie der Proteine

2.1.1 Proteinsynthese

Die Herstellung von Proteinen in Lebewesen, welche auch als *Proteinbiosynthese* bezeichnet wird, geschieht in folgenden Schritten. Zunächst wird der das entsprechende Protein codierende Abschnitt der Desoxyribonukleinsäure (DNA) gelesen und in ein mRNA-Molekül transkribiert. Dabei werden die Nukleinbasen der DNA (A, G, C, T) in die Nukleinbasen der RNA (A, G, C, U) umgeschrieben. Die Abfolge der Basen der mRNA wird anschließend in eine Sequenz von Aminosäuren übersetzt. Dieser Schritt wird *Translation* genannt und passiert in der Zelle an den Ribosomen. Jeweils drei aufeinander folgende Basen bilden dabei ein *Codon*, das für eine bestimmte Aminosäure steht [50, 89]. Die Zuordnung erfolgt nach einem bei allen Lebewesen nahezu gleichen Schema, welches in Abbildung 2.1 veranschaulicht wird. Die so erzeugte Aminosäurenkette faltet sich schließlich zum fertigen Protein, welches seine Funktion erst durch diese räumliche Anordnung erhält.

Durch den genetischen Code werden 20 Aminosäuren codiert. Deren Strukturformeln zeigt Abbildung 2.2. Grundsätzlich besteht eine Aminosäure aus mindestens einer Carboxylgruppe (–COOH) und mindestens einer Aminogruppe (–NH_2). Die verschiedenen Aminosäuren unterscheiden sich durch ihre jeweiligen Seitenketten.

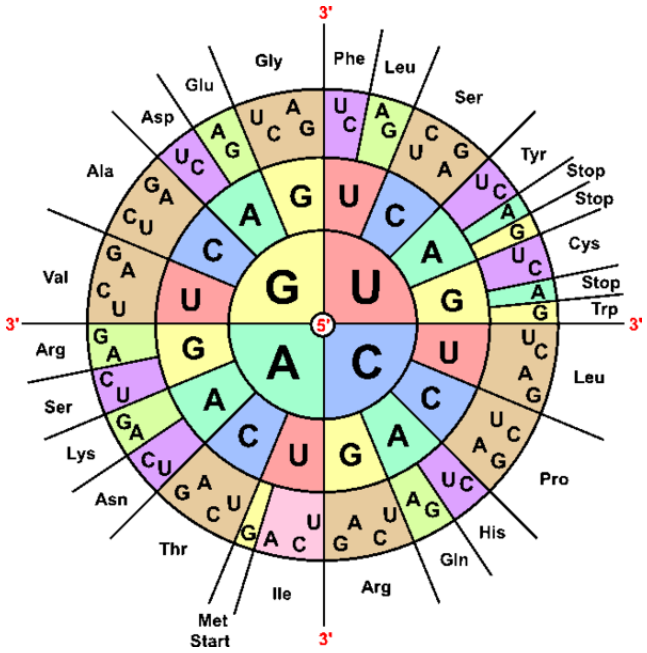


Abbildung 2.1: Veranschaulichung des genetischen Codes. Die zu einem Codon gehörende Aminosäure wird gefunden, indem man die entsprechenden Kreissegmente von innen (5') nach außen (3') durchläuft. Das Codon CGA steht z.B. für Arginin (Arg) [103].

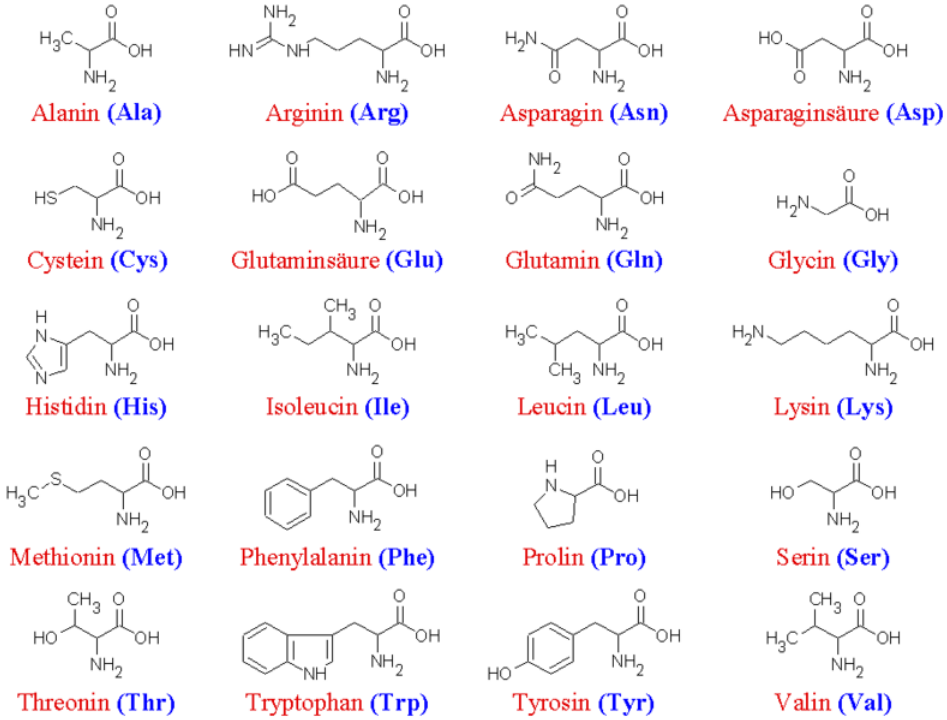


Abbildung 2.2: Strukturformeln der 20 kanonischen Aminosäuren [102].

Tabelle 2.1: Chemische Eigenschaften der 20 kanonischen Aminosäuren [55, 91].

Aminosäure	**Abkürzung**		**Polarität**	**Acidität/ Basizität**	**Hydrophobizität**
Alanine	Ala	A	unpolar	neutral	1.8
Arginine	Arg	R	polar	basisch	-4.5
Asparagine	Asn	N	polar	neutral	-3.5
Aspartic acid	Asp	D	polar	sauer	-3.5
Cysteine	Cys	C	polar	neutral	2.5
Glutamic acid	Glu	E	polar	sauer	-3.5
Glutamine	Gln	Q	polar	neutral	-3.5
Glycine	Gly	G	unpolar	neutral	-0.4
Histidine	His	H	polar	basisch	-3.2
Isoleucine	Ile	I	unpolar	neutral	4.5
Leucine	Leu	L	unpolar	neutral	3.8
Lysine	Lys	K	polar	basisch	-3.9
Methionine	Met	M	unpolar	neutral	1.9
Phenylalanine	Phe	F	unpolar	neutral	2.8
Proline	Pro	P	unpolar	neutral	-1.6
Serine	Ser	S	polar	neutral	-0.8
Threonine	Thr	T	polar	neutral	-0.7
Tryptophan	Trp	W	unpolar	neutral	-0.9
Tyrosine	Tyr	Y	polar	neutral	-1.3
Valine	Val	V	unpolar	neutral	4.2

Diese bezeichnet man auch als Rest, sie sind verantwortlich für die unterschiedlichen chemischen Eigenschaften der Aminosäuren, welche bei der Bildung von Proteinen und Proteinkomplexen eine entscheidende Rolle spielen. Tabelle 2.1 fasst einige davon für die 20 Standardaminosäuren zusammen [55, 91].

2.1.2 Proteinkomplexe

Klassifikation

Protein-Protein-Komplexe können mittels einer Vielzahl von Kriterien unterschieden werden. Dazu zählen unter anderem die Funktion, die Lebensdauer oder die Art der Bindung. Bei der Entwicklung von Docking-Verfahren ist eine Ein-

teilung von Proteinkomplexen in Klassen hilfreich, da sie es ermöglicht spezielle Parametersätze für die verschiedenen Typen zu verwenden, wodurch die Qualität der berechneten Komplexe gesteigert werden kann [41].

Eine häufig verwendete Einteilung der Komplexe bezüglich ihrer Funktion unterscheidet die Gruppen Enzym/Inhibitor oder Enzym/Substrat (E), Antibody/Antigen (A), Antigen/Bound Antibody (AB) und Sonstige (O) [69]. Unterteilt man die Komplexe nach der Bindungsart, so differenziert man zwischen solchen, die aus identischen Proteinen gebildet werden (*Homooligomeren*), und solchen, die aus verschiedenen Untereinheiten zusammengesetzt sind (*Heterooligomere*). Des Weiteren unterscheidet man *obligate* Komplexe, deren Bestandteile ohne den jeweiligen Bindungspartner nicht stabil sind, und *nicht obligate* Komplexe, deren Bestandteile auch unabhängig voneinander existieren können. Bezogen auf die Lebensdauer von Komplexen bezeichnet man diese entweder als *permanent* oder *transient*. Wobei letzteres bedeutet, dass ein Komplex nur für kurze Zeit existiert [73].

Komplexbildung

An der Bildung von Proteinkomplexen sind eine Reihe unterschiedlicher Wechselwirkungen beteiligt. Allen voran sind hier Elektrostatik, van-der-Waals-Kräfte, die Bildung von Wasserstoffbrückenbindungen und der hydrophobe Effekt zu nennen. Letzterer bezeichnet die entropiegetriebene Energieabsenkung, welche durch die Zusammenlagerung von unpolaren Molekülen im polaren Medium zustande kommt [80, 17]. Dieser Effekt wird von einigen Autoren als energetischer Hauptbeitrag der Protein-Interaktion gesehen [47]. Es zeigt sich, dass bei obligaten Komplexen die Hydrophobizität im Interface stärker ausgeprägt ist als bei nicht obligaten [48].

Van-der-Waals-Kräfte habe eine sehr geringe Reichweite, sie wirken zwischen allen benachbarten Atomen, also auch zwischen Protein und Lösungsmittel. Da aber das Interface gedockter Proteine häufig eng gepackt ist, treten dort zwischen Protein und Protein mehr van-der-Waals-Kontakte auf als zwischen Protein und Lösungsmittel im ungedockten Fall. Dadurch können van-der-Waals-Kräfte einen Beitrag zur Bindungsenergie von Proteinkomplexen liefern [47].

Atome mit unterschiedlicher Elektronegativität können kovalente oder ionische Bindungen miteinander eingehen. Ionenbindungen werden auch Salzbrücken genannt. Diese sind für Proteininteraktionen nicht zwingend erforderlich, können sie aber deutlich stabilisieren [47]. Eine Wasserstoffbrücke liefern einen wesentlich geringeren Beitrag zur Bindungsenergie als eine Salzbrücke oder kovalente Bindung, dafür finden sich zum Teil deutlich mehr davon im Interface. Pro 170 Å^2 Bindungsfläche existiert im Mittel eine Wasserstoffbrücke, was eine Gesamtzahl von 10 ± 5 für eine durchschnittliche Gesamtkontaktfläche ergibt [93].

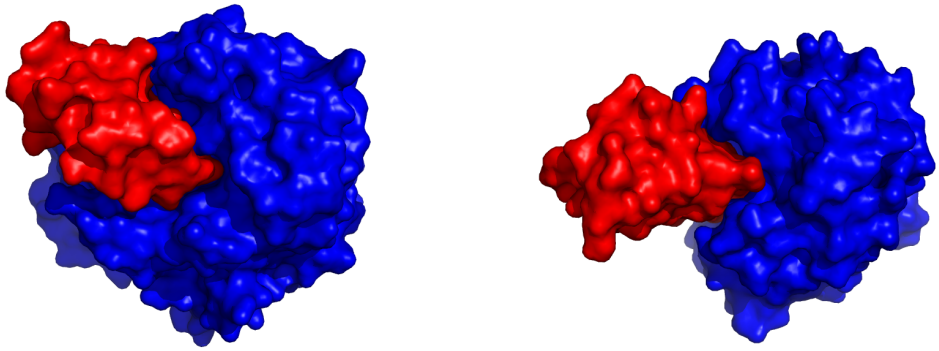
Komplementarität

Die Grundannahme aller geometriebasierten Docking-Methoden besteht darin, dass an der Bindungsstelle der beiden Proteine eine mehr oder weniger stark ausgeprägte Komplementarität vorhanden ist. Diese Passgenauigkeit der Kontaktflächen ist bei Homodimeren, Enzym-Inhibitor-Komplexen, und stabilen Heterokomplexen stärker ausgeprägt, als es bei Antigen-Antikörper-Komplexe und instabile Heterokomplexe der Fall ist [93].

Abbildung 2.3 zeigt die Oberflächen des Rezeptors und des Liganden von zwei gebundenen Komplexe (1BVN, 1EAW) in unterschiedlichen Farben. Man erkennt so leicht, dass die Suche nach komplementären Regionen der Bindungspartner plausibel ist. Praktisch alle zur Zeit im Einsatz befindlichen Docking-Programme verwenden daher die geometrische Korrelation als ein Bewertungskriterium, worauf im Abschnitt 2.2.4 noch näher eingegangen wird. Die Suche nach geometrisch zueinander passenden Stellen auf der Oberfläche von Proteinen ist auch die Grundlage für den in dieser Arbeit entwickelten Ansatz.

2.2 Aufbau eines Protein-Docking-Algorithmus

Dieser Abschnitt erläutert die einzelnen Bestandteile, die im Allgemeinen in einem Docking-Algorithmus implementiert werden. Zunächst wird dabei auf das Format der Eingabedaten eingegangen und schrittweise deren Verarbeitung beschrieben. Zur Veranschaulichung möglicher Realisationen der einzelnen Schritte



(a) 1BVN **(b)** 1EAW

Abbildung 2.3: Verdeutlichung der geometrischen Komplementarität zweier gebundener Proteinkomplexe.

werden zwei etablierte Verfahren herangezogen. Zum einen der mittels Fourier-Transformation arbeitende Algorithmus, der ursprünglich von Katchalski-Katzir vorgeschlagen wurde [49]. Und zum anderen der von Wolfson und Nussinov entwickelte Geometric-Hashing Ansatz [23]. Der in dieser Arbeit entwickelte Algorithmus wird im Kapitel 3 vorgestellt und dort in Bezug zu den hier angeführten Verfahren gesetzt.

2.2.1 Proteinstrukturdaten

Zur Entwicklung einer Protein-Docking-Software werden Proteinstrukturdaten in maschinenlesbarer Form benötigt. Diese finden sich als Textdateien in der Protein Data Bank (PDB) [11], einer Datenbank mit 3D-Strukturdaten von zur Zeit rund 56000 Proteinen und 2400 Nukleinsäuren (Oktober 2009). Die Strukturen wurden hauptsächlich mittels Röntgenstrukturanalyse (X-Ray) oder Kernresonanzspektroskopie (NMR) bestimmt. Der Anteil der mit Röntgenbeugung bestimmten Strukturen überwiegt deutlich und beträgt z.B. bei den Proteinen über 87%.

Röntgenstrukturanalyse (X-Ray)

Die Bestimmung von Proteinstrukturen mittels Röntgenstrahlen [65] macht sich zu Nutze, dass deren Wellenlänge im Bereich von einigen Ångström liegt und

damit in der Größenordnung der Atomabstände in einem Kristall. Dadurch sind die Atome für das Röntgenlicht ein dreidimensionales Beugungsgitter. Aus dem auftretenden Beugungsmuster kann die Kristallstruktur berechnet werden. Dabei kann die Geometrie der Elementarzelle des Kristallgitters vollständig anhand der Winkel abgeleitet werden, unter denen die Beugungsmaxima auftreten. Da aber nur die Intensität der gebeugten Strahlung gemessen werden kann und die Information über die Phase verloren geht, kann die genaue Position der Atome im Kristall nicht direkt bestimmt werden. Dieser Umstand wird als *Phasenproblem* bezeichnete, welches zunächst gelöst werden muss.

Bei kleineren Molekülen lässt sich die Phasen iterativ ab initio, d.h. ohne Zusatzinformation, berechnen. Bei Proteinen verwendet man zumeist andere Verfahren, zum Beispiel Multiple Isomorphous Replacement (MIR) oder Multiwavelength Anomalous Diffraction (MAD). Einen Überblick über die verschiedenen Methoden und weitere Referenzen finden sich in [90].

Hat man das Phasenproblem gelöst, kann mittels Fourier-Transformation aus dem Beugungsbild eine Karte der Elektronendichte des Proteins berechnen werden. In diese Karte wird iterativ die Aminosäurekette eingepasst, was sich mit abnehmender Auflösung der Elektronendichte schwieriger gestaltet. Man beginnt mit den C_α-Atomen des Peptidrückgrats und passt anschließend die Positionen der Seitenketten an.

Die größte Schwierigkeit bei der Strukturbestimmung mittels Röntgenbeugung besteht, neben dem Lösen des Phasenproblems, in der Züchtung der benötigten Kristalle [19]. Die Bestimmung der Struktur innerhalb eines Kristalls bringt zudem folgende Probleme mit sich. Es ist praktisch unmöglich, zeitaufgelöste Aufnahmen zu erstellen und so die Funktionsweise eines Proteins zu beobachten. Zudem ist nicht ohne Weiteres klar, ob die Struktur des Proteins in einem Kristall auch der Struktur in der natürlichen Umgebung, z.B. innerhalb einer Zelle, entspricht.

Kernspinresonanzspektroskopie (NMR)

Atomkerne bestehen aus Protonen und Neutronen, welche die quantenmechanische Eigenschaft *Spin* besitzen. Der Gesamtspin eines Kerns kann ganz- oder

halbzahlige Werte annehmen oder Null sein. Letzteres ist der Fall, wenn sich der Atomkern aus der gleichen Anzahl Protonen und Neutronen zusammensetzt. Hat der Kern einen nicht verschwindenden Spin, so besitzt er ein magnetisches Dipolmoment. Dieses besitzt, gemäß der Quantenmechanik, für einen Kern mit dem Spin I genau $2I+1$ mögliche Orientierungen. Ein Kern mit Spin 1/2 besitzt daher genau zwei, die ohne äußeres Magnetfeld energetisch zusammenfallen. Legt man ein Magnetfeld an, so spalten die Energieniveaus auf. Aus der Thermodynamik ergibt sich, dass das niedrigere Niveau eine geringfügig höhere Besetztungswahrscheinlichkeit besitzt.

Ein Übergang zwischen den beiden Niveaus entspricht anschaulich einer Änderung der Orientierung des magnetischen Dipolmoments. Erreicht wird ein Übergang durch die Zufuhr eines passenden Energiequants in Form von elektromagnetischer Strahlung. Diese Resonanzfrequenz des Übergangs, auch *Larmorfrequenz* genannt, ist abhängig von der Stärke des äußeren Magnetfeldes und der Art des Atomkerns. Sie wird bei der Kernspinresonanzspektroskopie gemessen. Die genannten Abhängigkeiten erlauben jedoch keine Rückschlüsse über die Struktur eines spektroskopierten Moleküls. Die NMR-Spektroskopie wäre so nur eine aufwendige Möglichkeit, die enthaltenen Elemente zu bestimmen.

Die gemessene Frequenz hängt jedoch davon ab, welche Atome sich in der Umgebung des jeweiligen Kerns befinden, da die Elektronenwolke der umliegenden Kerne zu einer mehr oder weniger großen Abschirmung des äußeren Magnetfeldes führt. D.h. es existiert ein lokales Magnetfeld, das eine andere Aufspaltung der Energieniveaus ergibt als es das äußere Magnetfeld tun würde. Dieser Effekt wird *chemische Verschiebung* genannt und führt zu einer für die Umgebung des Kerns charakteristischen Larmorfrequenz, die daher Informationen über die Molekülstruktur enthält.

Zur konkreten Strukturaufklärung haben sich einige (mehrdimensionale) Verfahren etabliert, auf die aber nicht näher eingegangen wird. Es gibt z.B. die Correlation Spectroscopy (COSY), die Diffusion Ordered Spectroscopy (DOSY) oder die Nuclear Overhauser Enhancement Spectroscopy (NOESY). Details entnimmt man der jeweiligen Literatur [37, 46, 81].

Der Vorteil der NMR-Spektroskopie besteht darin, dass Proteinstrukturen in Lösung bestimmt werden können und es nicht nötig ist, einen Proteinkristall zu

züchten. Zum anderen ist es mit ihr möglich, zeitaufgelöste Aufnahmen zu erstellen und so Proteinbewegungen zu untersuchen. Ein Nachteil besteht in der maximalen Größe der zu bestimmenden Strukturen. Mit aktuellen Verfahren sind aber inzwischen Proteine bis zu einer Größe von gut 400 Aminosäuren zu bewältigen [14].

PDB-Dateien

Proteinstrukturdaten werden in den oben erwähnten PDB-Dateien gespeichert. Das sind einfache Textdateien, die im Wesentlichen die Koordinaten der Atome eines Proteins enthalten. Jeweils eine Zeile beinhaltet die Informationen über ein Atom. Die ersten 10 Atome des Serin Protease/Inhibitor Komplexes beschreiben zum Beispiel folgende Zeilen aus der Datei 1CGI.PDB:

```
ATOM   1  N   CYS E 1 11.377  21.513  11.770  1.00  7.18 N
ATOM   2  CA  CYS E 1 12.025  21.956  13.016  1.00  5.40 C
ATOM   3  C   CYS E 1 11.406  21.350  14.300  1.00  6.41 C
ATOM   4  O   CYS E 1 10.216  21.020  14.517  1.00  5.73 O
ATOM   5  CB  CYS E 1 12.168  23.454  12.852  1.00  3.26 C
ATOM   6  SG  CYS E 1 10.913  24.625  13.296  1.00  2.00 S
ATOM   7  N   GLY E 2 12.379  21.161  15.213  1.00  6.48 N
ATOM   8  CA  GLY E 2 12.408  20.728  16.555  1.00  5.36 C
ATOM   9  C   GLY E 2 11.698  19.535  17.075  1.00  5.75 C
ATOM  10  O   GLY E 2 10.898  19.506  18.027  1.00  4.47 O
```

Neben den drei Raumkoordinaten enthält jeder Eintrag auch die Kennzeichnung des Elementes (3. Spalte) und der Aminosäure (4. Spalte) des betreffenden Atoms. Der Wert in der vorletzten Spalte ist der sogenannten *Temperaturfaktor*, er ist ein Maß für die Genauigkeit der Bestimmbarkeit der Atomkoordinaten. Beweglichere Teile eines Proteins erscheinen bei einer Messung unschärfer, sie erhalten daher einen höheren Faktor als fest in die Struktur eingebundene Atome. Der Temperaturfaktor kann in Docking-Verfahren als Hinweis auf flexible Regionen innerhalb eines Proteins verwendet werden [107].

Zur Erprobung von unbound Protein-Protein-Docking-Algorithmen werden Testdaten benötigt, die neben der gedockten Komplexstruktur auch die ungedockten

Einzelstrukturen der Bindungspartner enthalten. Die Anzahl der Strukturen, für die diese Daten vorliegen, ist im Vergleich zu der Gesamtzahl der Proteine in der PDB-Datenbank sehr gering. Ein Zusammenstellung von 84 Komplexen findet man im Protein-Protein Docking Benchmark 2.4 [69]. Die Strukturen sind nach Komplexart, d.h. Antikörper-Antigen, Enzym-Inhibitor/Substrat und Sonstige Komplexe, sowie in drei Schwierigkeitsklassen unterteilt. Weitere Testdaten finden sich in dem von Heuser und Martin zusammengestellten Unbound Unbound Protein Protein Docking Dataset (UUPPDD) [40].

2.2.2 Parser und interne Darstellung

Die in den PDB-Dateien gespeicherten Proteine werden zunächst, in Abhängigkeit von der Form der weiteren Verarbeitung, in eine interne Darstellung der Docking-Software überführt. Generell wird meistens eine vereinfachte Repräsentation der Proteinoberfläche gewählt. Dies ermöglicht einerseits eine effiziente algorithmische Verarbeitung. Andererseits können durch die verringerte Auflösung der Darstellung bereits gewisse Konformationsänderungen während der Komplexbildung berücksichtigt werden. Es ist ein weit verbreiteter - in vielen Fällen der einzige - Ansatz in geometriebasierten Verfahren, die Flexibilität der Proteine dadurch zu modellieren, dass ein gewisser Überlapp der simplifizierten Proteinoberflächen erlaubt wird.

Einen Überblick verwendeter Darstellungen entnimmt man [96]. Viele Verfahren setzen auf eine Diskretisierung der Proteinstrukturen in Form eines Gitters. Die Gitterzellengröße wird abhängig von der gewünschten Genauigkeit und der vorhandenen Rechenleistung gewählt. Neben der schnellen Berechnung der Oberfläche besitzt die Methode den Vorteil, dass ein Gitter mathematisch gut zu handhaben ist. Dies machten sich zunächst Katchalski-Katzir et.al. in Form der Fast Fourier Transformation (FFT) zunutze [49]. Heute existieren zahlreiche weitere FFT-basierte Docking-Verfahren [18, 34, 53, 71, 106].

Andere Ansätze gehen von einer Repräsentation der *Solvent Accessible Surface* (SAS) aus, welche 1971 von Lee und Richards definiert wurde [58]. Ein Algorithmus zur Berechnung und Visualisierung der SAS wurde von Connolly entworfen und implementiert [20], man spricht daher auch von der *Connolly-Oberfläche*. Anschaulich entsteht diese Oberfläche, wenn eine Kugel mit dem Durchmesser eines

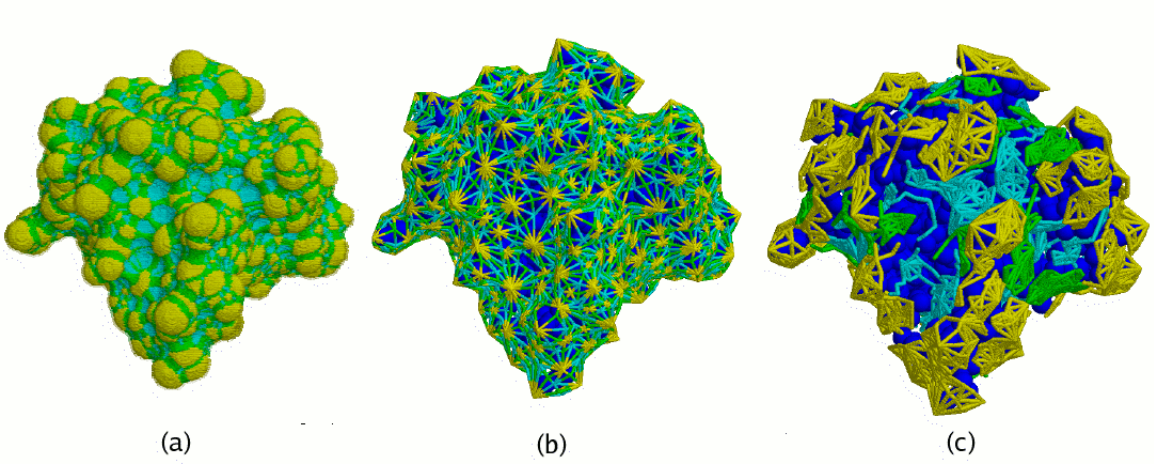


Abbildung 2.4: Vereinfachte Repräsentation der Proteinoberfläche. Ausgehend von der Solvent Accessible Surface (a) über ein Netzwerk kritischer Punkte (b) zu wenigen Surfaces Patches. Quelle: [38]

Wassermoleküls über ein Protein gerollt und deren Spur aufzeichnet wird. Ausgehend von der Connolly-Oberfläche wurden weitere Vereinfachungen entwickelt [8, 63, 64]. Abbildung 2.4 verdeutlicht das Vorgehen von Wolfson und Nussinov, von der SAS zu einer geringen Anzahl von Surface-Patches zu gelangen [38].

2.2.3 Komplex-Bildung

Der nächste Schritt eines Docking-Verfahrens besteht in der Generierung von Proteinkomplexen. Dabei unterscheidet man im Allgemeinen zwei grundsätzlich verschiedene Herangehensweisen. Entweder wird probiert, den Suchraum vollständig zu erfassen, d.h. alle möglichen Translationen und Rotationen eines Proteins in Bezug auf das andere zu berechnen. Oder man beschränkt sich auf die Erzeugung von Strukturen, die gewissen Randbedingungen genügen, um so die Zahl der anschließend zu beurteilenden Komplexe zu verringern. Die geringere Anzahl wird durch einen erhöhten Rechenaufwand bei der Überprüfung der Randbedingungen erkauft. Die gesparte Rechenzeit bei der Bewertung der Komplexe gleicht diesen Nachteil aber meistens aus. Beiden Ansätzen ist gemein, dass sie keine Vorkenntnis über die Region des Interfaces benötigen. Viele Verfahren setzen hingegen voraus, dass die Region des Interfaces bereits bekannt ist, was die Suche stark vereinfacht, hier aber nicht als Voraussetzung angenommen wird.

Gitterbasierte Ansätze behandeln den gesamten Suchraum. Sie machen sich dabei zunutze, dass mittels einer Fourier Transformation alle Translationen zu einer gegebenen Orientierung des Proteins implizit in einem Rechenschritt behandelt werden können [49]. Die Generierung von Komplexen reduziert sich dadurch auf die Berechnung verschiedener Rotationen eines Proteins, für die dann jeweils die geometrische Korrelation aller Translationen gleichzeitig bestimmt werden, was im nächsten Abschnitt genauer erklärt wird. Das Rotationssampling erfolgt in einer Schrittweite zwischen 10° und 20° [67, 106]. Wie man eine gleichmäßige nicht redundante Abdeckung des Suchraumes erhält, beschreibt Lattman in [57].

Verfahren, welche die Proteinoberfläche nicht in Form eines Gitters diskretisieren, konzentrieren sich bei der Erzeugung von Proteinkomplexen auf die Suche lokal zusammenpassender Gegensätze [8, 43]. Dazu werden charakteristische Stellen auf den Proteinen bestimmt, beispielsweise Berge und Täler, und komplementäre Paare zur Deckung gebracht. Wolfson und Nussinov berechnen die bereits erwähnten und in Abbildung 2.4 (c) gezeigten Surface-Patches und erzeugen Komplexe, indem sie bis zu zwei Patches eines Proteins mit der gleichen Anzahl kompatibler Patches des anderen überlagern [23]. Die Suche nach zueinander passenden Patches erfolgt mittels Geometric Hashing, einer von Lamdan und Wolfson zur Bilderkennung entwickelten Technik [56], die 1991 erstmals auch für das Protein-Docking verwendet wurde [75].

2.2.4 Komplex-Bewertung

Der zentrale Teil eines Docking-Algorithmus besteht in der Beurteilung der generierten Proteinkomplexe. Die dafür benutzten Bewertungsfunktionen berücksichtigen verschiedene Faktoren, angefangen bei der geometrischen Komplementarität bis hin zu physiko-chemischen Wechselwirkungen (vgl. Abschnitt 2.1.2), und gewichten diese unterschiedlich. Eine Übersicht verwendeter Ansätze enthält [38]. Die Bewertung erfolgt zumeist in mehreren Schritten, wobei von Schritt zu Schritt weniger Komplexe mit immer genaueren und somit rechenintensiveren Funktionen behandelt werden. Die einzelnen Schritte behandeln zum Teil mehrere Parameter gleichzeitig.

Geometrische Korrelation

In einer frühen Phase des Dockings wird die geometrische Korrelation bestimmt, die allgemein mit der Fläche wächst, an der die beiden Proteine einen geringen Abstand voneinander haben, sich aber nicht durchdringen. Wie die Korrelation konkret berechnet wird, hängt von der verwendeten Oberflächendarstellung ab.

Gitterbasierte Verfahren errechnen die Korrelation für einen Komplex, indem die beiden Gitter, auf welche die Proteine diskretisiert wurden, überlagert werden und die Summe über die Produkte der übereinander liegenden Gitterzellen bestimmt wird. Der Wert einer Zelle ist dabei abhängig davon, ob sich die Zelle im Inneren, am Rand oder außerhalb des dazugehörigen Proteins befindet. Bei fester Orientierung der Proteine bezüglich der beiden $(N \times N \times N)$ Gitter wird die Summe

$$C_{\alpha,\beta,\gamma} = \sum_{l=1}^{N} \sum_{m=1}^{N} \sum_{n=1}^{N} A_{l,m,n} \cdot B_{l+\alpha,m+\beta,n+\gamma} \tag{2.1}$$

für alle Translationen α, β und $\gamma \in \{0 \dots N\}$ bestimmt. Dabei gilt

$$A_{l,m,n} = \begin{cases} 1 : \text{am Rand} \\ \rho : \text{innerhalb} \\ 0 : \text{außerhalb} \end{cases} \quad B_{l,m,n} = \begin{cases} 1 : \text{am Rand} \\ \delta : \text{innerhalb} \\ 0 : \text{außerhalb} \end{cases} \quad \text{des Proteins} \tag{2.2}$$

mit $\rho < 0$ und $\delta > 0$. Diese Rechnung wird für die verschiedenen Rotationen des kleineren Proteins wiederholt. Durch das negative ρ wird ein Überlapp der Proteine bestraft. Wie der Betrag von ρ und δ und die Größe des Randes gewählt werden können, entnimmt man z.B. [106]. Dort wird auch erklärt, wie es mit Hilfe der Fouriertransformation möglich ist, die Berechnung von 2.1 zu beschleunigen.

Andere Ansätze, die kein Gitter zur Repräsentation der Proteinoberfläche verwendet, berechnen den geometrischen Score nach einem ähnlichen Muster. Es werden die Abstände zwischen den Atomen der Bindungspartner bestimmt. Für jeden Abstand innerhalb eines gewissen Intervalls wird ein positiver Beitrag zum Score hinzugezählt, zu kleine Abstände entsprechen einer Kollision und werden negativ bewertet. Abstände ab einer gewissen Größe haben keinen Einfluss auf die Bewertung. Beispiele für derartige Bewertungsfunktionen gibt es in [8, 23, 59].

Weitere Faktoren und lokale Optimierung

Neben der bloßen Geometrie fließen weitere physikalische und biologische Kriterien in die Beurteilung von Proteinkomplexen ein. Dazu zählen z.B. Hydrophobizität, Wasserstoffbrückenbindungen, Elektrostatik, Desolvationsenergie oder die Konserviertheit von Residuen im Interface.

Für eine gewisse Anzahl von Komplexen mit einem hohen Score führen manche Docking-Algorithmen eine lokale Optimierung durch. Dabei wird versucht, die Position eines Proteins, oder von einzelnen Seitenketten bzw. Atomen im Interface, so um die im vorangegangenen Schritt vorhergesagte Position zu variieren, dass eine energetisch günstigere Anordnung entsteht. Die Energie wird mit heuristischen Kraftfeldern berechnet, die eine Mittelung der physikalischen Kräfte enthalten. Die Variation der Atompositionen erfolgt entweder in Form von zufälligen Auslenkungen um die Ausgangsposition (Monte Carlo Simulation) oder über eine Energieminimierung mittels Molekulardynamik (MD) Simulation.

Flexibilität

Die bisher beschriebenen Verfahren behandeln die Proteine vor der lokalen Optimierung als starre Körper (Rigid Body) und modulieren die Flexibilität der Bindungspartner durch das Zulassen eines gewissen Überlapps der Proteinoberflächen. Dieser wird von der Bewertungsfunktion zwar bestraft, aber führt nicht zu einem Verwerfen des Komplexes. Die Beweglichkeit der Proteine wird somit implizit von der Bewertungsfunktion berücksichtigt. Das funktioniert für relativ kleine Abweichungen der gebundenen von der nativen Struktur.

Die Flexibilität der Proteine kann auch explizit berücksichtigt werden. Dies geschieht zum einen unter Verwendung von Rotamer-Bibliotheken [16, 44], welche Statistiken über Anordnungen von Seitenketten in Proteinen enthalten. Oder es wird versucht, die beweglichen Teile zu berechnen. Beispielsweise in dem sogenannte *Scharniere* (Hinges) gesucht werden, um die sich ganze Teile eines Proteins bewegen können [84]. Weitere Details zu aktuellen Ansätzen im Bereich der Behandlung von Flexibilität findet man im Review Artikel von Bonvin [13].

Welche Bewertungsansätze in aktuellen Docking-Verfahren eingesetzt werden, fasst Tabelle 2.2 zusammen. Die Angaben stammen aus einem Artikel von Vajda und Kozakov, der im Jahr 2009 erschienen ist [95].

Tabelle 2.2: Ansätze aktueller Protein-Docking-Verfahren gemäß [95].

Arbeitsgruppe	Ansatz	Programm
Weng [18, 68, 79]	FFT Suche mit detailierter Bewertungsfunktion mit Paarpotentialen (ZDOCK), Nachjustierung mit lokaler Minimierung (RDOCK) und Neubewertung mit globalem Potential (ZRANK)	ZDOCK RDOCK ZRANK
Vajda, Camacho [15, 53, 54]	FFT Suche mit detailierter Bewertungsfunktion mit Paarpotentialen (PIPER), Clustering (ClusPro), Nachjustierung mit stochastischer globaler Minimierung (SDU) und Stabilitätstests des Minimums	PIPER SDU ClusPro
Abagyan [32]	Starre Körper Pseudo-Braunsche Monte Carlo Minimierung mit gitterbasierter Energiefunktion, Monte Carlo Nachjustierung ausgesuchter Konformationen mit flexiblen Interface-Seitenketten	ICM-DISCO
Baker [36, 100]	Starre Körper Monte Carlo Minimierung mit vereinfachter Proteingeometrie und Bewertungsfunktion, Nachjustierung aller Atome von Niedrig-Energie Clustern mit iterativer Neuordnung der Seitenketten und möglicher Einstellung von Backbone-Segmenten	RosettaDock
Wolfson, Nussinov [31, 84, 85]	Geometrisches Docking durch den Abgleich lokaler Oberflächen-Merkmale und geometrisches Hashing. FlexDock behandelt Scharnierbewegungen eines Moleküls. FireDock vollzieht eine schnelle Nachjustierung mit Seitenketten-Anpassung	PatchDock FlexDock FireDock
Eisenstein [52]	FFT mit einer gewichteten Oberflächen-Komplementaritätszielfunktion, Clustern guter Lösungen, Filterung mit a priori Informationen, kleine lokale Rotationen um ausgewählte Konformationen	MolFit

2.2.5 Auswertung

Bei der Entwicklung von Docking-Algorithmen wird die Qualität der berechneten Proteinkomplexe beurteilt, indem die Abweichung der berechneten Struktur von der tatsächlich gemessenen Struktur bestimmt wird, die im Testdatensatz enthalten ist. Als Maß für die Übereinstimmung der Komplexe wird üblicherweise

der mittlere Abstand der C_α-Atome im Interface zwischen berechneter und richtiger bzw. bestmöglicher Lösung verwendet. Dieser RMSDC_αI wird im Folgenden verkürzt als *RMSD* (Root Mean Square Deviation) bezeichnet. Als Interface-Atome werden gewöhnlich all diejenigen gezählt, deren Abstand zu einem Atom im Bindungspartner kleiner als sechs Ångström ist.

Bei Verfahren, die Proteine als starre Körper behandeln, entspricht die beste Lösung einer optimalen Überlagerung der ungebundenen (unbound) und der gebundenen (bound) Strukturen. Ein solches Alignment kann mit Hilfe des Programms *CE* [88] bestimmt werden. Die nativen Strukturen des Benchmark 2.4 Datensatzes [69] sind bereits entsprechend ausgerichtet und können unmittelbar als Referenz für die Berechnung des RMSD benutzt werden.

Komplexe mit einem RMSD kleiner als fünf Ångström gelten als *nahe native* Lösungen. Es existieren Alternativen und Erweiterungen zur Beurteilung berechneter Konformationen. Beispielhaft sollen hier die Kriterien aufgeführt werden, die im Rahmen des CAPRI Experiments [45] verwendet werden. CAPRI steht für *Critical Assessment of PRedicted Interactions* und bezeichnet einen Wettbewerb, bei dem Docking-Verfahren verschiedener Arbeitsgruppen gegeneinander antreten können. Die Aufgabe dabei ist es, bisher nicht veröffentliche Komplexe aus den bekannt gegebenen Strukturen der Bindungspartner vorherzusagen.

Die Bewertung der eingereichten Lösungen umfasst folgende Punkte: die Anzahl der nativen bzw. nicht nativen Residuum-Residuum Kontakte im Verhältnis zu der Gesamtzahl der Kontakte. Den Anteil an nativen Kontakten im vorhergesagten Interface. Neben dem RMSD des Interfaces auch der RMSD des gesamten Liganden. Sowie die Rotation und Translation, die den vorhergesagten Liganden auf den nativen Liganden platzieren, wenn der vorhergesagte Rezeptor optimal auf den nativen Rezeptor gelegt wurde.

2.3 Alpha-Shapes

2.3.1 Hintergrund

Die Grundlage für den in dieser Arbeit entwickelten Docking-Algorithmus bilden die sogenannten *Alpha-Shapes*, die sowohl zur Repräsentation der Proteinoberflächen als auch zur Erzeugung und Bewertung von Proteinkomplexen verwendet werden. Der folgende Abschnitt erklärt allgemein die Grundlagen von Alpha-Shapes indem diese sowohl anschaulich als auch mathematisch beschrieben werden.

Alpha-Shapes wurden 1983 von Edelsbrunner et al. in zwei Dimensionen eingeführt [29], bereits mit dem Hinweis auf die Erweiterbarkeit des Konzeptes auf drei und mehr Dimensionen. Der dreidimensionale Fall wird von Edelsbrunner und Mücke ausführlich in [30] behandelt. Alpha-Shapes sind eine Verallgemeinerung der konvexen Hülle und erlauben eine mathematische Beschreibung der *Form einer Punktmenge*. Der Detailierungsgrad, *Level of Detail* (LOD), wird über einen Parameter α festgelegt, der eine feine bis grobe Annäherung an die Form erlaubt. Die beiden Extreme der Auflösung sind die Punktmenge selbst, für $\alpha = 0$, und die konvexe Hülle der Menge, für $\alpha = \infty$.

Eine anschauliche Vorstellung von der Entstehung und der Form von Alpha-Shapes ermöglicht folgendes Bild, das von Fischer in [33] verwendet wird. Man stelle sich einen Block aus Speiseeis vor, in dem kleine harte Stücke, z.B. aus Schokolade, enthalten sind. Aus dem Block entfernt man mit einen runden Löffel alles Eis, das man erreichen kann ohne ein Schokoladenstück zu berühren. Das können auch Teile im Inneren des Blocks sein, die man von der Oberfläche aus nicht erreichen kann. Reduziert man die festen Bestandteile des Blocks auf Punkte, entsteht das Bild eines Vorläufers der Alpha-Shape. Man erhält daraus eine Alpha-Shape, indem man alle Rundungen zwischen den Punkten durch Geraden ersetzt. Der Radius des Löffels entspricht in dieser Anschauung dem Parameter Alpha der Alpha-Shape: mit einem sehr kleinen Löffel kann alles Eis entfernt werden, es bleiben nur die festen Stücke übrig, die der Punktmenge für $\alpha \to 0$ entsprechen. Ebenso gilt die Analogie zwischen einem sehr großen Löffel und $\alpha \to \infty$. Die beschriebene Veranschaulichung der Alpha-Shape erleichtert das Verständnis der folgenden formalen Definition.

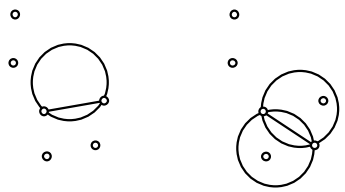


Abbildung 2.5: Alpha-offen (links) und nicht alpha-offen (rechts) [33]

Betrachtet wird eine Punktmenge $S \subset \mathbb{R}^3$. Eine *α-Kugel* bezeichnet eine offene Kugel mit Radius α, wobei $0 < \alpha < \infty$. Eine 0-Kugel ist ein Punkt, eine ∞-Kugel ein offener Halbraum. Eine α-Kugel b ist *leer*, falls $b \cap S = \emptyset$. Jede Teilmenge $T \subseteq S$ der Größe $|T| = k + 1$ mit $0 \leq k \leq 3$ definiert einen *k-Simplex* σ_T, der die *konvexe Hülle* von T ist, bezeichnet als $\text{conv}(T)$. Ein k-Simplex wird α-offen genannt, falls eine α-Kugel b existiert, für die $T = \partial b \cap S$, wobei ∂b der Rand von b ist (siehe Abbildung 2.5).

Zu einem festen α existiert ein Satz $F_{k,\alpha}$ von α-offenen k-Simplizes. Die Alpha-Shape von S ist das Polytop, dessen Rand ∂S_α sich aus allen α-offenen k-Simplizes zusammensetzt:

$$\partial S_\alpha = \{\sigma_T \mid T \cap S,\ |T| \leq 3 \text{ und } \sigma_T \ \alpha\text{-offen}\} \tag{2.3}$$

Der Rand der Alpha-Shape besteht somit aus den Dreiecken aus $F_{2,\alpha}$, den Kanten aus $F_{1,\alpha}$ und den Knoten aus $F_{0,\alpha}$ [30]. Abbilung 2.6 verdeutlicht die beschriebene Bestimmung der Alpha-Shape anhand einer zweidimensionalen Punktmenge. In diesem Fall werden anstatt von Kugeln Kreise mit dem Radius α verwendet.

Für die Berechnung der Alpha-Shape ist folgender Zusammenhang von entscheidender Bedeutung: für jeden Wert von $0 \leq \alpha \leq \infty$ wird die Alpha-Shape S_α durch einen Subkomplex der *Delaunay-Triangulation* von S bestimmt. Die Delaunay-Triangulation ist eine spezielle Triangulation, die im Allgemeinen, nicht degenerierten Fall, die konvexe Hülle einer Punktmenge $S \subset \mathbb{R}^3$ eindeutig in Tetraeder zerlegt [25]. Diese Triangulation wurde 1934 von Boris Delaunay vorgeschlagen [22]. Sie ist dual zu einem anderen Komplex, dem nach Georgy Voronoi benannten *Voronoi Diagramm* [98, 99].

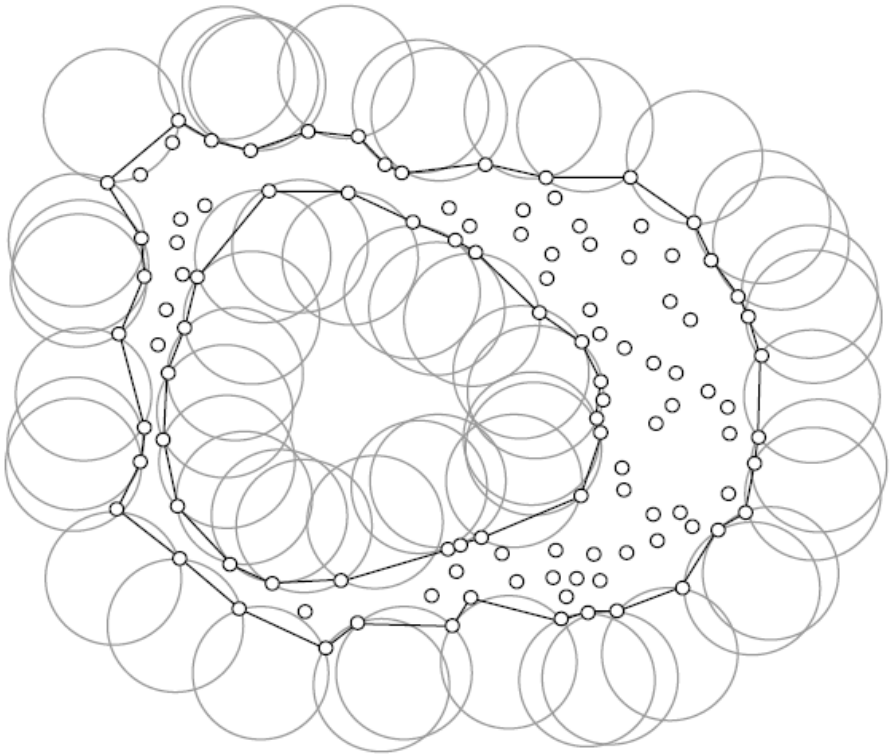


Abbildung 2.6: Beispiel für die Bestimmung der Alpha-Shape eine zweidimensionalen Punkmenge gemäß [33].

Anschaulich lässt sich mit Hilfe der Delaunay-Triangulation die Alpha-Shape auf folgende Weise bestimmen: Die Knoten des Komplexes werden als Mittelpunkte von Kugeln mit wachsendem Radius gewählt. Der Radius der Kugeln entspricht dem Parameter α. Bestandteil der Alpha-Shape für ein festes α sind alle Simplizes, d.h. alle Knoten, Kanten, Dreiecke und Tetraeder der Triangulation, die vollständig von Kugeln überdeckt sind. Zu beachten ist dabei, dass sich die Kugeln nur innerhalb der zugehörigen Voronoi-Region ausdehnen können. Die Analogie dieses Vorgehens zur oben eingeführten Definition der Alpha-Shape wird von Edelsbrunner et al. in [27] und [30] gezeigt.

Abbildung 2.7 verdeutlicht die Beziehung zwischen der Delaunay-Triangulation, dem entsprechenden Voronoi Diagramm und der resultierenden Alpha-Shape. Ausgangspunkt ist die im Abschnitt 2.2.2 beschriebene SAS. In diesem Fall wurde

eine gewichtete Triangulation benutzt, die auch zu einer gewichteten Alpha-Shape führt, welche im nächsten Abschnitt behandelt wird.

Gewichtete Alpha-Shapes

Die bisherigen Ausführungen betrachten Alpha-Shapes als eine Möglichkeit, Aussagen über die Form eine Punktmenge $S \subset \mathbb{R}^3$ zu treffen. Um Alpha-Shapes zur Beschreibung und Behandlung von Proteinen verwenden zu können, die in erster Näherung als eine Überlagerung von unterschiedlich großen Kugeln verstanden werden können, muss das Konzept der Alpha-Shape von Punkten auf Kugeln mit unterschiedlichen Radien erweitert werden. Die so entstehenden *gewichteten Alpha-Shapes* werden von Herbert Edelsbrunner in [26] beschrieben.

Eine Überleitung zum gewichteten Fall gelingt am leichtesten unter Ausnutzung des oben erwähnten Zusammenhangs zwischen dem Voronoi-Diagramm, der Delaunay-Triangulation und der Alpha-Shape. Wir betrachten zunächst noch einmal die Konstruktion des Voronoi-Diagramms einer Punktmenge. Dabei wird der Raum in *Voronoi-Regionen* unterteilt, die jeweils einen Punkt der Menge enthalten und den Teil des Raumes, zu dem der Punkt den geringen euklidischen Abstand besitzt. Die Grenzen der Regionen stehen senkrecht auf den Verbindungen benachbarter Punkte und teilen diese genau in der Mitte. Verwendet man anstelle von Punkten Kugeln mit einheitlichem Radius, erfolgt die Konstruktion des Diagramms analog. Um den Fall unterschiedlicher Radien zu behandeln, benutzt man einen *gewichteten Abstand* anstatt des euklidischen.

Betrachtet man eine Kugel mit Radius ρ um den Mittelpunkt p als einen gewichteten Punkt p mit dem Gewicht $w_p = \rho^2$, so lässt sich mittels

$$\pi_p(x) = \|p - x\|^2 - w_p \tag{2.4}$$

ein gewichteter Abstand definieren, wobei $\|p-x\|$ den euklidischen Abstand zwischen p und x bezeichnet. Eine geometrische Interpretation erkennt man anhand von Abbildung 2.8 für den Fall, dass p eine Kugel mit Radius $\sqrt{w_p}$ darstellt. In diesem Fall ist $\pi_p(x)$ die Wurzel der Länge eines Tangentensegmentes von x zur Kugel.

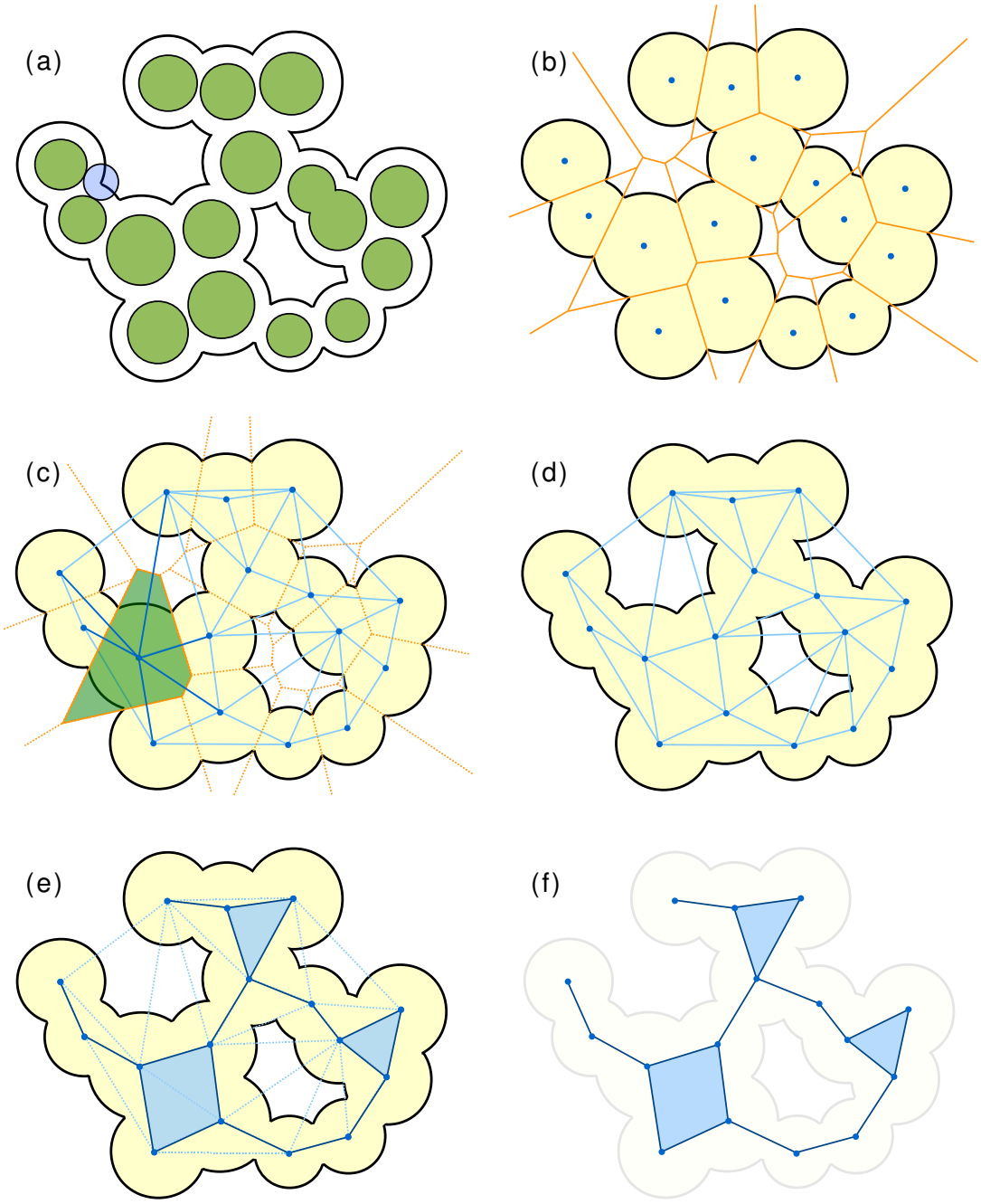


Abbildung 2.7: Zweidimensionale Veranschaulichung des Weges von der *Solvent Accessible Surface* (a) über das *Voronoi Diagramm* (b) und die *Delaunay-Triangulation* (c, d) zur gewichteten *Alpha-Shape* (e, f). Erläuterung im Text.

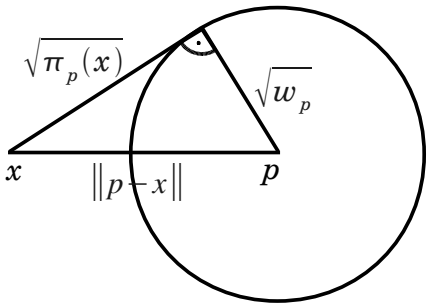


Abbildung 2.8: Geometrische Interpretation des gewichteten Abstands. Erklärung im Text.

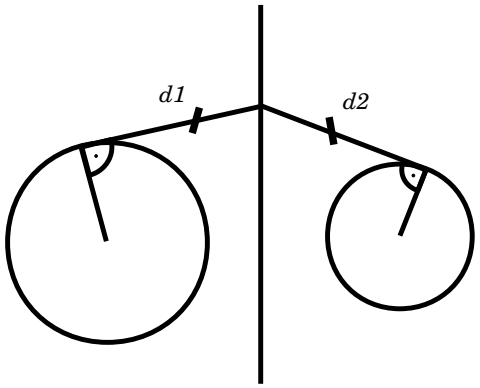


Abbildung 2.9: Bestimmung der Grenzlinie zwischen zwei Voronoi Regionen im Fall eines gewichteten Abstands. Die Linie wird so gewählt, dass die beiden Tangenten $d1$ und $d2$ den selben Abstand besitzen.

Bei der Konstruktion eines gewichteten Voronoi Diagramms bestimmt man die Grenzen der Voronoi Regionen mit Hilfe des gewichteten Abstands. Abbildung 2.9 zeigt die Entstehung der Grenzlinie im zweidimensionalen Fall. Aus dem gewichteten Voronoi Diagramm ergibt sich, analog zum ungewichteten Fall, eine Delaunay-Triangulation indem man die Mittelpunkte der Kugeln verbindet, die in aneinander angrenzenden Regionen liegen. Dieser Zusammenhang wird in der bereits angesprochenen Abbildung 2.7 deutlich.

Analog wie oben beschrieben, erhält man aus der Delaunay-Triangulation für den gewichteten Fall die zugehörige Alpha-Shape, indem man die Überdeckung der Simplizes durch Kugeln betrachtet, deren Radius über den Parameter α variiert wird. In diesem Fall können die Kugeln jedoch unterschiedliche Radien haben.

Man erkennt leicht den Zusammenhang zur Konstruktion des gewichteten Voronoi Diagramms.

Das bisher gesagte lässt sich auf die Bestimmung der Alpha-Shape eines Moleküls übertragen. Hierbei verwendet man als Anfangsradius r_0 den van-der-Waals-Radius der jeweiligen Atome und lässt diese gemäß

$$r_\alpha = \sqrt{r_0^2 + \alpha^2} \tag{2.5}$$

wachsen. Auf diese Weise bleibt das gewichtete Voronoi Diagramm, und damit der Delaunay Komplex, unverändert während α wächst. Damit r_α kleiner werden kann als r_0 muss α^2 negativ werden können. Dies ist nur möglich, wenn wir für α neben positiven reellen Zahlen auch Vielfache der imaginären Einheit $i = \sqrt{-1}$ zulassen.

2.3.2 Anwendungen

Bevor im nächsten Kapitel die in dieser Arbeit entwickelte Realisation eines Protein-Docking-Algorithmus auf der Basis von Alpha-Shapes vorgestellt wird, folgen zunächst einige Beispiele für andere Anwendungen.

Edelsbrunner et al. haben sich 1998 in zwei Artikeln ausführlich mit der Berechnung und Analyse von Molekülоberflächen mittels Alpha-Shapes befasst [61, 62]. Dabei haben sie einen Algorithmus entwickelt, der die exakte Berechnung von Flächen und Volumen von Makromolekülen erlaubt, die z.B. zur Bestimmung der Solvent Accessible Surface verwendet werden können. Das zweite wichtige Ergebnis ist eine Definition von Taschen und Höhlen in Proteinen und ein Verfahren zur Ermittelung dieser Merkmale. Letzteres wird ebenfalls in [28] und [60] behandelt. Dabei wird zudem versucht, aus den gefundenen Strukturen auf den Oberflächen der Proteine Vorhersagen über deren Funktionsweisen zu machen.

Liang et al. haben diese Algorithmen zur Behandlung der geometrischen Proteineigenschaften mit Alpha-Shapes in einer Software namens *CASTp* (Computed Atlas of Surface Topography of proteins) implementiert. Es existiert eine Webseite [24] über die von jedermann mit *CASTp* Berechnung durchgeführt werden können.

Eine weitere Arbeit von Ban et al. befasst sich mit der Möglichkeit mit Hilfe von Alpha-Shapes eine Beschreibung des Interfaces von Protein-Protein-Komplexen zu formulieren [9]. Abbildung 2.10 zeigt drei Visualisierung des so definierten Interaces für Komplexe aus Barnase und Barstar.

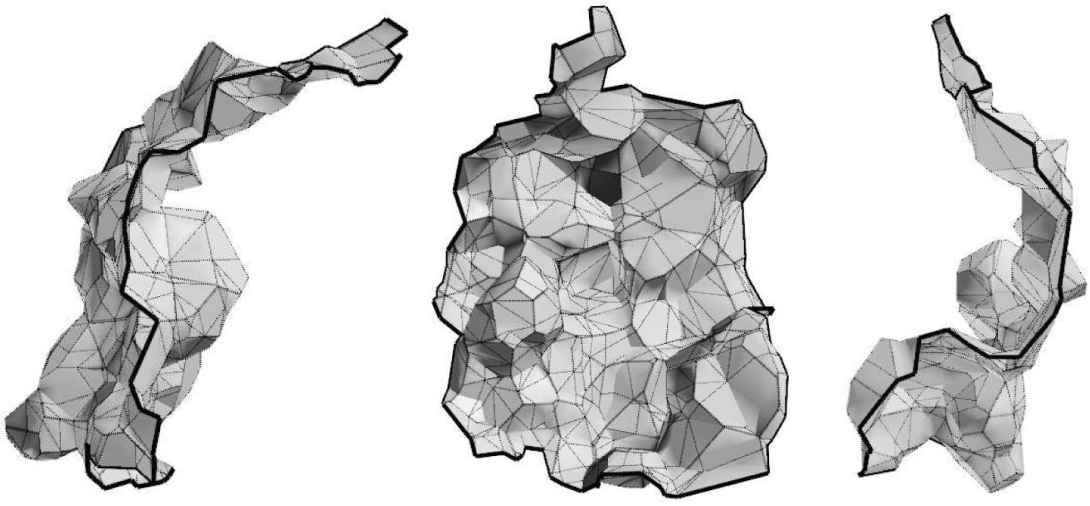


Abbildung 2.10: Drei Darstellungen des Interface zwischen Barnase und Barnstar mit der Hilfe von Alpha-Shapes. Quelle: [9].

Neben den Anwendungen von Alpha-Shapes in bioinformatischen Fragestellungen existieren zahlreiche Beispiele in anderen Gebieten [10, 51, 76, 77, 86, 92, 105]. Dabei geht es zum einen darum, die Oberfläche von einem dreidimensionalen Körper anhand von Messpunkten zu rekonstruieren, die z.B. mittels eines 3D-Scanners bestimmt worden sind [10, 77, 92, 105]. Oder es wird zum anderen versucht, die Ähnlichkeit von Formen zu bestimmen, bzw. verformte und verdrehte Abbildungen den entsprechenden Körpern zuzuordnen [51, 76, 86]. Eine aktuelle Anwendung aus dem Bereich der medizinischen Bildverarbeitung, in diesem Fall die Berechnung von Hüllen in einem bildgebenden Verfahren, findet sich in [66].

2.3.3 Grenzen und Erweiterungen

Mit klassischen Alpha-Shapes lässt sich die Oberfläche einer Punktmenge in verschiedenen Auflösungen definieren. Die genaue Form der Darstellung hängt dabei stark von der Wahl von Alpha ab. Es stellt sich daher die Aufgabe, für

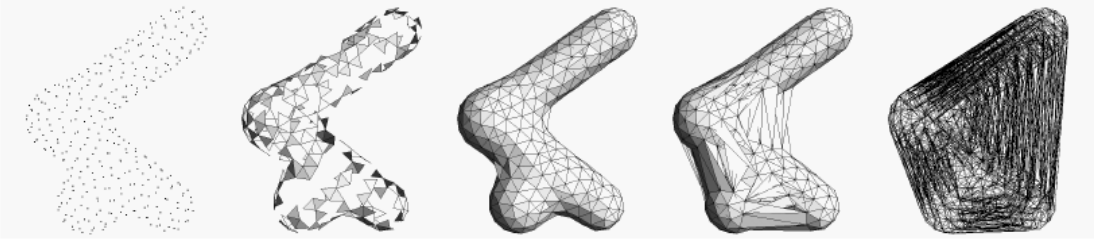


Abbildung 2.11: Alpha-Shape einer Punktmenge für fünf verschiedene Werte von Alpha. Man erkennt die Schwierigkeit der Wahl von Alpha, um die Oberfläche bestmöglich anzunähern. Quelle: [92].

die jeweilige Anwendung den besten Wert von Alpha zu bestimmen. Abbildung 2.11 verdeutlicht die Problematik anhand von fünf verschiedenen Alpha-Werten für eine Punktmenge. Die mittlere Darstellung liefert augenscheinlich die beste Annäherung an die zugrundeliegende Form. Gesucht wird eine mathematische Formulierung, die diesen Alpha-Wert von den anderen unterscheidet. In diesem Beispiel ist es prinzipiell möglich einen Wert von Alpha zu finden, der die Aufgabe löst, da die Punkte relativ gleichmäßig auf der Oberfläche verteilt sind.

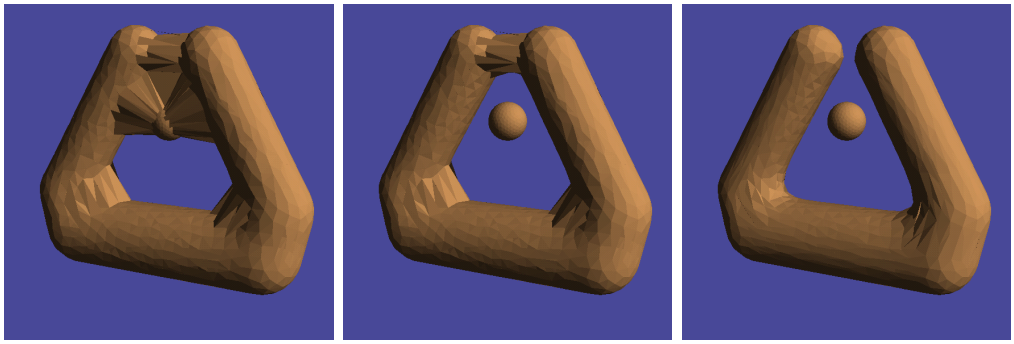


Abbildung 2.12: Beispiel für die Grenzen von klassischen Alpha-Shapes. Für die Figur läßt sich kein eindeutiges Alpha finden (links). Die von Teichmann beschriebenen Erweiterungen lösen das Problem (mittig und rechts). Quelle: [92].

In anderen Fällen, insbesondere bei der Rekonstruktion von Oberflächen aus unregelmäßig verteilten Messpunkten, ergibt sich jedoch das Problem, dass kein einzelner Wert für Alpha gefunden werden kann, der die gesuchte Form zufriedenstellend wiedergibt. Abbildung 2.12 zeigt einen solchen Fall. Eine mögliche Lösung ist die Verwendung von *anisotropic density-scaled Alpha-Shapes*, wie sie in [92] eingeführt werden. Ein weiteres Beispiel einer dichtebasierten Erweiterung

von Alpha-Shapes findet sich in [105, 35], die zudem vollständig automatisch funktioniert.

Kapitel 3

Alpha-Shape Docking

In diesem Kapitel wird gezeigt, wie es mittels der im letzten Abschnitt vorgestellten Alpha-Shapes möglich ist, die einzelnen Schritte eines Docking-Algorithmus zu realisieren. Alle Bestandteile, die in 2.2 allgemein eingeführt wurden, sind in dieser Arbeit auf Alpha-Shapes basierend implementiert worden. Die dabei entstandene Software heißt *AlphaDock*. An dieser Stelle werden die einzelnen Entscheidungen bei der Realisierung des Algorithmus begründet und notwendige Zwischenschritte bei der Entwicklung erläutert.

Der gesamte Ablauf des Docking-Prozesses ist schematisch in Abbildung 3.1 dargestellt. Zunächst werden die Alpha-Shapes der beiden Bindungspartner aus deren Proteinstrukturdaten berechnet. Mittels dieser Darstellung werden markante Stellen auf den Oberflächen der Proteine bestimmt. Im nächsten Schritt werden jeweils komplementäre Paare der gefundenen Stellen überlagert und die dadurch entstandenen Proteinkomplexe mit einer schnellen Bewertungsfunktion beurteilt. Alle Komplexe, die ein gewisses Qualtitätsmaß erfüllen, werden lokal optimiert und mit einer weiteren Bewertungsfunktion überprüft. Die folgenden Abschnitte beleuchten die Details des Verfahrens.

3.1 Interne Repräsentation

Die Basis jedes Docking-Verfahrens ist die interne Darstellung der Proteinstrukturen. Im Gegensatz zu den in 2.2.2 beschriebenen Ansätzen, die entweder mit

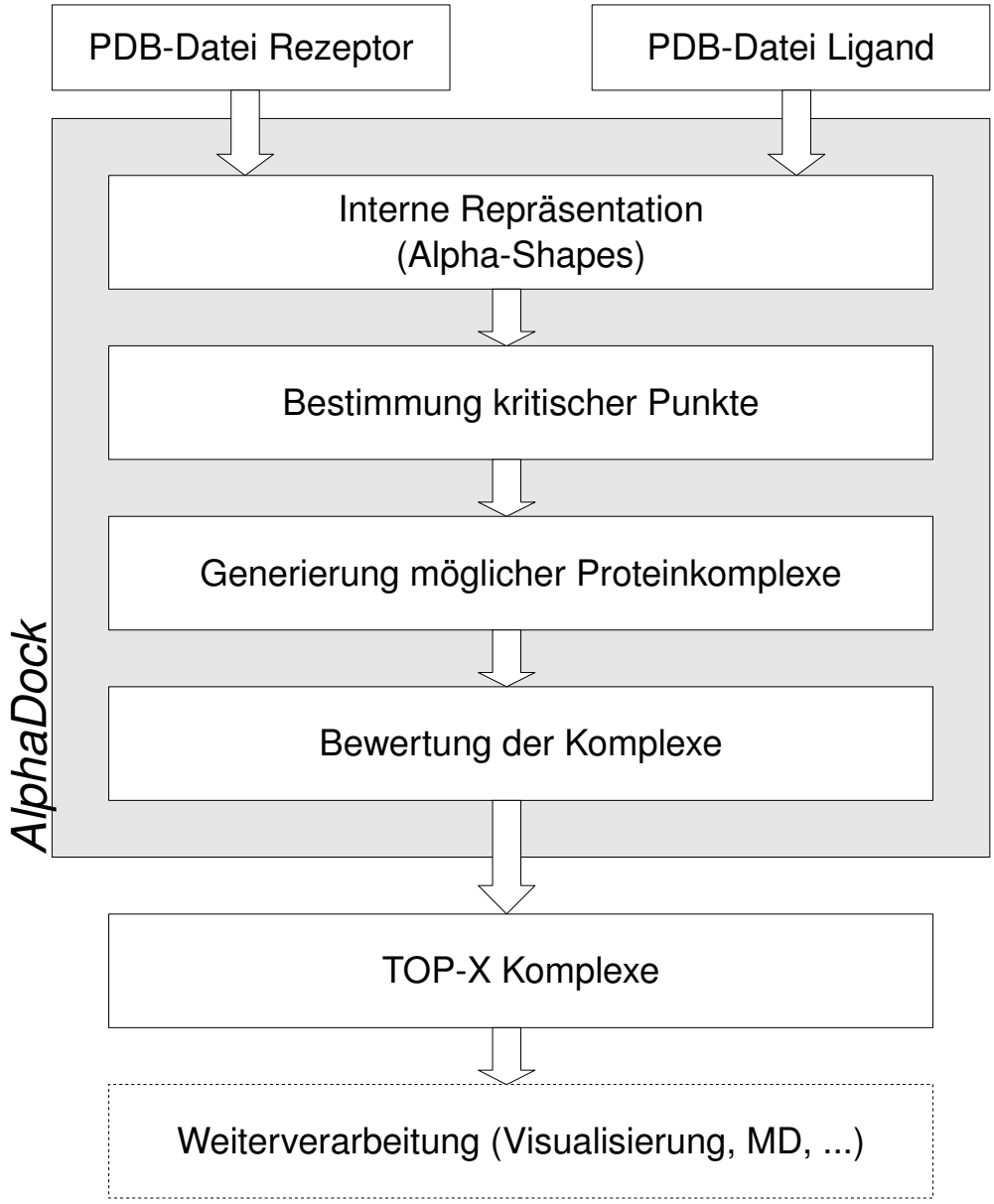


Abbildung 3.1: Schematische Darstellung des in *AlphaDock* implementierten Docking-Verfahrens.

Tabelle 3.1: Van-der-Waals-Radien der in Proteinen vorkommenden Elemente.

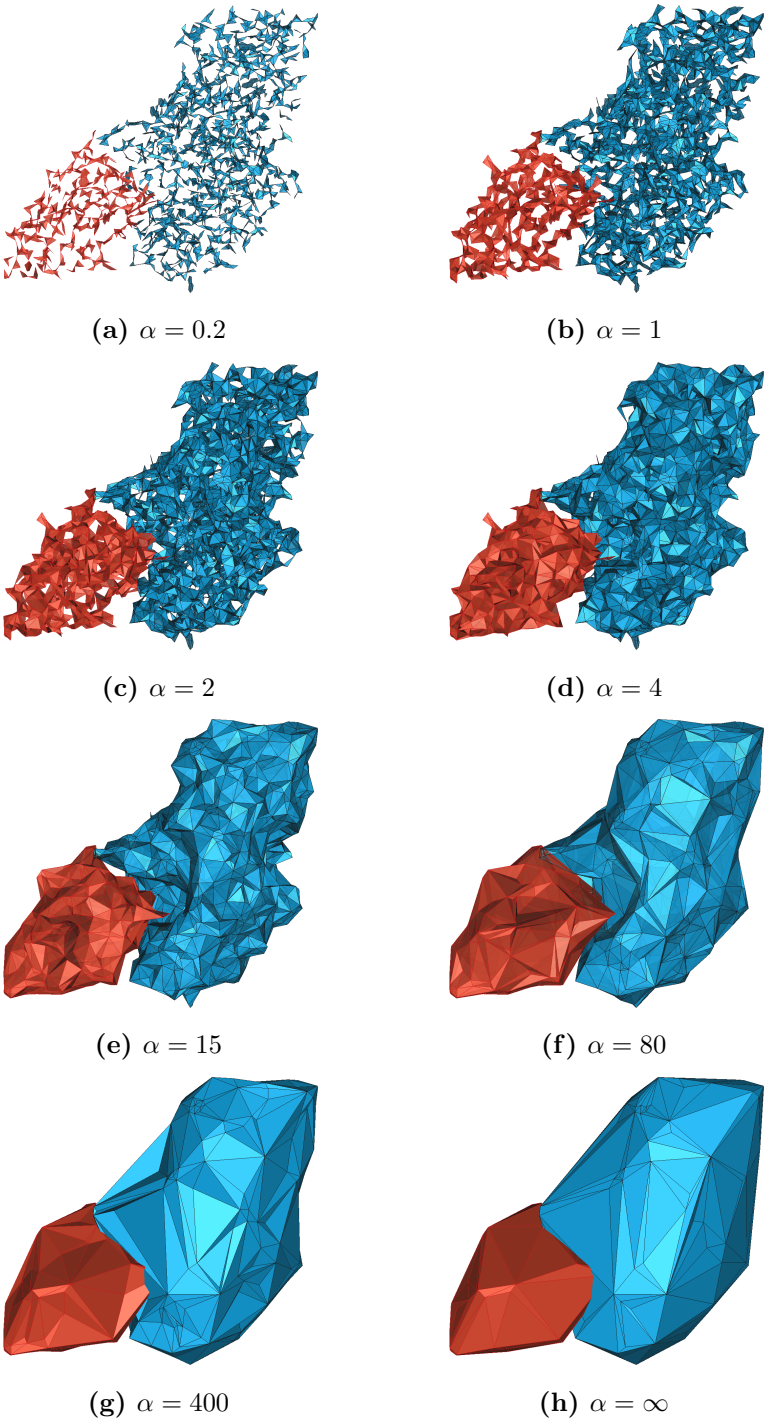
Element	Van-der-Waals-Radius (Å)
Kohlenstoff (C)	1.7
Wasserstoff (H)	1.20
Sauerstoff (O)	1.52
Stickstoff (N)	1.55
Schwefel (S)	1.85
Phosphor (P)	1.9

einer Diskretisierung der Proteine auf ein Gitter arbeiten, oder von der SAS ausgehen und diese auf wenige kritische Punkte reduzieren, verwendet der hier beschriebene Ansatz gewichtete Alpha-Shapes.

Für jedes Atom eines Proteins wird gemäß der Angaben in der entsprechenden PDB-Datei ein gewichteter Punkt erzeugt, entsprechend der Definition im vorherigen Kapitel. Die Koordinate ergibt sich in diesem Fall aus der Position im Molekül, wie sie experimentell bestimmt wurde. Das Gewicht errechnet sich aus den jeweiligen van-der-Waals-Radien der Atome. Die Größen für die verschiedenen Elemente sind nach [12] gewählt und in Tabelle 3.1 aufgeführt. Bei der Berechnung der Alpha-Shape wird für alle Simplizes der zugrunde liegenden gewichteten Delaunay-Triangulation, d.h. für alle Knoten, Kanten, Dreiecke und Tetraeder, jeweils das Intervall von Alpha bestimmt, in dem sie Teile der Alpha-Shape sind. Damit liegt die Alpha-Shape für alle Werte von Alpha vor.

Der van-der-Waals-Radius wird als Gewicht für $\alpha = 0$ verwendet. Entsprechend sind die Alpha-Werte für kleinere Radien negativ. Der kleinste Wert gehört zur Alpha-Shape, die nur noch aus den Punkten besteht, an denen sich die Zentren der Atome befinden. Der Maximalwert für Alpha wird bei Überdeckung der Kanten der konvexen Hülle erreicht.

Die Programmierung der Docking-Software erfolgte unter Verwendung der robusten Implementierung gewichteter Alpha-Shapes aus der *Computational Geometry Algorithms Library* (CGAL) [2, 21]. Diese orientiert sich an der mathematischen Formulierung von Mücke in [72]. Wie dort beschrieben, lässt sich das Problem der Notwendigkeit der *General Position Assumption* durch infinitesimale Perturbationen der Koordinaten lösen.



(a) $\alpha = 0.2$ **(b)** $\alpha = 1$

(c) $\alpha = 2$ **(d)** $\alpha = 4$

(e) $\alpha = 15$ **(f)** $\alpha = 80$

(g) $\alpha = 400$ **(h)** $\alpha = \infty$

Abbildung 3.2: Alpha-Shape von 1KXQ für verschiedene Werte von Alpha. Für $\alpha = \infty$ entspricht die Alpha-Shape der konvexen Hülle.

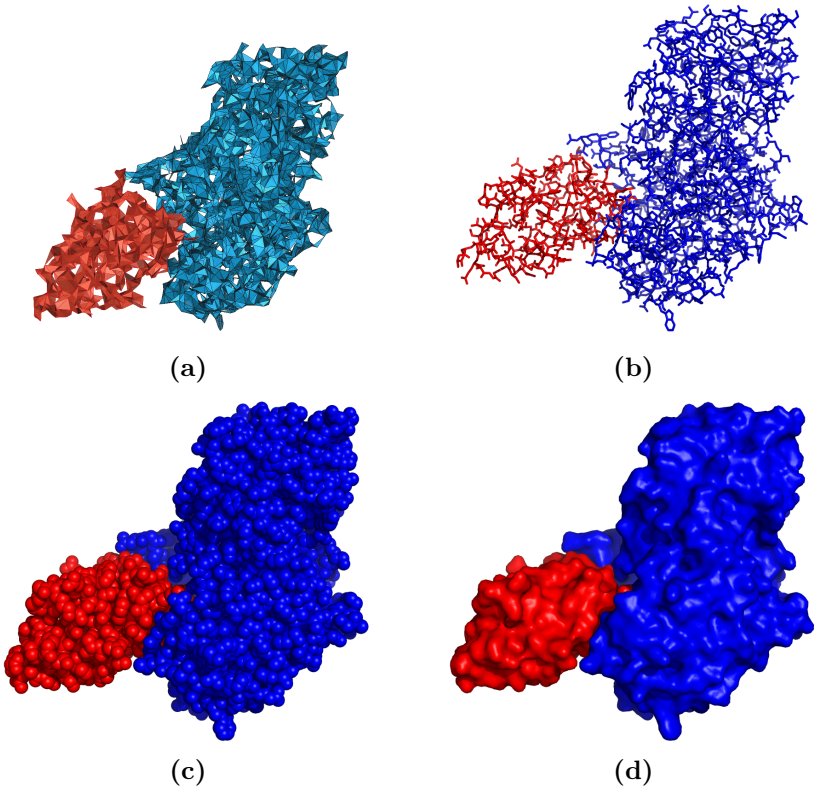


Abbildung 3.3: Vergleich der Alpha-Shape von 1KXQ für $\alpha = 2$ (a) und dem Komplex in *Stab-* (b), *Kugel-* (c) und *Oberflächendarstellung* (d).

Abbildung 3.2 zeigt die Alpha-Shape von 1KXQ für verschiedene Alpha. Es ist gut zu erkennen, wie die Alpha-Shape die Form des Proteins annähert und deren charakteristischen Merkmale wiedergibt. Ein Vergleich der Alpha-Shape des selben Komplexes mit dessen Stab-, Kugel- bzw. Oberflächendarstellung in Abbildung 3.3 verdeutlicht dies.

Abbildung 3.4 zeigt die Alpha-Shape vom Komplex 1EER, wobei die Dreiecke der Alpha-Shape gemäß der Temperaturfaktoren in der PDB-Datei eingefärbt worden sind. Alpha wurde für die Bindungspartner so gewählt, dass die entsprechende Alpha-Shape aus genau einer zusammenhängenden Komponente besteht. Im Bild sind zudem einige besonders lange Kanten der Delaunay-Triangulationen der Proteine zu sehen, die nicht Teil der Alpha-Shape sind. Ihre Bedeutung für die Entwicklung des Docking-Algorithmus wird an spätere Stelle ersichtlich.

Entscheidend für die Wahl von Alpha-Shapes zur Repräsentation von Proteinen in dieser Arbeit sind mehrere Faktoren. Zum einen die Möglichkeit, mit einem

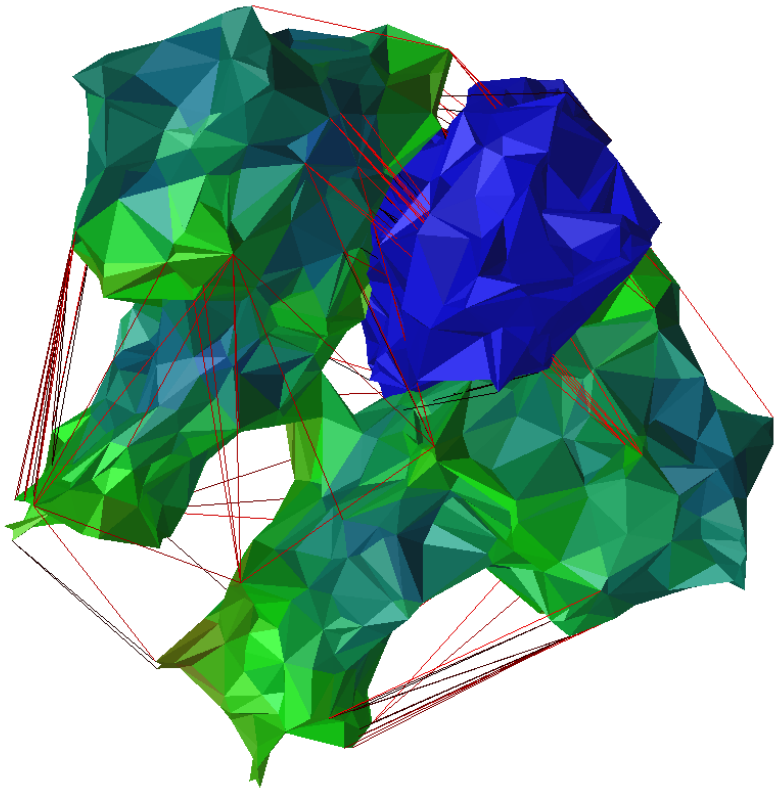


Abbildung 3.4: Alpha-Shapes und ausgewählte Kanten der Delaunay-Triangulation des Komplexes 1EER. Farbgebung der Dreiecke gemäß der Temperaturfaktoren.

Parameter verschiedene Detailstufen der Geometrie auswählen zu können, was eine flexible Verarbeitung in einer Docking-Software erlaubt. Zum anderen wählt man eine grobe Auflösung in der nur wenige Dreiecke behandelt werden müssen, um eine schnelle Berechnung zu erreichen. Wird eine höhere Präzision verlangt, kann die Auflösung entsprechend erhöht werden. Die Zahl der zu verarbeitenden Objekte kann so im Laufe des Algorithmus den Anforderungen an die Genauigkeit und die vorhandene Rechenleistung angepasst werden.

Ein weiterer Faktor ist, dass die internen Darstellung der Proteine mittels Alpha-Shapes sämtliche Informationen über die Geometrie der Moleküle beinhaltet. Es handelt sich nicht um eine verlustbehaftete Vereinfachung, wie beispielsweise die Gitterdarstellung. Diese Tatsache wird unter anderem auch in [61] bei der exakten Berechnung von Oberflächen und Volumina von Proteinen ausgenutzt. Die mathematischen Zusammenhänge finden sich in [27]. Das Vorliegen der vollständigen

geometrischen Information ermöglicht die analytische Behandlung entsprechender Protein-Eigenschaften während des Dockings.

Die Berechnung der Alpha-Shape und die Abfrage der Zugehörigkeit zur Alpha-Shape erfolgen im direkten Raum. Das bedeutet, dass im Gegensatz zu den FFT-Verfahren keine Transformation in den Frequenzraum erfolgt und auch keine Rücktransformation nötig ist. Der Docking-Algorithmus läuft vollständig im direkten Raum, so dass alle Protein-Konformationen, die erzeugt und bewertet werden, unmittelbar visualisiert werden können.

3.2 Komplex-Bildung

Der grundlegende Ansatz bei der Erzeugung von Proteinkomplexen in dem hier beschriebenen Algorithmus besteht darin, basierend auf einer groben Klassifizierung der Oberflächengeometrie plausible Konformationen zu bestimmen. Es wird also nicht versucht, möglichst den gesamten Konformationsraum abzusuchen, wie es beispielsweise bei den FFT-Verfahren der Fall ist. Vielmehr wird durch eine entsprechende Vorverarbeitung eine deutlich kleinere Untermenge ausgewählt, die eine noch zu beschreibende lokale Komplementarität besitzt. Durch die geringere Anzahl an Komplexen beschleunigt sich die weitere Verarbeitung. Es es jedoch darauf zu achten, dass die Auswahl nicht zu strikt ausfällt und dadurch keine oder sehr wenige nahe native Komplexe enthält. Des Weiteren sollte die Auswahlprozedur nicht zu rechenintensiv sein, damit der Vorteil der schnellen Weiterverarbeitung nicht aufgewogen wird.

Motiviert wird diese Herangehensweise zum einen durch die Reduktion der Proteinoberfläche auf wenige maßgebliche Punkte, sogenannte *sparse critical points*, wie sie von Li et al. in [64] beschrieben und von Wolfson und Nussinov in ihrem auf Geometric Hashing basierenden Protein-Docking-Verfahren verwendet werden [104, 74, 23]. Zum anderen begründet eine Arbeit von Edelsbrunner den Ansatz, in der gezeigt wird, wie mittels Alpha-Shapes Taschen und Höhlen in Proteinen definiert werden können [28]. Die Idee besteht nun darin, diese beiden Konzepte zu kombinieren, d.h. Alpha-Shapes zu verwenden, um entscheidende geometrische Merkmale zu detektieren und diese in einem Docking-Algorithmus zu benutzt.

3.2.1 Analyse des Interfaces

Um den im letzten Abschnitt vorgestellten Ansatz verfolgen zu können, mussten zunächst bekannte Protein-Interfaces analysiert werden. Dabei galt es zu verstehen, wie charakteristische Eigenschaften der Bindungspartner in der Alpha-Shape-Darstellung wiedergespiegelt werden. Besonderes Augenmerk wurde dabei auf geometrische Merkmale gelegt, im Speziellen die Komplementarität der Bindungspartner im Interface. Es wurden aber auch andere physiko-chemische Parameter betrachtet. Zu diesem Zweck wurden die Proteinkomplexe aus dem Benchmark 2.4 Datensatz [69] bei verschiedenen Alphas untersucht. Im folgenden werden die dazu verwendeten Analyseprozeduren beschrieben.

Zuerst wurde bestimmt, wie Alpha im Allgemeinen zu wählen ist, um die Form von Proteinen entweder ausreichend grob oder eher detailliert anzunähern. Hierzu wurden die Alpha-Shapes der beiden Bindungsparter für alle Komplexe im Datensatz bestimmt und Alpha ausgehend von der konvexen Hülle schrittweise verkleinert. Dabei wurden jeweils die Anteile von Atomen des einen Proteins innerhalb bzw. außerhalb der Alpha-Shape des anderen Proteins berechnet, wobei sich beide in korrekt gedockter Anordnung befanden. Enthält die Alpha-Shape des einen Proteins Atome des anderen Proteins bedeutet dies, dass noch nicht alle Details der Geometrie durch die Alpha-Shape wiedergegeben werden.

Gesucht wurden zwei Schwellen für Alpha: ein möglichst großes Alpha mit einem nicht zu großen Anteil von Atomen innerhalb der Alpha-Shape des Bindungspartners für eine grobe Beschreibung der Proteine, im Folgenden bezeichnet als α_{grob}, und ein ausreichend kleines Alpha mit möglichst keinen Atomen in der anderen Alpha-Shape für eine detailliertere Auflösung, genannt α_{fein}. Zu ermitteln war zudem, ob sich verallgemeinerbare Werte für diese Alphas finden lassen, d.h. Werte die unabhängig von der Größe und der Klasse der Proteine sind, oder abhängig von diesen Faktoren verschiedene Alphas herangezogen werden müssen. Der Grund für die Bestimmung von α_{grob} und α_{fein} besteht darin, dass aus der Differenz der dazugehörigen Alpha-Shapes für ein Protein Vertiefungen und Erhebungen auf dessen Oberfläche berechnet werden können, wie weiter unten erläutert wird.

Klassifikation von Tetraedern

Die CGAL-Implementierung von Alpha-Shapes ermöglicht es, die Klassifikation in Bezug zur Alpha-Shape für ein gegebenes Alpha für alle Simplizes der zugrunde liegenden Delaunay-Triangulation abzufragen. Dabei werden die folgenden drei Möglichkeiten unterschieden, die für alle Tetraeder, Dreiecke, Kanten und Knoten jeweils abhängig von Alpha vorliegen können: Als *regulär* werden Simplizes bezeichnet, die sich auf dem Rand des Alpha-Komplexes befinden und damit Teil der Alpha-Shape sind. *Interne* Simplizes sind Bestandteil des entsprechenden Alpha-Komplexes, liegen aber nicht auf dessen Rand. *Externe* Simplizes liegen für das gegebene Alpha außerhalb des Alpha-Komplexes.

Ausgehend von den drei Klassen wurde für die Analyse der Protein-Interfaces die Unterteilung der Tetraeder weiter verfeinert. Für jedes Tetraeder wird dabei die Klassifikation der vier Dreiecke betrachtet aus denen dieser besteht. Abhängig von deren Zugehörigkeit zum Alpha-Komplex werden folgende Tetraederklassen definiert: liegen alle vier Seiten außerhalb des Alpha-Komplexes wird das Tetraeder wie gehabt als *extern* bezeichnet. Analoges gilt für Tetraeder, deren Dreiecke alle innerhalb des Komplexes liegen, auch diese werden als *intern* klassifiziert. Liegt jedoch mindestens eine der vier Seiten auf dem Rand des Komplexes und ist damit *regulär*, so wird abhängig von den restlichen Dreiecken zwischen *intern-regulär* und *extern-regulär* unterschieden. Ersteres ist der Fall, wenn sich ein Tetraeder aus internen und regulären Dreiecken zusammensetzt. Zweiteres analog, wenn es sich um externe und reguläre Seiten handelt. Tetraeder aus den beiden letztgenannten Klassen werden auch als *interne* bzw. *externe Randtetraeder* bezeichnet. Sie sind bei der Erzeugung und Bewertung von Komplexen besonders relevant.

Klassifikation von Kanten

Ähnlich wie bei den Tetraedern lässt sich für die Kanten der Triangulation eine erweiterte Klassifikation einführen. Betrachtet werden nur die regulären Kanten, d.h. alle die bei einem gegebenen Alpha auf dem Rand des Alpha-Komplexes liegen. Ermittelt werden jeweils die beiden regulären Dreiecke, die eine Kante bilden. Abhängig vom Winkel zwischen deren Flächen erfolgt die Einteilung der dazugehörigen Kante. Ist der Winkel kleiner als 180° handelt es sich um eine

konkave Kante. Ein Winkel größer 180° charakterisiert eine *konvexe* Kante. Der Winkel wird jeweils in Bezug auf die sichtbar Seite der Dreiecke berechnet, die über den entsprechenden Normalenvektor bestimmt werden kann.

Klassifikation von Knoten

Den Knoten der Triangulation, die jeweils für ein Atom im Protein stehen, lässt sich mittels obiger Klassifikation der einfallenden Kanten ebenfalls eine Klasse zuordnen. Da eine Einteilung ausschließlich für reguläre Kanten existiert, werden nur diese berücksichtigt. D.h. es werden die Atome an der durch die Alpha-Shape definierten Oberfläche des Proteins betrachtet. Ebenso wie die Einteilung der Kanten ist die Klassifikation dadurch abhängig von Alpha. Enden in einem Knoten nur Kanten einer Klasse, wird dieser analog als *konkav* oder *konvex* bezeichnet. Treffen in einem Knoten sowohl konkave als auch konvexe Kanten aufeinander, entscheidet die höhere Anzahl über die Zuordnung. Fallen gleich viele Kanten aus beiden Klassen ein, liegt eine Art Sattelpunkt vor und der Knoten wird *flach* genannt.

Weitere Eigenschaften innerhalb der Triangulation

Um noch mehr über Protein-Interfaces in der Alpha-Shape-Darstellung zu erfahren, wurden neben der oben beschriebenen Klassifikation der verschiedenen Simplizes noch folgende Parameter bestimmt: Das *Volumen* der Tetraeder, die *Länge* der Kanten, der *Abstand* zum nächsten Tetraeder und der *Abstand* zum nächsten Atom des Bindungspartners. Die Werte wurden jeweils innerhalb des tatsächlichen Interfaces ermittelt, d.h. für alle Atome mit einem Abstand unter sechs Ångström zu einem Atom des Bindungspartners in korrekt gedockter Anordnung. Zum Vergleich wurden die Faktoren für einige Atome außerhalb des Interfaces berechnet.

Weiterhin wurden die physiko-chemischen Eigenschaften der vier zu einem Tetraeder gehörenden Atome gemittelt und der Wert dem entsprechenden Tetraeder zugeordnet. So kann jedes Tetraeder z.B. eine *Polarität* und eine *Hydrophobizität* erhalten. Analog ergibt sich ein mittlerer *Temperaturfaktor* pro Tetraeder und es

lassen sich *Meta-Elemente* und *Meta-Aminosäuren* definieren, die sich jeweils aus den vier Elementen bzw. Aminosäuren eines Tetraeders ableiten.

3.2.2 Geometrische Vorverarbeitung

Aus den Erkenntnissen der oben vorgestellten Analyse bekannter Interfaces wurde ein Ablauf für eine geometrische Vorverarbeitung entwickelt. Die Ausgabe dieser Prozedur dient als Eingabe für die im nächsten Paragraph erläuterten Komplex-Generatoren.

Als erstes werden für die zu dockenden Proteine jeweils die Simplizes bestimmt, die zur Alpha-Shape für α_{grob} und α_{fein} gehören, d.h. auf dem Rand des entsprechenden Alpha-Komplexes liegen. Die Dreiecke der beiden Alpha-Shapes bilden jeweils eine Oberfläche, die gemäß der oben beschriebenen Anforderung die Form des Proteins im Gesamten (G) oder im Detail (D) wiedergibt. Die Flächen werden im Folgenden entsprechend *GF* und *DF* genannt. Aus der Differenz der beiden Oberflächen lassen sich Einbuchtungen auf der tatsächlichen Proteinoberfläche bestimmen, die als potentielle Bindungsstellen gewertet werden können. Konkret wird für jedes Atom i in DF der Abstand d_i zu GF berechnet. Die Abstände d_i sind ein Maß für die *Tiefe* der Atome innerhalb des Proteins in Bezug auf GF. Mögliche Bindungstaschen zeichnen sich dadurch aus, dass hinreichend viele benachbarte Atome in ähnlicher Tiefe liegen, d.h. die Tasche ausreichend groß ist. Um diese Anforderung besser überprüfen zu können, wird für jedes Atom jeweils der Mittelwert der Abstände über alle Nachbaratome berechnet, wobei die Nachbarschaftsbeziehung durch die Triangulation bestimmt wird. Man erhält dadurch einen geglätteten Abstand $\hat{d}_i$, der folgendermaßen definiert ist:

$$\hat{d}_i = \frac{d_i + \sum_j d_j}{1 + \sum_J}, \text{ mit } j \text{ Nachbar von } i. \tag{3.1}$$

Hierbei bezeichnet $\sum_J$ die Anzahl der Nachbarknoten von i, deren Abstände zu GF entsprechend d_j heißen. Mögliche Bindungsstelle erhält man durch das Clustern der $\hat{d}_i$. Dabei fasst man alle Atome in einem gewissen Radius R_d zusammen, deren Differenz $\Delta_{ij} = \hat{d}_i - \hat{d}_j$ kleiner als eine Schwelle $\delta_{\hat{d}}$ ist. Eine vergleichbare Herangehensweise verwenden Peters et al. [78], um Bindungsstellen beim Protein-Ligand-Docking zu detektieren.

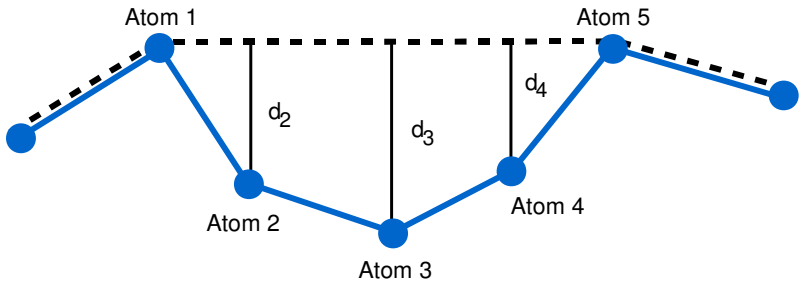


Abbildung 3.5: Berechnung der Atomtiefen mit Hilfe der groben (- - -) und der detailierten Oberfläche (—).

Die Bestimmung der potentiellen Gegenstücke zu den möglichen Bindungstaschen geschieht auf folgende Weise. Gesucht werden Atome, die möglichst exponiert auf der Proteinoberfläche liegen. Man findet sie, indem man erneut die Flächen GF und DF vergleicht: alle Knoten, die auf *beiden* Flächen zur Alpha-Shape gehören werden ausgewählt. Dass sie die gestellte Anforderung an die Position im Protein erfüllen, erkennt man gut im zweidimensionalen Beispiel auf Abbildungg 3.5. Eine weitere Unterteilung der so gefundenen Atome erfolgt anhand der Winkel zwischen den Dreiecken der DF, die diese Atome jeweils als Eckpunkt besitzen. Je nachdem wie groß die Winkel zwischen diesen Flächen sind, unterscheidet man zwischen *spitzen* und *stumpfen* möglichen Bindungsstellen.

3.2.3 Komplex-Generatoren

Der nächste Schritt im Docking-Algorithmus besteht darin, mit Hilfe der zuvor auf der Proteinoberfläche bestimmten Stellen Proteinkomplexe zu erzeugen. Im Folgenden werden zwei der unterschiedlichen Verfahren vorgestellt, die bei der Entwicklung der Docking-Software implementiert worden sind. Grundsätzlich ist dabei zu unterscheiden, wie viele Gegensatzpaar zur Erzeugung eines Komplexe herangezogen werden. Hier wurden Generatoren verwendet, die ein oder zwei Paare benutzen. Die besten jeder Klasse werden im Detail vorgestellt.

Wird nur eine potentielle Bindungsstelle zur Komplexbildung verwendet, d.h. genau eine Erhebung des einen Proteins in eine Tasche des anderen platziert, so stellt sich die Frage, wie die Position der Proteine zueinander genau gewählt werden soll. Nimmt man jeweils einen Punkt zur Charakterisierung einer Erhe-

bung bzw. einer Tasche, so erhält man nur zwei Punkte. Das reicht jedoch nicht aus, um die Position der Proteine im Raum eindeutig festzulegen. Dafür würden pro Protein drei nicht kolineare Punkte benötigt. Um die übrigen Freiheitsgrade zu berücksichtigen, müssen weitere Merkmale der möglichen Bindungsstellen extrahiert und zur Festlegung der Anordnung der Proteine verwendet werden. Zu diesem Zweck wird für jedes Atom eine sogenannte *Pseudonormale* $\tilde{\mathbf{n}}$ berechnet. Diese stellt eine Näherung an die Normale auf der Proteinoberfläche an der Stelle des Atoms dar und wird bestimmt, indem über die Normalen der einfallenden Dreiecke der detaillierten Alpha-Shape gemittelt wird, wobei diese jeweils mit dem Flächeninhalt des Dreiecks gewichtet werden. Für ein Atom p mit den einfallenden Dreiecken i und den zugehörigen Normaleneinheitsvektoren $\mathbf{n_i}$ ergibt sich somit für ein festes Alpha:

$$\tilde{\mathbf{n}}_p(\alpha) = \sum_i A_i \mathbf{n}_i(p, \alpha), \text{ wobei } A_i \text{ der Flächeninhalt von } i \text{ ist.} \tag{3.2}$$

Die Anordnung der Bindungspartner erfolgt nun so, dass zusätzlich zu den zwei Punkten die entsprechenden Pseudonormalen von zwei gegensätzlichen Stellen zur Deckung gebracht werden. Dies geschieht, nachdem das Vorzeichen der zum Ligand gehörenden Normalen umgekehrt wird, um der Komplementarität der Stellen Rechnung zu tragen. Damit bleibt aber immer noch ein Freiheitsgrad. Dieser wird durch eine schrittweise Drehung eines der Proteine um die durch die Normale bestimmte Achse berücksichtigt, d.h. es wird pro Punktepaar nicht nur eine Anordnung erzeugt, sondern mehrere. Die genaue Anzahl hängt von der Schrittweite der beschriebenen Drehung ab.

Die Erzeugung von Proteinkomplexen mit jeweils zwei kritischen Punkten pro Protein erweist sich im Vergleich zu dem zuvor beschriebenen Ansatz als erfolgreicher, d.h. es werden damit mehr nahe native Komplexe produziert. Dieses Vorgehen wird deshalb im folgenden detaillierter beschrieben. Grundsätzlich sind zunächst drei Fälle zu unterscheiden: 1. es werden zwei Erhebungen des Liganden und zwei Vertiefungen des Rezeptors behandelt, 2. der umgekehrte Fall von zwei Vertiefungen des Liganden und zwei Erhebungen des Rezeptors und 3. sowohl eine Vertiefung und eine Erhebung des Liganden als auch des Rezeptors. Diese drei Fälle werden in der Docking-Software unterschieden und können dadurch in der Berechnung mit angepassten Parametern behandelt werden. Die Fallunterschei-

dung kann auch in der Auswertung verwendet werden. Die weiteren Ausführungen gelten aber gleichermaßen für alle drei Konstellationen.

Werden zwei Punkte pro Bindungspartner verwendet, stellt sich zunächst die Frage, wie die Auswahl pro Protein im Vorfeld eingeschränkt werden kann, da die Kombination von jedem kritischen Punkt mit jeweils allen anderen dieses Proteins einen zu großen Suchraum ergeben würde. Die Beschränkung der Wahl ist leicht dadurch zu begründen, dass die Ausdehnung des Interfaces sich nicht über das ganze Protein erstreckt und dadurch eine Nachbarschaft der gemeinsam auszuwählenden Punkte anzunehmen ist. Im konkreten Fall bedeutet dies, dass pro kritischem Punkt des Liganden nur die anderen der gesuchten Klasse gewählt werden, die einen Maximalabstand nicht überschreiten. Zusätzlich wird auch eine untere Schranke für den Abstand angegeben, um direkte Nachbarn auszuschließen. Für die daraus gebildeten Paare werden jeweils komplementäre Paare des Rezeptors gesucht, die den gleichen Abstand wie das betrachtete Ligand-Paar besitzen, wobei jeweils noch eine Toleranz von $\pm\Delta$ Ångström zugelassen wird.

Bei der Erzeugung von Komplexen in der Software wird der Rezeptor, welcher meistens das größere Protein ist, festgehalten und der Ligand wie oben beschrieben darum herum platziert. Für jede Position des Liganden wird die dazu nötige affine Transformation in Bezug auf eine Grundstellung berechnet und die zugehörige Transformationsmatrix gespeichert. Als Ausgangsposition wird die Position des Liganden in der PDB-Datei verwendet. Die gesamte Transformation wird schrittweise berechnet. Die einzelnen Schritte sind jeweils affine Transformationen, die noch im einzelnen beschrieben werden. Die Matrix der Gesamttransformation erhält man durch Multiplikation der einzelnen Transformationsmatrizen in der richtige Reihenfolge.

Bei den affinen Transformationen handelt es sich entweder um Translationen ($\mathcal{T}_n$) oder Rotationen ($\mathcal{R}_n$), die jeweils in Form einer 4×4 Matrix $(\mathbf{m_{ij}})_{i,j=0\ldots3}$ beschrieben werden können. Dabei sind m_{30}, m_{31} und m_{32} immer Null und m_{33} gleich Eins. Bezeichnen $\mathbf{l_a}$ und $\mathbf{l_b}$ die Ortsvektoren der zwei gewählten Punkte L_a und L_b des Liganden und sind $\mathbf{r_a}$ und $\mathbf{r_b}$ die entsprechenden Ortsvektoren des Rezeptors, so ergeben sich die beiden Vektoren $\mathbf{l} = \mathbf{l_b} - \mathbf{l_a}$ und $\mathbf{r} = \mathbf{r_b} - \mathbf{r_a}$, die jeweils die beiden gewählten Punkte des Liganden und des Rezeptors verbinden. Zur Positionierung des Liganden werden nun zunächst $\mathbf{l}$ und $\mathbf{r}$ zur Deckung ge-

bracht. Dies geschieht in drei Schritten. Als erstes wird L_a in den Ursprung des Koordinatensystems gelegt, was mittels einer Translation um $-\mathbf{l_a}$ erreicht wird, womit $\mathcal{T}_1$ bestimmt ist. Die Verschiebung ist nötig, damit die Rotation $\mathcal{R}_1$, die $\mathbf{l}$ auf $\mathbf{r}$ dreht, um die Achse $\mathbf{d} = \mathbf{l} \times \mathbf{r}$ erfolgen kann. Der Drehwinkel dabei ist durch folgendes Skalarprodukt festgelegt:

$$\varphi = \arccos\left(\frac{\mathbf{l} \cdot \mathbf{r}}{|\mathbf{l}||\mathbf{r}|}\right). \tag{3.3}$$

Mit $\mathbf{v} = \mathbf{d}/|\mathbf{d}|$ und $C = (1 - cos\varphi)$ lässt sich die Drehmatrix schreiben:

$$\begin{pmatrix} \cos\varphi + v_1^2 C & v_1 v_2 C - v_3 \sin\varphi & v_1 v_3 C + v_2 \sin\varphi \\ v_2 v_1 C + v_3 \sin\varphi & \cos\varphi + v_2^2 C & v_2 v_3 C - v_1 \sin\varphi \\ v_3 v_1 C - v_2 \sin\varphi & v_3 v_2 C + v_1 \sin\varphi & \cos\varphi + v_3^2 C \end{pmatrix}. \tag{3.4}$$

Bevor nun eine Verschiebung des Liganden um $\mathbf{r_a}$ erfolgt, die L_a auf R_a positioniert ($\mathcal{T}_2$), wird eine weitere Rotation ($\mathcal{R}_2$) vorgenommen. Als Drehachse wird $\mathbf{r}$ benutzt, was nach der ersten Drehung auch die Richtung von $\mathbf{l}$ ist. Durch die zweite Drehung wird der noch verbleibende Freiheitsgrad des Liganden festgesetzt. Der Drehwinkel wird in diesem Fall mit Hilfe der oben beschriebenen Pseudonormalen berechnet. Bei der Wahl von zwei Punkten pro Protein ergeben sich insgesamt vier dieser Vektoren und damit auch mehrere Möglichkeiten, durch eine Überlagerung einen Winkel der dazu nötigen Drehung zu ermitteln.

Im vorliegenden Programm werden drei Varianten berücksichtigt: Zum einen werden die beiden Normalen eines Proteins gemittelt und die beiden resultierenden Vektoren zur Deckung gebracht. Die beiden anderen Fälle drehen jeweils die Pseudonormalen von R_a und L_a, respektive die von R_b und L_b, aufeinander. Es werden somit jeweils zwei Pseudonormalen behandelt, die im Folgenden mit $\tilde{\mathbf{n}}_r$ und $\tilde{\mathbf{n}}_l$ bezeichnet werden. Da die Drehachse, wie oben beschrieben, bereits durch $\mathbf{r}$ vorgegeben ist, können nicht die Pseudonormalen selbst übereinander gedreht werden, sondern nur die beiden Ebenen, die durch $\mathbf{r}$ und $\tilde{\mathbf{n}}_r$ bzw. durch $\mathbf{l}$ und $\tilde{\mathbf{n}}_l$ aufgespannt werden. Der gesuchte Drehwinkel kann dabei mittels der Normalenvektoren dieser Ebenen berechnet werden, die sich jeweils aus dem Kreuzprodukt der entsprechenden Pseudonormalen mit der Drehachse ergeben.

Bezeichnen $\mathbf{n}_r = \tilde{\mathbf{n}}_r \times \mathbf{r}$ und $\mathbf{n}_l = \tilde{\mathbf{n}}_l \times \mathbf{r}$ die Normalenvektoren auf den Ebenen, so ergibt sich analog zu 3.3 der Drehwinkel

$$\varrho = \arccos\left(\frac{\mathbf{n}_r \cdot \mathbf{n}_l}{|\mathbf{n}_r||\mathbf{n}_l|}\right) \tag{3.5}$$

und mit $\mathbf{v} = \mathbf{r}/|\mathbf{r}|$ und $C = (1 - cos\varrho)$ lässt sich erneut die Drehmatrix mit Gleichung 3.4 schreiben ($\mathcal{R}_2$).

Nach der Positonierung von L_a auf R_a erfolgen noch zwei weitere Translationen ($\mathcal{T}_3$, $\mathcal{T}_4$) . Die erste verschiebt den Liganden so in Richtung von $\mathbf{r}$, dass der Abstand zwischen L_a und R_a genau dem zwischen L_b und R_b entspricht. Das ist sinnvoll, da die Abstände der kritischen Punkte der beiden Proteine, wie beschrieben, um bis zu Δ Ångström abweichen können. Die letzte Bewegung vergrößert den Abstand des Liganden zum Rezeptor in Richtung der bei der Berechnung von $\mathcal{R}_2$ verwendeten Pseudonormalen des Rezeptors. Das ist notwendig, da bisher nur Punkte ohne Ausdehnung in der Berechnung angenommen wurden, aber in Wirklichkeit Atome mit einer Ausdehnung behandelt werden.

Die Gesamtbewegung, die den Liganden in eine zu bewertende Position bringt, läßt sich somit durch die folgende affine Transformation beschreiben, die sich aus der Multiplikation der eben erläuterten Matrizen ergibt:

$$\mathcal{M} = \mathcal{T}_4 \cdot \mathcal{T}_3 \cdot \mathcal{T}_2 \cdot \mathcal{R}_2 \cdot \mathcal{R}_1 \cdot \mathcal{T}_1. \tag{3.6}$$

Wie in den letzten Abschnitten zu erkennen ist, gibt es eine Vielzahl von Parametern, die bei der Berechnung der Matrizen variiert werden können. Im vierten Kapitel werden die Ergebnisse für verschiedene Werte dieser Einflussfaktoren vorgestellt und die Wahl gewisser Größen motiviert.

3.3 Komplex-Bewertung

Der nächste Schritt im Ablauf des Docking-Algorithmus besteht darin, die im letzten Abschnitt berechneten affinen Transformationen auf den Liganden anzuwenden und die daraus entstehenden Komplexe zu bewerten. Da die grundlegende Idee des hier vorgestellten Ansatzes darin besteht, möglichst alle Berechnungen

anhand der Alpha-Shape der Proteine durchzuführen, gilt dies wie zuvor bei der Ermittlung der Ligand-Transformationen, auch für die Beurteilung der erzeugten Protein-Anordnungen.

3.3.1 Geometrische Korrelation

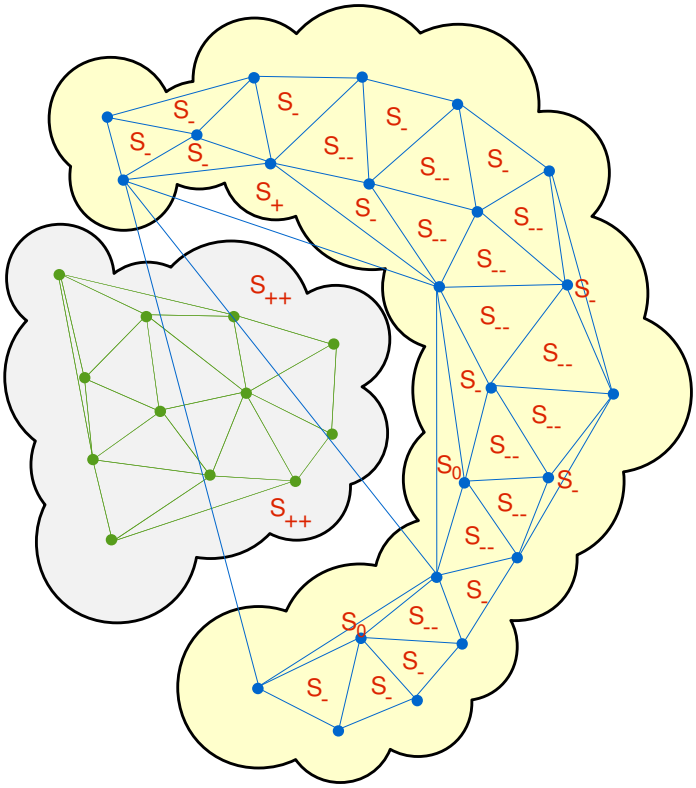


Abbildung 3.6: Veranschaulichung der prinzipiellen Idee der Bewertung der geometrischen Korrelation mit Hilfe von Alpha-Shapes. Nähere Erläuterung im Text.

Grundsätzlich kann die Bewertung in mehreren Stufen erfolgen. Wie bereits in 2.2.4 erwähnt, ist die geometrische Korrelation eine gute erste Annäherung an die Beurteilung der Qualität eines betrachteten Komplexes. Verwendet man Alpha-Shapes zur Beschreibung der Proteine, so lässt sich die geometrische Passgenauigkeit eines Komplexes auf folgende Weise bewerten: für jedes Atom des Liganden wird dessen Position innerhalb der Alpha-Shape des Rezeptors bestimmt und abhängig davon ein Score berechnet. Die gesamte Bewertung ergibt sich dann aus der Summe der einzelnen Scores.

Zu klären bleib noch, welche Positionen mit welchen Werten belegt werden. Dabei erfolgt zunächst die Unterscheidung, ob der betrachtete Punkt, d.h. der Mittelpunkt des Atoms, innerhalb oder außerhalb der konvexen Hülle des Rezeptors liegt. Im ersten Fall wird das Tetraeder der der Alpha-Shape zugrunde liegenden Triangulation bestimmt, das den Punkt enthält. Im zweiten Fall wird das Atom des Rezeptors ermittelt, dass den geringsten Abstand zum betrachteten Atom besitzt. Liefert die Positionsabfrage ein Tetraeder, ergibt sich der Score aus dessen Position innerhalb der zugehörigen Alpha-Shape, wobei analog zu Abschnitt 3.2.1 zwischen *internen*, *externen*, *intern-regulären* und *extern-regulären* Tetraedern unterschieden wird. Die im einzelnen benutzten Werte für die verschiedenen Tetraeder und eine noch feinere Klasseneinteilung werden in Kapitel 4 im Zusammenhang mit den erzielten Ergebnissen vorgestellt. Grundsätzlich gilt es noch einmal hervorzuheben, dass die Klassenzugehörigkeit jeweils von Alpha abhängt und damit auch der Gesamtscore dieser Abhängigkeit unterliegt. Bei gegebenem Alpha kann aber jedem Tetraeder ein fester Score zugewiesen werden und die Scoring-Funktion sehr effizient mit einer Hashtabelle realisiert werden. D.h. zu Beginn des Algorithmus wird für jedes Tetraeder der Score berechnet und in einer Hashtabelle notiert. Anschließend kann die Tabelle für alle Bewertungen von Komplexen verwendet werden. Abbildung 3.6 veranschaulicht die Idee dieser geometrischen Bewertungsfunktion anhand eines zweidimensionalen Beispiels. Das Prinzip ist dasselbe wie in drei Dimensionen, nur dass die Gewichte in diesem Fall in Dreiecken und nicht in Tetraedern stehen.

Liegt das betreffende Atom des Liganden außerhalb der konvexen Hülle des Rezeptors, kann der Score über eine Funktion bestimmt werden, die von der Entfernung zum nächstgelegenen Atom des Rezeptors abhängt. Der qualitative Verlauf der hier verwendeten Gewichtungsfunktion ist wie folgt. Bei einem bestimmten Abstand d_{opt} existiert ein absolutes Maximum g_{max} von dem aus die Funktion in beide Richtungen linear abfällt. Rechts vom Maximum endet der Verlauf auf der X-Achse, d.h. es gibt einen Abstand d_{max} ab dem Atome keinen Beitrag mehr zum Gesamtscore liefern, da die Gewichtungsfunktion Null ist. Links vom Maximum fällt die Funktion etwas schneller ab und endet für einen verschwindenden Abstand auf der Y-Achse bei $-g_{min}$. Da sich zwei Atome nicht beliebig nah kommen können, wird ein zu geringer Abstand mit einem negativen Score bestraft und so ein abstoßender Effekt zwischen den Atomen simuliert.

Verwendet man die gerade eingeführten Bezeichnungen, so ist die Funktion $g(d)$ wie folgt abschnittsweise definiert:

$$g(d) = \begin{cases} (g_{max} + g_{min})/d_{opt} \cdot d - g_{min} \\ -g_{max}/(d_{max} - d_{opt}) \cdot d + \frac{g_{max} \cdot d_{max}}{(d_{max} - d_{opt})} \\ 0 \end{cases} \quad \text{für} \quad \begin{cases} 0 \leq d < d_{opt} \\ d_{opt} \leq d < d_{max} \\ d \geq d_{max} \end{cases} \tag{3.7}$$

Bezeichnet $s(T)$ die Funktion, die einem Tetraeder einen Score zuordnet, lässt sich zusammen mit Gleichung 3.7 die Bewertungsfunktion für alle Atome des Liganden schreiben:

$$S_L(\alpha) = \sum_{i_{int}} s(T_i) + \sum_{i_{ext}} g(d_i) \tag{3.8}$$

Hier bezeichnen i_{int} die Atome des Liganden innerhalb und i_{ext} die außerhalb der konvexen Hülle des Rezepors.

Erweiterung der Alpha-Shape

Das Problem der Bewertung von Atomen außerhalb der konvexen Hülle des Bindungspartners läßt sich alternativ zu der oben beschriebenen Vorgehensweise mit einer abstandsabhängigen Bewertungsfunktion auch durch eine Erweiterung der Alpha-Shape erreichen. Zur Verdeutlichung der Problematik und der weiteren Lösungsmöglichkeit, dienen die Abbildungen 3.7 und 3.8. Es ist gut zu erkennen, dass mit der bisher benutzten Bewertung kein zählbares Resultat erreicht würde, da alle Atome des Liganden außerhalb der konvexen Hülle des Rezeptors liegen. Um dennoch eine positive Bewertung mit dem beschriebenen Verfahren zu erhalten, wird die Alpha-Shape des Rezeptors um sogenannten *virtuelle Atomen* erweitert. Diese werden wie folgt platziert. Für jedes Atom, das Teil der konvexen Hülle ist, wird ein zusätzliches Atom mit gleichem Radius in Richtung der entsprechenden Pseudonormalen positioniert. Abbildung 3.8 zeigt das in zwei Dimensionen. Anschließend wird die Alpha-Shape neu berechnet. Sie enthält nun eine Randschicht aus Tetraedern, die virtuelle und reguläre Atome als Eckpunkte besitzen. Diesen Tetraedern wird ein positiver Wert zugewiesen, wodurch auch Komplexe favorisiert werden können, die zuvor nicht gefunden werden konnten.

Randtetraeder erhalten im Allgemeinen einen geringeren Score als Tetraeder im inneren der konvexen Hülle. Der Grund dafür ist, dass Komplexe, bei denen Atome innerhalb einer Tasche des Bindungspartner liegen, von der Bewertungsfunktion gegenüber Komplexen bevorzugt werden sollen, bei denen sich Flachstellen gegenüber stehen. In der konkreten Implementierung des Docking-Programms erfolgt die Bewertung mit zwei Alpha-Shapes. Zunächst wird der Score für die Atome des Liganden bezüglich der randlosen Alpha-Shape ermittelt. Liegen zuviel Atome im Inneren der Alpha-Shape, und erhalten damit einen negativen Wert, wird der Komplex verworfen. Anderenfalls werden alle Atome des Liganden, die außerhalb der konvexen Hülle des Rezeptors liegen, in der Alpha-Shape mit Rand verortet und damit die Bewertung verfeinert.

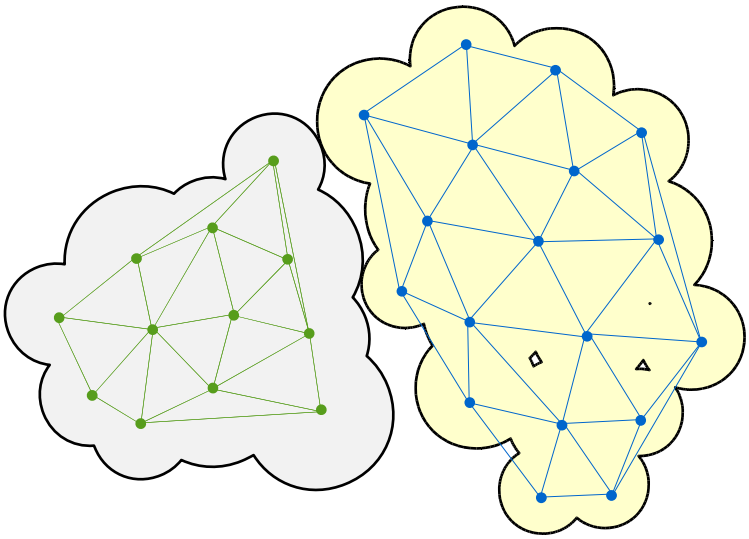


Abbildung 3.7: Problemfall für die Bewertung mittels Scores nur innerhalb der konvexen Hülle der Proteine.

Gesamtscore

Analog wie für den Liganden, läßt sich für den Rezeptor ein Score S_R berechnen. Dabei werden die Atome des Rezeptors in der Alpha-Shape des Liganden verortet und bewertet. Die Parameter können wie zuvor gewählt oder speziell angepasst

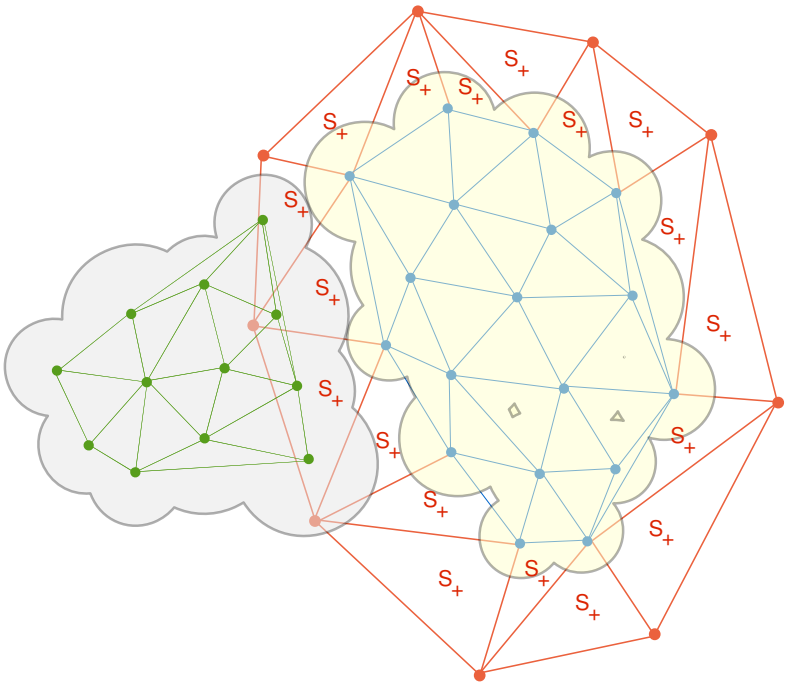


Abbildung 3.8: Lösung der Bewertungsproblematik durch Einführung von *virtuellen Atomen* außerhalb der konvexen Hülle.

werden, insbesondere kann Alpha für die beiden Proteine unterschiedlich sein. Der Gesamtscore für einen Komplex ergibt sich damit zu:

$$S_{Total}(\alpha_l, \alpha_r) = S_L(\alpha_r) + S_R(\alpha_l). \tag{3.9}$$

3.3.2 Erweiterte Bewertungskriterien

Die erste Erweiterung der geometrische Bewertung besteht in einer Einteilung der Atome in solche auf der Oberfläche und solche im Inneren des Proteins. Diese wird für zweierlei benutzt. Zum einen ermöglicht es die schnelle Berechnung eines ersten Scores, indem nur die Oberflächenatome in der Alpha-Shape des Bindungspartners bewertet werden. Zum anderen können so unterschiedliche Gewichte für die beiden Atomklassen vergeben werden, um die Bedeutung von Kontakte von Oberflächenatome zu erhöhen. Die Bestimmung der Oberfläche der Proteine erfolgt über deren Alpha-Shapes, d.h. alle regulären Atome bei einem gegebenen

Alpha werden als Teil der Oberfläche angesehen. Wie Alpha dabei zu wählen ist, wird im nächsten Kapitel anhand von Beispielen gezeigt.

Die folgenden Erweiterungen machen sich zu Nutze, dass die Tetraeder der Alpha-Shape neben dem geometrischen Score noch weitere Bewertungen aufnehmen können. Anschaulich heißt das, dass in jedem Tetraeder für jedes Bewertungskriterium ein Wert steht und der entsprechende Score als Summe über die Tetraeder gebildet wird, in denen Atome des Bindungspartners enthalten sind. Genau so, wie es bei der geometrischen Korrelation erfolgt. Praktisch erfordert das eine zusätzliche Hashtabelle für jedes Kriterium, die die Zuordnung der Tetraeder zu den jeweiligen Scores vornimmt.

In Anlehnung an die in Abschnitt 3.2.1 beschriebenen Kriterien werden im vorliegenden Algorithmus noch folgende Eigenschaften zur Bewertung herangezogen: die *Tiefe* eines Tetraeders wird als Mittelwert der Tiefen der vier Eck-Atome berechnet, die gemäß Gleichung 3.1 bestimmt werden. Ebenfalls als Mittelwert aus vier Werten wird ein *Temperaturfaktor* pro Tetraeder bestimmt. Die einzelnen Faktoren entsprechen dabei denen in der PDB-Datei des Proteins. Qualitativ kann die Bewertung eines Komplexes dann angepasst werden, indem Atome, die tief in der Alpha-Shape des Bindungspartners liegen, einen höheren Score erhalten. Ebenso wie solche, die in Regionen mit einem niedrigen Temperaturfaktor liegen. Der Temperaturfaktor spiegelt die Ungenauigkeit der Bestimmung der Atomposition im Experiment wieder, d.h. je höher der Faktor ist, desto größer kann die Schwankung um die ermittelte Position sein. Da nicht anzunehmen ist, dass sehr bewegliche Atome eine Bindungstasche bilden, wird ihr Vorkommen im Interface eines erzeugten Komplexes durch die Bewertungsfunktion bestraft.

Physiko-chemische Bewertung

Neben den rein geometrischen Kriterien können auch physiko-chemische Eigenschaften in den Tetraedern gespeichert und mittels einer Hashfunktion zur Bewertung verwendet werden. In dieser Arbeit wurden zwei Ansätze in dieser Richtung untersucht. Einerseits die Verwendung einer Funktion zur Behandlung der elektrostatischen Wechselwirkungen, wobei auf eine ähnliche Näherung wie in [34] zurückgegriffen wird. Und andererseits eine aminosäurenspezifische Gewich-

tungsfunktion, wie sie bei FFT-basierten Ansätzen z.B. in [70] und [41] zu finden ist.

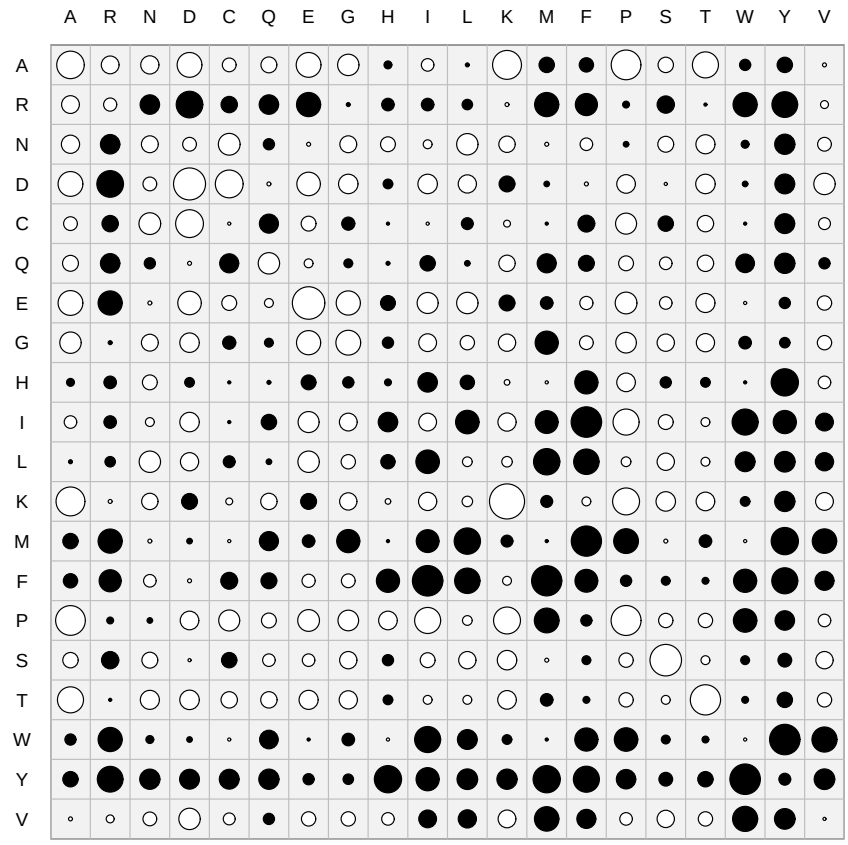


Abbildung 3.9: Visualisierung der aminosäurenspezifischen Gewichtungsfaktoren nach Moont et al. [70]. Zeilen und Spalten sind mit den einbuchstabigen Aminosäuren-Codes beschriftet. In der symetrischen Matrix entsprechen die Radien der Kreise den Gewichten. Weiße Kreise stehen für negative Werte, schwarze für positive. Der größte negative Wert ist -0.78 (K,K), der größte positive Wert ist 0.58 (W,Y).

Im Detail erfolgt die Berechnung der elektrostatischen Gewichtung dadurch, dass für alle Tetraeder des Rezeptors ein elektrisches Feld bestimmt wird, das durch

$$\phi_i = \sum_j \frac{q_j}{\epsilon(r_{ij}) r_{ij}} \tag{3.10}$$

beschrieben wird. Dabei ist ϕ_i das Feld im Schwerpunkt des Tetraeders i, q_j die Ladung des Atoms j und r_{ij} die Entfernung zwischen i und j. $\epsilon(r_{ij})$ ist eine von

der Entfernung abhängige dielektrische Funktion, die gemäß Hingerty et al. [42] durch

$$\epsilon(r_{ij}) = \begin{cases} 4 : r_{ij} \leq 6\text{Å} \\ 38r_{ij} - 224 : 6\text{Å} < r_{ij} < 8\text{Å} \\ 80 : r_{ij} \geq 8\text{Å} \end{cases} \tag{3.11}$$

gegeben ist. Die Ladungen werden wie in [34] beschrieben gewählt. Die elektrostatische Gewichtung berechnet sich damit als Summe aus den Produkten der Ladungen der Atome des Liganden und den entsprechenden Feldern der Tetraeder, in denen diese Atome in einem Komplex zu finden sind:

$$E_L = \sum_i \phi_{T(i)} \cdot q_i \tag{3.12}$$

$T(i)$ in Gleichung 3.12 bezeichnet die Funktion, die den Schwerpunkt des Tetraeders des Rezeptors liefert, in dem sich das Ligand-Atom i befindet. Energetisch günstige Konformationen zeichnen sich durch benachbarte Ladungen mit unterschiedlichem Vorzeichen aus. In diesen Fällen liefert Gleichung 3.12 einen großen negativen Wert, und entsprechen bei einer großen Anzahl nahe beieinander liegender gleichnamiger Ladungen einen großen positiven Wert. Das bedeutet, wenn man die Funktion als Score verwenden will, muss man noch das Vorzeichen des Gesamtergebnisses umdrehen. Alternativ verwendet man 3.12 als Filterkriterium und schließt Lösungen mit einem Wert oberhalb einer gewissen Schranke aus.

Wie zuvor bei der geometrischen Korrelation kann neben dem Score für den Liganden auch ein Score für den Rezeptor berechnet werden. Es müssen dazu analog die Rollen der beiden Proteine vertauscht werden. Auch die Unterscheidung zwischen Rand und Innerem der Proteine kann in die Bewertung eingebaut werden. Zudem kann man überlegen, ob die elektrostatische Bewertung parallel zur geometrischen erfolgen soll, und somit ein Gesamtscore berechnet wird, oder ob sie als *Postfilter* verwendet werden soll, d.h. im Anschluss an die geometrische Beurteilung auf die besten x Komplexe angewendet wird.

Die aminosäurenspezifische Gewichtung basiert auf der Überlegung, dass im Interface eines Proteinkomplexes manche Aminosäuren statistisch relevant häufiger in der Nähe von gewissen Aminosäuren vorkommen als andere. Daher sollte es

möglich sein, anhand der Auswertung existierender Komplexe eine Verteilung zu bestimmen, die in Form einer Gewichtungsfunktion für die Bewertung von künstlich erzeugten Anordnungen verwendet werden kann. Moont et al. [70] haben eine solche Statistik ermittelt und verwenden die daraus resultierenden Gewichtungsfaktoren in ihrer Software RPScore (Residue Level Pair Potential Scoring). Abbildung 3.9 veranschaulicht die von Moont et al. benutzten Gewichtungsfaktoren in Form einer Matrix. Die dahinter stehenden Zahlenwerte werden in der aminosäurenspezifischen Gewichtung von *AlphaDock* verwendet. Dabei wird, wie bei den anderen Bewertungen, einem Tetraeder ein Mittelwert der Faktoren der vier Eck-Atome zugewiesen und eine Summe über die Tetraeder gebildet, in denen sich bei einem Komplex Atome des Bindungpartners befinden:

$$A_L = \sum_i a(i, T(i)) \tag{3.13}$$

$a(i, T(i))$ ist hier die Funktion, die einem Atom i und dem i beinhaltenden Tetraeder $T(i)$ gemäß der entsprechend gewichteten aminosäurespezifischen Faktoren einen Score zuordnet.

3.3.3 Komplex-Weiterverarbeitung

Das Resultat des Algorithmus ist eine Liste von Transformationen des Liganden für die jeweils ein oder mehrere Scores nach den oben beschriebenen Verfahren berechnet wurden. Eine Weiterverarbeitung kann dann mit den ebenfalls in dieser Arbeit entwickelten Auswertungsroutinen erfolgen, oder die besten x Lösungen werden an ein anderes Programm übergeben, dass z.B. eine genauere Berechnung der physiko-chemischen Korrelation vornimmt und damit die Bewertung der Komplexe verfeinern kann. Ebenso wäre es möglich, die Ergebnisse als Startpunkt für eine molekulardynamische Simulationsrechnung zu verwenden. Wie groß x dabei zu wählen ist, hängt von der jeweils zu erzielenden Genauigkeit und der vorhandenen Rechenleistung ab.

Wie mit der Repräsentation der Proteine durch Alpha-Shapes weitergerechnet werden könnte, ist im Ausblick in Kapitel 5 dargestellt. Hier soll kurz erläutert werden, welche Auswertungsmöglichkeiten in dieser Darstellung bereits entwickelt wurden. Besonders hilfreich bei der Entwicklung von AlphaDock war die direk-

te Anbindung an den 3D-Viewer *geomview* [3]. Dieser erlaubt die unmittelbare Kontrolle der erzeugten Komplexe und macht Fehler in der Transformation des Liganden sofort sichtbar. Neben der automatisch ablaufenden und über Steuerdateien zu bedienenden Version von *AlphaDock* gibt es daher auch eine interaktive Variante, die mit einem grafischen Benutzerinterface zu bedienen ist und *AlphaDockGUI* genannt wird. Die Ergebnisse werden darin in einer Tabelle präsentiert, wobei jede Zeile zu einer Ligand-Transformation gehört und die Spalten die jeweils berechneten Scores enthalten. Zudem existiert eine Spalte mit dem RMSD, der die Qualität der Lösung widerspiegelt, wie es in Abschnitt 2.2.5 erläutert ist. Ein Doppelklick auf eine Zeile startet *geomview* und stellt darin den zugehörigen Komplex dar. Es existieren verschiedene Schalter in *AlphaDockGUI*, die das Hervorheben verschiedener Regionen und Merkmale des betrachteten Komplexes ermöglichen. So kann z.B. das Interface markiert werden oder die Tetraeder entsprechend der Größe der verschiedenen Gewichtungsfaktoren (Temperaturfaktor, aminosäurenspezifisches Gewicht, elektrisches Feld, usw.) eingefärbt werden. Weitere Einzelheiten zur Bedienung des Programms finden sich im Anhang.

Kapitel 4

Ergebnisse und Diskussion

In diesem Kapitel werden die mit dem Programm *AlphaDock* erzielten Ergebnisse vorgestellt. Dabei wird in der Reihenfolge des Entwicklungsprozesses und des Ablaufs des Programms vorgegangen. Das heißt zunächst werden die Ergebnisse der Analysen der Proteinstrukturen in Alpha-Shape-Darstellung vorgestellt. Anschließend wird der verwendete Komplex-Generator untersucht und die Resultate der Bewertungsfunktionen präsentiert. Darauf folgt eine Untersuchung des gesamten Docking-Prozesses unter Verwendung der besten Parameter, soweit sie im Rahmen dieser Arbeit optimiert werden konnten. Abschließend werden die Resultate diskutiert und die Vor- und Nachteile des verwendeten Ansatzes besprochen.

4.1 Testdatensatz

Nachfolgend sind die Testfälle des Benchmark 2.4 Datensatzes aufgeführt. Die Einteilung erfolgt, wie in [69] beschrieben, in drei Schwierigkeitsgrade. Die leichtesten Fälle werden als *Starre Körper* bezeichnet. Diese stellen mit 63 Komplexen auch den größten Teil des Datensatzes. Dazu kommen noch 13 Fälle *mittlerer* und 8 Fälle *hoher Schwierigkeit*. Je nach Art der Interaktion der Proteine, werden diese zudem noch in folgende Kategorien unterteilt, die in der Tabelle in der zweiten Spalte angegeben sind: Enzym/Inhibitor oder Enzym/Substrat (E), Antibody/Antigen (A), Antigen/Bound Antibody (AB) und Sonstige (O).

Tabelle 4.1: Protein-Protein Docking Benchmark 2.4 [69].

Komplex	Kat.	PDB	Protein1	PDB	Protein2
Starre Körper (63)					
1AVX	E	1QQU	Porcine Trypsin	1BA7	Soybean Trypsin Inhibitor
1AY7	E	1RGH	Barnase	1A19	Barstar
1BVN	E	1PIG	Alpha-Amylase	1HOE	Tendamistat
1CGI	E	2CGA	Bovine Chymotrypsinogen	1HPT	PSTI
1D6R	E	2TGT	Bovine Trypsin	1K9B	Bowman-Birk Inhibitor
1DFJ	E	9RSA	Ribonuclease A	2BNH	Rnase Inhibitor
1E6E	E	1E1N	Adrenoxin Reductase	1CJE	Adrenoxin
1EAW	E	1EAX	Matriptase	9PTI	BPTI
1EWY	E	1GJR	Ferredoxin Rreductase	1CZP	Ferredoxin
1EZU	E	1TRM	D102N Trypsin	1ECZ	Ecotin
1F34	E	4PEP	Porcine Pepsin	1F32	Ascaris Inhibitor 3
1HIA	E	2PKA	Kallikrein	1BX8	Hirustatin
1MAH	E	1J06	Acetylcholinesterase	1FSC	Fasciculin
1PPE	E	1BTP	Bovine Trypsin	1LU0	CMTI-1 Squash Inhibitor
1TMQ	E	1JAE	Alpha-Amylase	1B1U	RAGI Inhibitor
1UDI	E	1UDH	Uracyl-DNA Glycosylase	2UGI	Glycosylase Inhibitor
2MTA	E	2BBK	Methylamine Dehydrogenase	2RAC	Amicyanin
2PCC	E	1CCP	Cyt C Peroxidase	1YCC	Cytochrome C
2SIC	E	1SUP	Subtilisin	3SSI	Streptomyces Ssubtilisin Inhibitor
2SNI	E	1UBN	Subtilisin	2CI2	Chymotrypsin inhibitor 2
7CEI	E	1UNK	Colicin E7 Nuclease	1M08	Im7 Immunity Protein
1A2K	O	1QG4	Ran GTPase	1OUN	Nuclear Transport Factor 2
1AK4	O	2CPL	Cyclophilin	1E6J	HIV Capsid
1AKJ	O	2CLR	MHC Class 1 HLA-A2	1CD8	T-cell CD8 Coreceptor
1B6C	O	1D6O	FKBP binding Protein	1IAS	TGFbeta receptor
1BUH	O	1HCL	CDK2 Kinase	1DKS	Ckshs1
1E96	O	1MH1	Rac GTPase	1HH8	p67 Phox
1F51	O	1IXM	Sporulation Response Factor B	1SRR	Sporulation Response Factor F

Komplex	Kat.	PDB	Protein1	PDB	Protein2
1FC2	O	1BDD	Staphylococcus Protein A	1FC1	Human Fc Fragment
1FQJ	O	1TND	Gt-Alpha	1FQI	RGS9
1GCQ	O	1GRI	GRB2 C-ter SH3 Domain	1GCP	GRB2 N-ter SH3 Domain
1GHQ	O	1C3D	Complement C3	1LY2	Epstein-Barr virus receptor CR2
1HE1	O	1MH1	Rac GTPase	1HE9	Pseudomonas toxin GAP dom.
1I4D	O	1MH1	Rac GTPase	1I49	Arfaptin
1KAC	O	1NOB	Adenovirus Fiber Knob Protein	1F5W	Adenovirus Receptor
1KLU	O	1H15	MHC Class 2 HLA-DR1	1STE	Staphylococcus Enterotoxin C3
1KTZ	O	1TGK	TGF-Beta	1M9Z	TGF-Beta Rreceptor
1KXP	O	1IJJ	Actin	1KW2	Vitamin D Binding Protein
1ML0	O	1MKF	Viral Chemokine Binding P. M3	1DOL	Chemokine Mcp1
1QA9	O	1HNF	CD2	1CCZ	CD58
1RLB	O	2PAB	Transthyretin	1HBP	Retinol binding protein
1SBB	O	1BEC	T-Cell Receptor Beta	1SE4	Staphylococcus Enterotoxin B
2BTF	O	1IJJ	Actin	1PNE	Profilin
1AHW	A	1FGN	Fab 5g9	1TFH	Tissue Factor
1BVK	A	1BVL	Fv Hulys11	3LZT	HEW Lysozyme
1DQJ	A	1DQQ	Fab Hyhel63	3LZT	HEW Lysozyme
1E6J	A	1E6O	Fab	1A43	HIV-1 Capsid Protein P24
1JPS	A	1JPT	Fab D3H44	1TFH	Tissue Factor
1MLC	A	1MLB	Fab44.1	3LZT	HEW Lysozyme
1VFB	A	1VFA	Fv D1.3	8LYZ	HEW Lysozyme
1WEJ	A	1QBL	Fab E8	1HRC	Cytochrome C
2VIS	A	1GIG	Fab	2VIU	Flu Virus Hemagglutinin
1BJ1	AB	1BJ1	Fab	2VPF	vEGF
1FSK	AB	1FSK	Fab	1BV1	Birch Pollen Antigen Bet V1
1I9R	AB	1I9R	Fab	1ALY	Cd40 Ligand
1IQD	AB	1IQD	Fab	1D7P	Factor VIII Domain C2

Komplex	Kat.	PDB	Protein1	PDB	Protein2
1K4C	AB	1K4C	Fab	1JVM	Potassium Channel Kcsa
1NCA	AB	1NCA	Fab	7NN9	Flu Virus Neuraminidase N9
1NSN	AB	1NSN	Fab N10	1KDC	Staphylococcal Nuclease
1QFW	AB	1QFW	Fv	1HRP	Human Chorionic Gonadotropin
1QFW	AB	1QFW	Fv	1HRP	Human Chorionic Gonadotropin
2JEL	AB	2JEL	Fab Jel42	1POH	HPr
Mittlere Schwierigkeit (13)					
1ACB	E	2CGA	Chymotrypsin	1EGL	Eglin C
1IJK	E	1AUQ	Von Willebrand Factor Dom. A1	1FVU	Botrocetin
1KKL	E	1JB1	HPr Kinase C-ter Domain	2HPR	HPr
1M10	E	1AUQ	Von Willebrand Factor Dom. A1	1M0Z	Glycoprotein IB-alpha
1GP2	O	1GIA	Gi-Alpha	1TBG	Gi-Beta,Gamma
1GRN	O	1A4R	CDC42 GTPase	1RGP	CDC42
1HE8	O	821P	Ras GTPase	1E8Z	PIP3 Kinase
1I2M	O	1QG4	Ran GTPase	1A12	RCC1
1IB1	O	1QJB	14-3-3 Protein	1KUY	Serotonin N-Acteylase
1K5D	O	1RRP	Ran GTPase	1YRG	Ran GAP
1N2C	O	3MIN	Nitrogenase Mo-Fe Protein	2NIP	Nitrogenase Fe Protein
1WQ1	O	6Q21	Ras GTPase	1WER	Ras GAP
1BGX	A	1AY1	Fab	1CMW	Taq Polymerase
Hohe Schwierigkeit (8)					
1FQ1	E	1B39	CDK2 Kinase	1FPZ	CDK inhibitor 3
1ATN	O	1IJJ	Actin	3DNI	Dnase I
1DE4	O	1A6Z	Beta2-Microglobulin	1CX8	Transferrin Receptor Ectodom.
1EER	O	1BUY	Erythropoietin	1ERN	EPO receptor
1FAK	O	1QFK	Coagulation Factor VIIa	1TFH	Soluble Tissue Factor
1H1V	O	1IJJ	Actin	1D0N	Gelsolin
1IBR	O	1QG4	Ran GTPase	1F59	Importin Beta
2HMI	AB	2HMI	Fab 28	1S6P	HIV1 Reverse Transcriptase

4.2 Größe der Triangulationen

Um einen Eindruck von der Größe der zu verarbeitenden Datenmengen in der verwendeten Darstellung zu bekommen, sind in Tabelle 4.2 die Zahl der Tetraeder und Dreiecke der zugrunde liegenden Triangulationen der Alpha-Shapes der betrachteten Proteine zu finden. Zudem ist angeführt, aus wie vielen Atomen der Rezeptor und der Ligand bestehen, was der Zahl der Knoten der Triangulation entspricht. Im Hinblick auf die Idee der Bewertungsfunktion, die Atome innerhalb der konvexen Hülle des Bindungspartners bevorzugt, ist neben der Zahl der Interface-Atome in Klammern deren Anteil genannt, der sich innerhalb der konvexen Hülle des anderen Proteins befinden. Zum Interface werden alle Atome gezählt, die einen Abstand von weniger als sechs Ångström zu einem Atom des Bindungspartners besitzen.

Tabelle 4.2: Eigenschaften der den Alpha-Shapes zugrunde liegenden Triangulationen. Die Zahlen in Klammern geben den jeweiligen Anteil der Interface-Atome an, die sich in der konvexen Hülle des Bindungspartners befinden.

	Rezeptor					Ligand				
PDB	Atome	Tetraeder	Dreiecke	Interface		Atome	Tetraeder	Dreiecke	Interface	
1A2K	1979	12910	25877	127	(18)	1613	10649	21364	105	(45)
1ACB	1799	11809	23691	194	(35)	575	3575	7205	134	(72)
1AHW	3304	21741	43561	177	(45)	1622	10442	20933	196	(51)
1AK4	1258	8214	16491	161	(7)	1062	6862	13769	68	(49)
1AKJ	3082	20350	40771	121	(25)	1812	11937	23934	117	(77)
1ATN	2782	18358	36788	115	(19)	2034	13390	26854	138	(32)
1AVX	1642	10717	21514	166	(9)	1258	8083	16219	117	(56)
1AY7	746	4730	9514	114	(8)	719	4546	9147	119	(45)
1B6C	832	5274	10603	155	(56)	2629	17315	34692	160	(36)
1BJ1	3307	21703	43487	184	(18)	1527	9924	19918	151	(65)
1BUH	2366	15586	31251	123	(22)	650	4114	8268	118	(21)
1BVK	1744	11432	22946	92	(3)	1000	6466	12994	105	(12)
1BVN	3896	25752	51607	231	(14)	558	3458	6971	168	(145)
1CGI	1799	11804	23681	207	(15)	440	2737	5516	151	(85)
1D6R	1629	10626	21323	175	(9)	420	2614	5267	78	(53)
1DFJ	3411	22401	44896	256	(79)	951	6126	12310	232	(182)
1DQJ	3244	21173	42430	172	(12)	1000	6467	12996	176	(47)
1E6E	3505	23273	46619	143	(6)	819	5186	10419	149	(124)
1E6J	3277	21359	42803	124	(1)	573	3602	7250	79	(20)

	Rezeptor					Ligand				
PDB	Atome	Tetraeder	Dreiecke	Interface		Atome	Tetraeder	Dreiecke	Interface	
1E96	1378	8963	17986	110	(40)	1566	10232	20531	110	(1)
1EAW	1864	12147	24361	243	(19)	454	2831	5699	152	(116)
1EER	3228	21098	42258	294	(62)	1297	8411	16884	250	(228)
1EWY	2338	15488	31042	110	(0)	750	4752	9555	104	(98)
1EZU	2260	14765	29593	203	(109)	1662	10926	21937	316	(152)
1F34	2429	15888	31857	213	(111)	995	6399	12844	216	(62)
1F51	2489	16413	32897	147	(9)	933	5927	11910	143	(116)
1FAK	2658	17487	35039	212	(104)	1467	9490	19034	212	(192)
1FC2	3312	21899	43871	95	(13)	478	2962	5965	112	(26)
1FQ1	1399	9091	18246	90	(30)	2335	15438	30954	92	(9)
1FQJ	2540	16808	33687	173	(30)	1111	7209	14468	137	(91)
1FSK	3347	22094	44276	197	(11)	1230	7952	15966	123	(66)
1GCQ	1734	11348	22769	87	(12)	549	3388	6826	98	(24)
1GHQ	2317	15238	30553	67	(2)	996	6357	12771	65	(2)
1GP2	2491	16459	32977	108	(36)	3128	20603	41284	114	(13)
1GRN	1488	9666	19391	145	(36)	1501	9832	19733	151	(57)
1H1V	2782	18360	36792	460	(277)	5691	37766	75626	412	(217)
1HE1	976	6260	12580	161	(45)	1378	8967	17994	142	(51)
1HE8	6802	45651	91385	96	(0)	1326	8597	17264	102	(34)
1HIA	1799	11859	23787	246	(25)	352	2125	4287	105	(84)
1I2M	3000	19862	39799	260	(34)	1613	10651	21368	235	(128)
1I4D	3217	21217	42502	135	(16)	1378	8961	17982	142	(122)
1IB1	3673	24555	49187	298	(108)	1312	8469	17003	322	(273)
1IBR	1478	9750	19559	401	(374)	3440	22721	45518	425	(116)
1IJK	2079	13584	27233	124	(6)	1667	10883	21832	96	(57)
1IQD	3089	20134	40349	213	(42)	1268	8131	16313	133	(52)
1JPS	3241	21182	42448	174	(42)	1467	9489	19032	160	(54)
1K4C	3252	21211	42498	181	(11)	2822	18673	37401	114	(51)
1K5D	2741	18066	36208	188	(127)	2689	17682	35438	253	(39)
1KAC	1401	9008	18086	138	(18)	943	6019	12087	139	(4)
1KKL	3447	22826	45710	116	(1)	625	3923	7899	105	(54)
1KLU	3044	20107	40292	119	(9)	1903	12480	25024	113	(25)
1KTZ	890	5693	11437	67	(12)	827	5272	10598	81	(16)
1KXQ	3908	26213	52524	223	(20)	916	5813	11672	223	(135)
1M10	2080	13477	27031	193	(71)	1667	10880	21826	272	(102)
1MAH	4159	27641	55367	214	(35)	464	2897	5834	146	(76)
1ML0	5706	37747	75583	179	(29)	568	3574	7194	156	(108)
1MLC	3290	21765	43623	124	(6)	1000	6465	12992	99	(22)
1NCA	3329	22173	44426	181	(39)	3067	20225	40534	169	(36)

	Rezeptor					Ligand				
PDB	Atome	Tetraeder	Dreiecke	Interface		Atome	Tetraeder	Dreiecke	Interface	
1NSN	3282	21829	43733	151	(25)	1091	7046	14139	154	(48)
1PPE	1629	10702	21482	215	(19)	222	1284	2595	110	(63)
1QA9	1447	9317	18702	87	(0)	1398	8954	17972	82	(3)
1QFW	1762	11587	23239	149	(40)	1468	9518	19101	127	(36)
1RLB	3488	23143	46346	150	(19)	1411	9164	18383	123	(55)
1SBB	1826	11935	23927	109	(44)	1975	13005	26069	114	(14)
1TMQ	3598	23780	47645	199	(10)	880	5679	11415	175	(90)
1UDI	1826	11958	23987	182	(38)	654	4115	8275	195	(77)
1VFB	1741	11343	22759	121	(7)	1000	6481	13020	106	(18)
1WEJ	3318	21859	43794	119	(3)	823	5252	10558	105	(11)
1WQ1	2590	17148	34364	226	(36)	1368	8912	17891	207	(159)
2BTF	2782	18361	36794	149	(21)	1044	6730	13511	186	(63)
2JEL	3297	21620	43316	155	(3)	640	4014	8076	118	(62)
2MTA	3737	24838	49743	152	(0)	807	5153	10354	105	(57)
2PCC	2339	15524	31118	92	(0)	847	5446	10934	85	(7)
2QFW	1672	11006	22065	142	(25)	1468	9518	19101	102	(33)
2SIC	1938	12805	25681	191	(12)	772	4832	9720	105	(63)
2SNI	1932	12710	25495	185	(10)	521	3247	6534	87	(53)
2VIS	7416	49236	98557	94	(2)	3261	21473	43041	89	(5)
7CEI	1058	6819	13690	114	(30)	698	4454	8948	121	(20)

Bis auf wenige Ausnahmen befindet sich jeweils ein beträchtlicher Anteil der Interface-Atome des Liganden in der konvexen Hülle des Rezeptors. Im Durchschnitt sind es 43%. Der entsprechende Anteil der Rezeptoratome ist hingegen deutlich geringer, hier sind es im Mittel nur 17%, die in der konvexen Hülle des Liganden liegen.

4.3 Wahl der Parameter

Ziel der hier angeführten Analysen ist es, Abschätzungen für die im Algorithmus benutzen Parameter zu erhalten. Insbesondere geht es dabei um die Größe von Alpha zur Berechnung der detaillierten und der groben Oberfläche der Proteine, die zur Bestimmung der charakteristischen Erhebungen und Vertiefungen benutzt werden. Die Anzahl der ermittelten Stellen hängt von Alpha ab und be-

einflusst direkt die Zahl der durch den Generator erzeugten Komplexe. Es gilt einen guten Mittelweg aus Genauigkeit und Rechenaufwand zu finden. Dabei hilft auch die Untersuchung der Eigenschaften der detektierten kritischen Punkte im Interface und ihrer Abstandsbeziehungen. Zudem soll die Untersuchung der korrekt gedockten Komplexe, die im Datensatz [69] sowohl für den *bound* als auch den *unbound* Fall vorliegen, Aufschluss über die zu verwendenden Gewichte der geometrischen Bewertungsfunktion geben.

4.3.1 Alpha$_{grob}$ und Alpha$_{fein}$

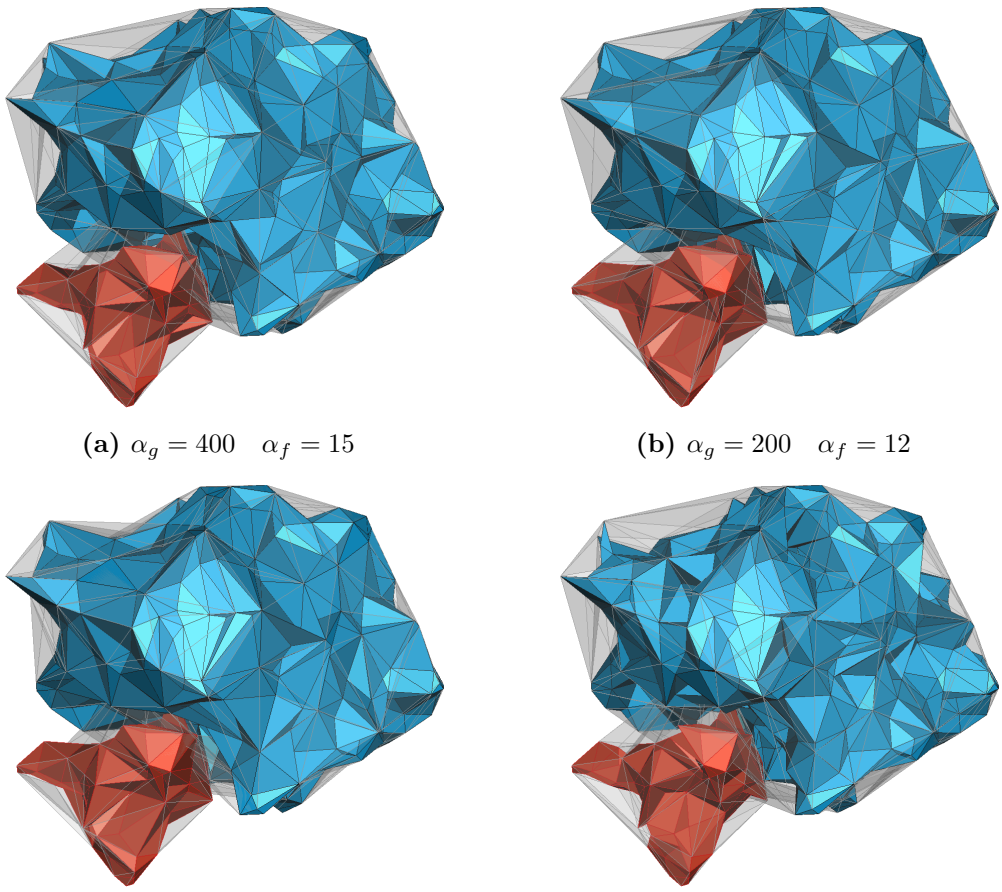
Abbildung 3.2 im letzten Kapitel hat bereits den Zusammenhang zwischen der Auflösung der Protein-Oberfläche und der Wahl von Alpha veranschaulicht. Um zu entscheiden, wie die beiden Werte für Alpha genau zu wählen sind, mit denen die grobe und die detaillierte Alpha-Shape festgelegt werden, ist eine nähere Untersuchung der dadurch jeweils generierten kritischen Punkte nötig. Es wurde dazu ermittelt, wie viele Paare von Erhebungen und Vertiefungen jeweils für unterschiedliche Werte von α_g und α_f im Interface der korrekt gedockten Komplexe gefunden werden, die den Bedingungen bei der Berechnung der Ligand-Transformationen entsprechen. D.h. es wurden jeweils zwei kritische Punkte im Interface des Rezeptors gewählt, die einen Abstand zwischen 4 und 12Å besitzen, und überprüft, ob im Interface des Liganden ein passendes Gegenstück-Paar zu finden ist, so dass die gegenüberliegenden Atome jeweils einen Abstand von weniger als 6Å besitzen. Liegt ein solches Gegensatz-Paar vor, bedeutet dies, dass der Komplex-Generator eine Transformation finden kann, die den Liganden im Interface positioniert. In Tabelle 4.3 sind die Ergebnisse zusammengestellt. Abbildung 4.1 zeigt den Komplex 1PPE für einen Teil der benutzten Parameter.

Tabelle 4.3: Anzahl der Gegensatz-Paare, die im Interface des korrekt gedockten Komplexes gefunden werden, die den Bedingungen des Generators entsprechen. Über den Spalten steht jeweils der Wert für α_g und α_f.

PDB	400/15	200/15	100/15	80/15	200/12	200/10	200/8
1A2K	35	84	81	67	236	116	12
1ACB	0	8	4	4	5	34	5
1AHW	192	170	31	18	273	138	285
1AK4	8	8	0	8	0	9	11

PDB	400/15	200/15	100/15	80/15	200/12	200/10	200/8
1AKJ	0	0	22	22	0	11	3
1ATN	80	107	60	74	137	167	63
1AVX	104	96	33	16	77	211	100
1AY7	33	63	37	21	148	421	143
1B6C	36	8	10	16	5	0	21
1BGX	43	56	95	89	69	49	29
1BJ1	12	11	3	3	30	65	66
1BUH	0	0	0	0	5	4	0
1BVK	55	48	24	21	24	73	38
1BVN	16	18	0	21	62	11	83
1CGI	5	2	0	0	0	2	0
1D6R	195	0	90	174	91	6	0
1DE4	36	21	24	17	155	138	145
1DFJ	93	121	184	202	72	121	53
1DQJ	22	87	57	14	45	61	30
1E6J	5	0	0	0	0	13	4
1E96	51	31	5	11	8	19	35
1EAW	2	0	0	0	116	93	17
1EER	194	186	141	56	270	317	154
1EWY	4	2	8	3	52	24	22
1EZU	133	66	47	101	163	56	155
1F34	21	72	46	77	135	66	22
1F51	60	46	61	103	80	68	22
1FAK	43	59	31	4	86	136	145
1FC2	118	113	84	26	342	111	154
1FQ1	11	13	3	0	11	14	2
1FQJ	0	17	5	0	27	70	4
1FSK	71	32	38	33	39	135	55
1GCQ	0	0	0	0	46	3	90
1GHQ	5	7	4	3	2	0	31
1GP2	40	17	9	2	24	107	108
1GRN	113	134	136	87	173	224	275
1H1V	59	34	17	16	43	19	30
1HE1	54	61	78	54	71	91	202
1HE8	4	0	0	0	4	28	22
1HIA	18	35	38	42	73	43	2
1I2M	103	66	58	65	36	51	157
1I4D	42	45	15	32	31	30	114
1I9R	47	31	4	0	32	83	104
1IB1	4	12	0	0	11	12	5

PDB	400/15	200/15	100/15	80/15	200/12	200/10	200/8
1IBR	3	3	6	3	37	40	29
1IJK	12	8	15	20	33	14	55
1IQD	116	124	126	183	24	33	7
1JPS	279	83	83	23	78	42	48
1K4C	36	33	4	2	47	79	14
1K5D	61	87	20	6	12	457	100
1KAC	3	2	0	0	8	7	13
1KKL	87	38	116	136	96	49	57
1KLU	19	6	6	16	43	13	19
1KTZ	71	53	0	0	106	107	437
1KXP	34	64	28	20	87	74	61
1KXQ	53	33	48	16	67	61	63
1M10	32	25	14	7	130	123	69
1MAH	0	0	0	0	6	2	2
1ML0	0	0	0	0	0	6	12
1MLC	14	17	7	7	22	163	284
1N2C	11	9	7	4	11	3	82
1NCA	82	64	58	54	41	60	75
1NSN	59	67	47	39	114	67	84
1PPE	0	10	0	54	333	0	4
1QA9	15	23	0	4	76	36	73
1QFW	20	27	38	28	28	5	4
1RLB	32	35	104	92	62	31	10
1SBB	21	19	22	12	79	115	26
1TMQ	75	98	68	38	352	373	198
1UDI	21	28	7	6	43	15	29
1VFB	2	2	2	2	15	71	61
1WEJ	20	5	2	2	31	96	289
1WQ1	9	22	36	23	53	36	3
2BTF	0	6	2	2	81	74	114
2HMI	0	3	0	2	0	0	0
2JEL	30	31	38	0	0	14	0
2MTA	7	41	13	13	25	30	15
2PCC	0	0	0	0	4	0	6
2QFW	139	106	102	91	618	379	145
2SIC	121	114	42	51	70	246	519
2SNI	4	4	19	19	2	0	15
2VIS	30	13	13	26	39	139	44
7CEI	192	184	149	148	179	113	101
Summe	3889	3485	2843	2668	6369	6630	6191



(a) $\alpha_g = 400 \quad \alpha_f = 15$ **(b)** $\alpha_g = 200 \quad \alpha_f = 12$

(c) $\alpha_g = 100 \quad \alpha_f = 15$ **(d)** $\alpha_g = 200 \quad \alpha_f = 8$

Abbildung 4.1: Darstellung der groben und detailierten Alpha-Shapes von 1PPE für unterschieliche Werte von α_{grob} und α_{fein}.

Die meisten passenden Paare werden bei einer Kombination aus $\alpha_g = 200$ und $\alpha_f = 10$ gefunden. Eine weitere Verfeinerung der detaillierten Alpha-Shape $\alpha_f = 8$ bringt keine Vorteile. Grundsätzlich ist noch zu bemerken, dass auch wenn auf die oben beschriebene Weise kein Paar gefunden wird, nicht zwangsläufig keine gute Lösung durch den Generator erzeugt werden kann. Es kann auch Fälle geben, in denen außerhalb des Interfaces liegende kritische Punkte eine ausreichende Bedingung liefern.

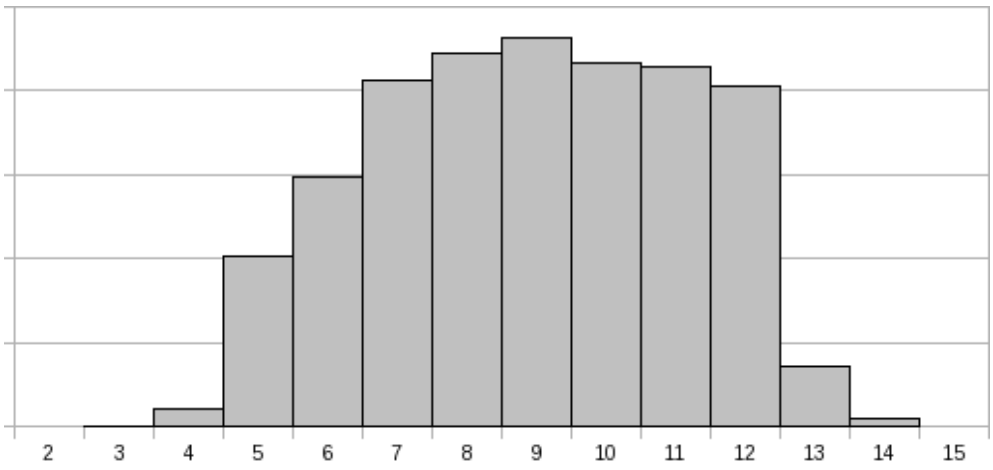


Abbildung 4.2: Histogramm der Abstände in Å zweier kritischer Punkte, die ein passendes Gegenstück im Interface des Bindungspartners besitzen.

4.3.2 Abstand kritischer Punkte

Bei der Berechnung von Ligand-Transformationen mit Hilfe von zwei kritischen Punkten pro Bindungspartner ist der Wertebereich, in dem die Abstände der Punkte liegen dürfen ein wichtiger Parameter, der die Anzahl und die Qualität der generierten Komplexe beeinflusst. Wie dieser Bereich zu wählen ist, wurde anhand einer Auswertung der Interfaces gedockter Proteine bestimmt. Dabei wurden, wie zuvor bei der Untersuchung der Oberflächen für verschiedene Alpha im letzten Abschnitt, nur die Abstände zweier kritischer Punkte berücksichtigt, die auch ein entsprechendes Gegenüber im Interfaces des anderen Proteins haben, d.h. zwei komplementäre kritische Punkte, die einen vergleichbaren Abstand voneinander besitzen. Abbildung 4.3 zeigt das daraus erzeugte Histogramm, welches die relativen Häufigkeiten der Abstände in Ångström enthält. Man sieht, dass bei einer Wahl des Abstandes zwischen 7 und 12Å der größte Teil der interessanten Paare gefunden werden kann. Weiterhin wurde untersucht, welche Differenz die Abstände der gemäß der Generatorvorgabe zueinander passenden Punktepaare im Interface besitzen. Das entsprechende Histogramm ist in Abbildung 4.3 gezeigt. Die Verteilung erlaubt keine Eingrenzung des Wertebereichs. Sowohl sehr geringe Abweichungen als auch größere Differenzen bis zu 1,5Å tauchen häufig auf. Es existiert kein eindeutiges Maximum, um das man den Wert herum wählen könnte. Es wird daher eine Abweichung von bis zu 1,5Å zugelassen.

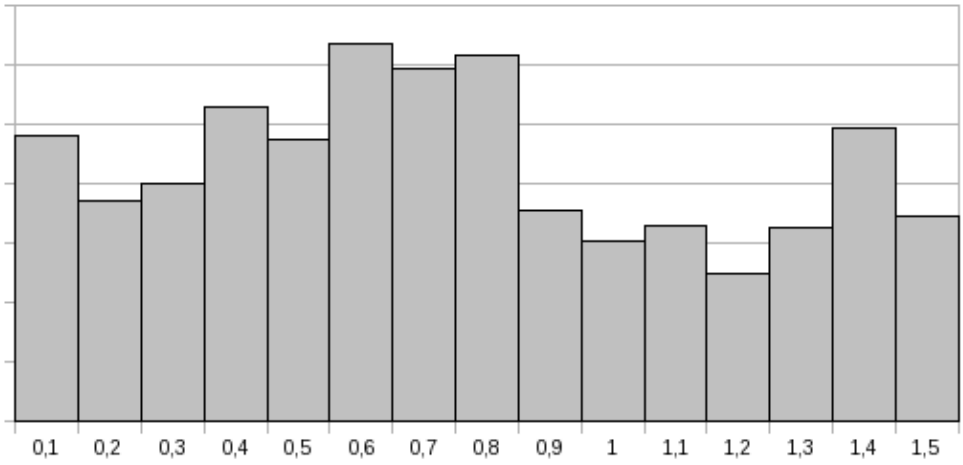


Abbildung 4.3: Histogramm der Abstandsdifferenz in Å zweier kritischer Punktepaare im Interface.

4.3.3 Minimale Tiefe

Bei der Bestimmung eines Schwellwerts für die Mindesttiefe einer Vertiefung, die als kritischer Punkt ausgewählt wird, hilft das Histogramm der Verteilung der mittleren Tiefen der Atome der Proteine im gedockten Zustand. Auch hierbei wird wie zuvor als Nebenbedingung für die betrachteten Atome verlangt, dass ein Atom des Bindungspartners in einem Abstand von weniger als sechs Ångström existiert. Da die Tiefe von der Wahl von Alpha für die grobe und die detaillierte Oberfläche abhängen, wurde die Verteilung der mittleren Tiefen für unterschiedliche Werte betrachtet. Die resultierenden Verteilungen haben qualitativ den gleichen Verlauf. In Abbildung 4.4 und 4.5 sind die Histogramme der mittleren Tiefen in Å im Interface für alle Fälle des Testdatensatzes gezeigt. Gezählt wurden nur Tiefen größer Null. Die erste Abbildung gibt einen Überblick über alle vorhandenen Tiefen. In der zweiten Abbildung wird der Bereich bis 4Å detaillierter aufgelöst. Es zeigt sich, dass der größte Teil der Atome nur eine mittlere Tiefe von knapp unter einem Ångström besitzt. Um nicht zu viele kritische Punkte zu erhalten und damit sehr viele Komplexe zu generieren, sollte eine Mindesttiefe von deutlich über einem Ångström gewählt werden. Es zeigt sich, dass bei einem Wert von 1,5Å noch mehrere Millionen Transformationen generiert werden und deshalb ein Wert ab 2Å verwendet werden sollte. Wie man in Abbildung 4.5 erkennt, wird damit noch ein ausreichend große Anzahl von Vertiefungen im Interface gefunden.

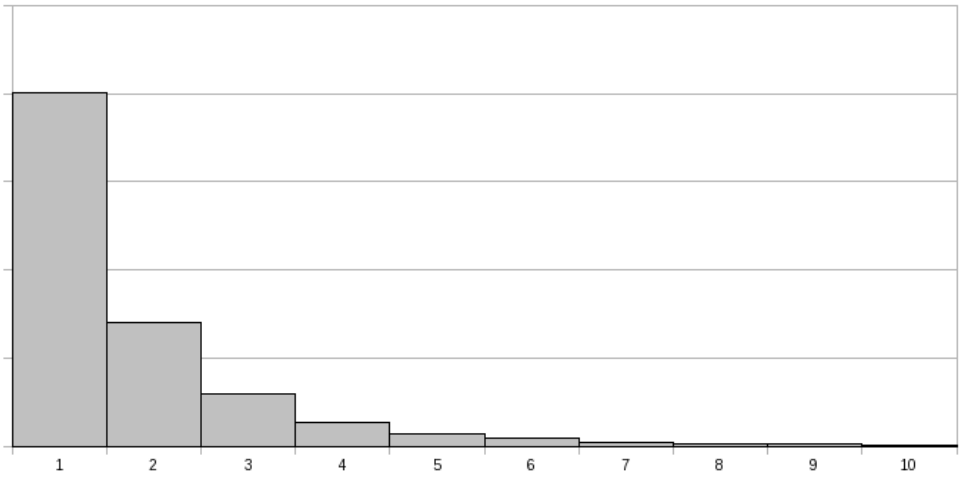


Abbildung 4.4: Histogramm der mittleren Tiefen von Atomen im Interface in Å, Wertebereich zwischen 0,1 und 10Å.

In Abschnitt 4.4 werden die tatsächlich produzierten Komplexe weiter untersucht und die Gesamtzahlen der erhaltenen Lösungen angegeben.

4.3.4 Geometrische Gewichtungsfaktoren

Nachdem der Wertebereich von Alpha abgesteckt wurde, gilt es noch, die Gewichte der geometrischen Bewertungsfunktion zu ermitteln. Dazu wurde die Besetzung der in Abschnitt 3.2.1 definierten Tetraeder-Klassen (*intern, extern, interner Rand, externer Rand*) bei korrekt angeordneten Proteinen untersucht und mit der verglichen, die bei künstlich erzeugten, nicht korrekt ausgerichteten Komplexen vorliegt. Eine noch genauere Einteilung der Tetraeder kann dadurch erfolgen, dass man die Position der vier Seiten des Tetraeders innerhalb der Alpha-Shape betrachtet. D.h. es wird geprüft, ob die Seite innerhalb (I), außerhalb (E) oder auf dem Rand (R) der Alpha-Shape liegt. Mit den genannten Abkürzungen ergeben sich die in Tabelle 4.4 aufgeführten Klassen. Es sind die im Algorithmus verwendeten Gewichte angegeben. Die Werte werden bisher für alle Protein-Klassen gleich gesetzt und sind noch nicht optimiert.

Hervorzuheben ist, dass auch ein Tetraeder, das keine Seite auf dem Rand der Alpha-Shape besitzt, einen kleinen positiven Wert erhält. Der Grund dafür ist, dass auch diese Tetraeder innerhalb der konvexen Hülle des betrachteten Pro-

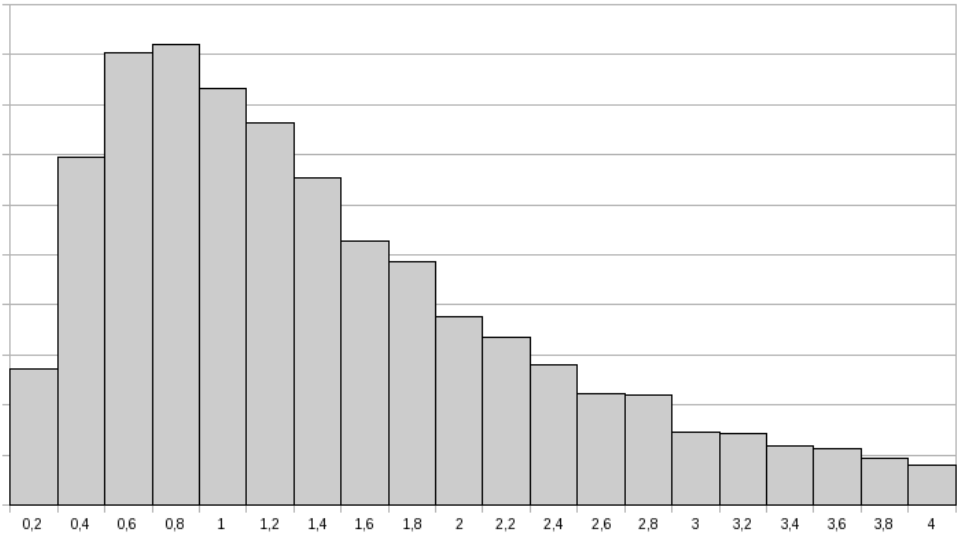


Abbildung 4.5: Histogramm der mittleren Tiefen von Atomen im Interface in Å, Wertebereich zwischen 0,1 und 4Å.

Tabelle 4.4: Tetraeder-Klassen und deren Gewichte bei der geometrische Bewertung. Die Erläuterung zu den benutzen Klasseneinteilungen findet sich im Text.

Tetraeder	Gewicht	Tetraeder	Gewicht	Tetraeder	Gewicht
IIII	-4	RRRE	3	AAAV	1
IIIR	-3	RREE	2	AAVV	0.5
IIRR	-3	REEE	2	AVVV	0.5
IRRR	-2	EEEE	0.5	VVVV	0

teins liegen und somit Atome des Bindungspartners darin zu begünstigen sind. Liegt ein Atom außerhalb der konvexen Hülle, wird geprüft, ob es im Rand der erweiterten Alpha-Shape (vgl. Abschnitt 3.3.1) liegt. Ist das der Fall, wird der Score abhängig davon vergeben, wie viele virtuelle (V) und wie viele reale Atome (A) die Ecken des betreffenden Tetraeders bilden. Je mehre reale Atome beteiligt sind, desto höher ist der Wert. Die zur Zeit verwendeten Gewichte finden sich in der dritten Spalte der Tabelle 4.4. Atome, die weder in der Alpha-Shape noch in der erweiterten Alpha-Shape liegen, bekommen keinen Score.

4.4 Erzeugte Komplexe

Im vorhergehenden Abschnitt wurde anhand der vorliegenden korrekt gedockten Komplexen gezeigt, dass die Voraussetzungen erfüllt sind, um mit dem im letzten Kapitel vorgeschlagenen Ansatz nahe native Komplexe zu erzeugen. In Tabelle 4.5 und 4.6 finden sich die Ergebnisse für den kompletten Datensatz der gebundenen und der ungebundenen Strukturen unter der Wahl von Parametern bei der Erzeugung, die einen Mittelweg aus Verallgemeinerbarkeit, Qualität und Geschwindigkeit darstellen.

In diesem Fall wurde $\text{Alpha}_{grob} = 80$ und $\text{Alpha}_{fein} = 15$ verwendet. Zur Komplexbildung wurden kritische Punkte mit einem Abstand zwischen 6 und 9Å benutzt und eine Abweichung von bis zu einem Ångström beim den komplementären Punkten des Bindungspartners erlaubt. Vertiefungen wurden ab einer mittleren Tiefe von 2Å als kritischer Punkt gewertet. In den Tabellen ist jeweils die Anzahl der insgesamt erzeugten Komplexe angegeben und wie viele davon einen RMSD von weniger als 2,5, 5 und 10Å besitzen. Zusätzlich ist jeweils der kleinste RMSD aufgeführt, der für jeden Datensatz erreicht wird. Die zugehörige Transformation charakterisiert jeweils die bestmögliche Lösung, die mit diesem Generator unter den gegebenen Parametern erzielt werden kann.

Hervorgehoben sind jeweils die Datensätze, für die keine Lösung mit einem RMSD unter 5Å erzeugt wurde. Bis auf 1GCQ für beide Datensätze, 1B6C für den gebundenen und 1AHW für den ungebundenen Fall hat die bestmögliche Struktur, aber auch in diesen Fällen einen RMSD unterhalb von 6Å. Das schlechteste Ergebnis ist mit einem minimalen RMSD von 7,69Å nicht sehr weit von einem akzeptablen Wert entfernt.

Tabelle 4.5: Untersuchung der Qualität der erzeugten Strukturen unter Verwendung der *gebundenen* Strukturen des Rezeptors und des Liganden als Eingabe. Betrachtet werden jeweils die besten 2500 Komplexe. Grau unterlegt sind die Fälle, in denen keine Lösung mit einem RMSD $\leq$ 5Å gefunden werden. Die verwendeten Parameter sind im Text erläutert.

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	$\leq$ 2,5	$\leq$ 5	$\leq$ 10		
1A2K	6	99	2229	2,06	1.379.644
1ACB	0	127	2500	3,05	403.777

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	≤ 2,5	≤ 5	≤ 10		
1AHW	0	10	486	3,25	1.679.872
1AK4	0	17	2386	3,17	637.537
1AKJ	0	9	728	3,86	1.816.690
1ATN	1	50	892	2,34	1.371.502
1AVX	2	45	1651	1,90	603.793
1AY7	9	151	2500	1,45	172.573
1B6C	0	0	221	6,49	500.854
1BJ1	0	74	1511	2,54	1.587.451
1BUH	0	0	82	5,21	257.323
1BVK	0	6	762	3,61	282.223
1BVN	0	8	712	2,67	388.963
1CGI	2	41	1449	2,12	374.716
1D6R	1	19	1310	2,40	178.729
1DFJ	0	25	1097	2,97	1.474.756
1DQJ	0	9	158	2,54	574.105
1E6E	0	3	387	4,02	1.100.950
1E6J	0	50	2500	2,86	1.683.673
1E96	0	6	675	3,03	1.026.985
1EAW	2	142	2500	2,43	315.871
1EER	1	24	1168	2,35	1.957.846
1EWY	0	4	572	4,11	305.296
1EZU	0	1	287	4,91	877.243
1F34	0	14	310	3,20	997.639
1F51	0	21	614	2,61	582.247
1FAK	0	9	182	3,92	925.819
1FC2	0	43	1965	3,25	402.052
1FQ1	0	2	238	4,66	989.203
1FQJ	0	1	643	4,88	1.171.120
1FSK	5	33	1663	2,16	1.423.816
1GCQ	0	0	11	7,69	20.212
1GHQ	0	15	1930	3,76	729.964
1GP2	0	0	69	5,18	2.524.819
1GRN	0	33	898	2,62	1.662.715
1H1V	0	2	192	4,38	2.332.957
1HE1	2	35	1703	2,14	702.616
1HE8	0	16	1237	2,82	2.565.100
1HIA	2	79	2500	1,86	258.667
1I2M	0	19	791	3,68	1.359.226
1I4D	0	9	643	3,64	2.654.890

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	≤ 2,5	≤ 5	≤ 10		
1IB1	0	25	607	2,85	1.538.461
1IBR	0	1	643	4,99	1.877.290
1IJK	0	19	2134	3,19	934.651
1IQD	1	8	305	2,45	747.823
1JPS	0	5	434	2,77	1.652.122
1K4C	0	14	1378	2,88	675.940
1K5D	0	8	404	2,76	2.607.805
1KAC	0	2	160	3,35	203.437
1KKL	1	98	1519	1,76	1.137.121
1KLU	0	8	891	2,70	1.466.356
1KTZ	0	17	724	2,52	145.870
1KXQ	0	18	393	3,50	421.741
1M10	0	7	342	4,29	826.756
1MAH	0	0	91	5,82	233.890
1ML0	1	41	1204	2,13	1.741.933
1MLC	8	82	546	1,82	752.062
1NCA	3	7	254	1,93	2.423.689
1NSN	0	29	1046	2,86	1.933.222
1PPE	5	272	2500	1,88	178.642
1QA9	1	7	435	2,21	270.940
1QFW	0	1	315	4,99	860.701
1RLB	0	11	1300	3,71	1.814.272
1SBB	3	80	1297	2,06	1.184.182
1TMQ	0	24	1197	2,57	883.135
1UDI	0	0	277	5,16	316.345
1VFB	2	20	958	2,281	478.993
1WEJ	0	1	110	4,44	853.813
1WQ1	0	41	1468	3,20	1.379.587
2BTF	0	8	655	4,03	955.840
2JEL	3	35	500	2,41	662.566
2MTA	2	17	631	2,33	534.268
2PCC	0	0	405	5,26	840.490
2QFW	0	135	1963	2,61	1.012.561
2SIC	5	182	2500	1,81	255.592
2SNI	6	94	2179	2,22	189.043
2VIS	2	27	1620	2,16	1.453.960
7CEI	1	72	1926	2,41	442.495

Tabelle 4.6: Untersuchung der Qualität der erzeugten Strukturen unter Verwendung der *ungebundenen* Strukturen des Rezeptors und des Liganden als Eingabe. Betrachtet werden jeweils die besten 2500 Komplexe. Grau unterlegt sind die Fälle, in denen keine Lösung mit einem RMSD $\leq$ 5Å gefunden werden. Die verwendeten Parameter sind im Text erläutert.

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	$\leq$ 2,5	$\leq$ 5	$\leq$ 10		
1A2K	0	10	1246	3,17	1.129.773
1ACB	0	0	202	5,40	231.857
1AHW	0	0	74	7,65	1.095.249
1AK4	0	35	1681	2,97	305.325
1AKJ	0	25	1679	3,70	1.591.121
1ATN	3	32	1112	1,68	1.029.097
1AVX	3	14	654	1,77	280.633
1AY7	0	26	550	3,57	78.109
1B6C	2	16	382	1,66	460.481
1BJ1	1	173	2500	2,26	867.025
1BUH	0	2	315	3,90	385.841
1BVK	0	1	309	4,55	184.461
1BVN	0	8	288	2,88	233.441
1CGI	0	7	611	4,23	298.673
1D6R	0	3	583	3,93	215.109
1DFJ	0	17	483	4,00	669.797
1DQJ	0	4	241	4,26	613.273
1E6E	0	4	326	4,17	638.633
1E6J	0	19	1297	3,52	425.749
1E96	0	11	1178	3,88	554.537
1EAW	0	99	2192	2,85	224.437
1EER	0	3	282	3,30	2.690.853
1EWY	0	6	782	4,15	230.321
1EZU	1	39	863	2,37	812.385
1F34	0	2	147	4,28	691.481
1F51	0	1	907	4,45	743.909
1FAK	0	15	1481	3,12	1.127.453
1FC2	0	3	1575	4,43	612.233
1FQ1	0	16	957	2,86	595.529
1FQJ	0	8	682	2,89	1.106.505
1FSK	0	17	1206	3,70	1.012.053
1GCQ	0	0	46	6,06	175.601
1GHQ	0	1	859	4,52	356.245

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	≤ 2,5	≤ 5	≤ 10		
1GP2	0	1	611	4,94	1.298.697
1GRN	0	1	728	4,92	943.681
1H1V	0	0	297	5,01	3.167.753
1HE1	0	8	656	3,69	353.645
1HE8	0	10	920	2,71	1.857.733
1HIA	1	9	775	2,32	164.969
1I2M	1	18	606	1,98	1.585.605
1I4D	0	2	175	3,94	1.715.013
1IB1	1	21	398	2,40	1.065.557
1IBR	0	0	90	5,08	2.040.057
1IJK	0	4	886	4,55	868.713
1IQD	0	2	357	3,94	624.289
1JPS	0	1	318	4,94	1.000.057
1K4C	0	5	331	3,44	1.893.185
1K5D	0	17	1120	3,95	2.018.001
1KAC	0	0	193	5,67	158.369
1KKL	0	7	496	3,99	478.557
1KLU	0	2	698	4,56	1.640.393
1KTZ	0	0	254	5,72	162.981
1KXQ	0	45	528	2,85	433.269
1M10	0	0	303	5,32	643.737
1MAH	0	0	61	5,87	265.133
1ML0	0	19	1187	3,38	1.922.669
1MLC	0	29	838	2,71	667.909
1NCA	0	1	179	4,52	1.852.401
1NSN	0	11	475	3,14	1.093.529
1PPE	0	7	1321	3,84	122.837
1QA9	0	0	230	5,77	486.449
1QFW	0	2	265	4,79	742.329
1RLB	0	30	1178	3,28	761.013
1SBB	0	7	194	3,47	1.025.045
1TMQ	0	2	300	4,27	381.985
1UDI	0	3	295	4,14	331.561
1VFB	1	12	898	2,00	303.805
1WEJ	0	0	1084	5,46	743.185
1WQ1	0	55	1189	2,58	1.303.081
2BTF	0	9	421	3,61	572.109
2JEL	0	11	426	3,28	404.133
2MTA	8	86	1054	1,91	564.117

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	≤ 2,5	≤ 5	≤ 10		
2PCC	0	1	383	4,75	364.029
2QFW	1	49	1696	2,28	1.031.593
2SIC	3	34	1050	1,57	83.677
2SNI	0	45	931	3,46	99.425
2VIS	1	8	958	2,3	4.104.941
7CEI	0	20	1294	3,25	514.689

Wählt man die Parameter bei der Erzeugung so, dass mehr Komplexe gebildet werden, kann man für alle Datensätze mindestens einen Komplex im nahen nativen Bereich produzieren. Dies wurde speziell für die zuvor erfolglosen Fälle mit folgenden Werten unternommen: der Abstand zweier kritischer Punkte wurde zwischen 4 und 12Å gewählt, bei einer zulässigen Abweichung von 1,5Å für das Komplementär. Vertiefungen wurden schon ab 1,5Å gezählt. Die Ergebnisse finden sich in Tabelle 4.7. Man sieht, dass die Zahl der erzeugten Komplexe unter diesen Bedingungen stark ansteigt. Es werden zum Teil mehrere Millionen Transformationen berechnet, die alle im Bewertungsschritt untersucht werden müssten. Die Berechnung der Transformationen an sich dauert hingegen selbst in dieser Größenordnung keine Minute.

Tabelle 4.7: Erzeugung von Komplexen für die zuvor erfolglosen Fälle mit angepassten Parametern. Die genauen Werte finden sich im Text.

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	≤ 2,5	≤ 5	≤ 10		
1ACB	0	19	1294	3,27	775.981
1AHW	1	30	920	1,81	4.440.250
1B6C	1	41	1610	2,22	1.393.801
1BUH	0	27	1868	2,51	1.530.946
1GCQ	0	62	1070	2,79	748.660
1GP2	0	7	2200	3,28	6.383.908
1H1V	0	7	769	3,70	7.288.783
1IBR	0	3	298	4,20	5.676.943
1KTZ	0	30	1813	2,59	1.723.486
1M10	0	11	1046	3,89	2.515.924
1MAH	1	139	2500	2,28	1.668.175
1QA9	0	2	1129	4,96	3.535.732

PDB	Anzahl RMSD (Å)			Min. RMSD (Å)	Komplexe
	$\leq 2{,}5$	≤ 5	≤ 10		
1UDI	0	42	1095	2,59	1.294.210
1WEJ	1	50	1932	2,36	1.283.425
2PCC	0	32	1917	2,97	1.745.146

4.5 Bewertung der Komplexe

Nachdem die Wertebereiche für die Parameter in den vorangegangenen Schritten abgesteckt wurden, kann jetzt der komplette Algorithmus auf den Datensatz angewendet und die Platzierung der nahen nativen bzw. besten Lösungen ermittelt werden. Eine genaue Auflistung, welche Parameter bei der Erzeugung und Beurteilung von Proteinkomplexen im Programm *AlphaDock* eingestellt werden können, findet sich im Anhang im Abschnitt A.3.

Die Bewertung erfolgt schrittweise und wird abgebrochen, sobald ein Teilwert den entsprechenden Schwellenwert über- bzw. unterschreitet. Zunächst werden die mittels der Hashfunktion schnell zu bestimmenden geometrischen Scores berechnet und nur wenn diese oberhalb eines geforderten Wertes liegen, die weiteren Funktionen bemüht. Als weiteres Abbruchkriterium bei der ersten Bewertung wird ein Zähler eingeführt, der die Atome im Inneren der Alpha-Shape des Bindungspartners zählt. Sobald dieser das eingestellte Limit überschreitet, wird die Berechnung beendet und die Lösung verworfen.

4.5.1 Gebundene Komplexe

Für alle Testfälle mit den gebundenen Strukturen sind in der kommenden Tabelle folgende Ergebnisse aufgeführt: Die beste Platzierung einer Lösung mit einem RMSD unter fünf Ångström und der zugehörige RMSD. Weiterhin ist die Position und der RMSD des gefundenen Komplexes mit der geringsten Abweichung von der korrekten Anordnung angegeben. Unterschieden werden die Ergebnisse, die bei Anwendung der geometrischen Bewertung alleine (GeoScore) und zusammen mit der Aminosäuren-spezifischen Gewichtung erzielt werden (GeoScore + AminoScore). Die Lösung mit dem besseren Rang ist jeweils grau unterlegt.

Tabelle 4.8: Ergebnis des Docking-Prozesse mit den gebundenen Strukturen als Testfälle. Hervorgehoben sind die jeweils besten Lösungen im Vergleich der Bewertung mit Aminosäuren-spezifischen Gewichten und ohne. Die benutzten Parameter und weitere Erklärung finden sich im Text.

	GeoScore + AminoScore				GeoScore	
	Erste n.n. Struktur		beste Struktur		Erste n.n. Struktur	
PDB	Rang	RMSD	Rang	RMSD	Rang	RMSD
1A2K	804	3,01	5022	1,51	673	3,86
1ACB	8	3,46	65	2,49	101	3,14
1AHW	1951	2,87	1951	2,87	36921	4,45
1AK4	3166	4,59	7490	3,11	101605	4,67
1AKJ	71	3,32	71	3,32	248	4,99
1ATN	734	2,49	734	2,49	4175	4,41
1AVX	10	3,74	181	3,70	81	3,81
1AY7	18	2,78	46	2,17	334	3,89
1B6C	605	3,79	605	3,79	1107	3,79
1BJ1	1096	2,21	1096	2,21	10974	2,21
1BGX	109	4.32	109	4.32	3736	4.32
1BVK	22266	3,67	22266	3,67	94929	3,67
1BVN	1	2,68	1	2,68	1	2,68
1CGI	4	2,12	4	2,12	44	2,29
1D6R	543	4,01	3950	2,89	69	4,35
1DE4	5607	4.74	29567	3.06	48567	4.74
1DFJ	564	2,98	564	2,98	1851	3,53
1DQJ	4136	4,83	4136	4,83	14598	4,51
1E6E	60	3,33	60	3,33	305	3,63
1E6J	9046	3,61	9046	3,61	30769	3,6
1E96	7028	3,21	7028	3,21	35255	3,21
1EAW	37	3,20	901	1,41	5	2,66
1EER	81	3,84	109	2,15	150	3,84
1EWY	18	3,93	211	2,51	36	4,73
1EZU	29	2,02	29	2,02	993	3.03
1F34	10	3,19	10	3,19	495	4,83
1F51	53	4,80	341	2,23	47	4,94
1FAK	1253	4,03	1253	4,03	27571	4.80
1FC2	6925	3,59	8705	2,53	19385	4,46
1FQ1	65	4,94	65	4,94	1900	4,94
1FSK	1454	4,38	2847	4,32	4681	4.48
1GHQ	11290	4,43	11290	4,43	9769	4,43
1GRN	1148	2,72	1148	2,72	2964	3,82

	GeoScore + AminoScore				GeoScore	
	Erste n.n. Struktur		beste Struktur		Erste n.n. Struktur	
PDB	Rang	RMSD	Rang	RMSD	Rang	RMSD
1H1V	1060	4,42	32525	4,42	2116	4,42
1HE1	1	3,53	16	1,96	6546	4,87
1HE8	33994	3,05	81417	3,05	32991	3,05
1HIA	9	3,37	85	2,91	3	3,62
1I2M	57	4,60	485	2,22	24	4,33
1I4D	667	4,05	667	4,05	7407	4,90
1I9R	291	3,73	291	3,73	857	3,97
1IB1	280	3,41	459	2,95	2056	4.58
1IBR	5989	4,51	5989	4,51	5238	4,67
1IJK	131	4,67	6304	4,67	19043	4,96
1IQD	14964	4,75	14964	4,75	22793	4,86
1JPS	2327	2,72	2444	1,74	21132	4,28
1K4C	4578	3,99	4578	3,99	6220	4,07
1K5D	1389	4,07	40158	4,07	8446	3,35
1KAC	6475	3,35	6475	3,35	59	4,63
1KKL	2	3,11	84	2,05	3485	4,83
1KLU	12096	3,02	16102	2,7	3642	3,52
1KTZ	924	3,52	6059	3,52	29	4,72
1KXP	7	4.88	776	3.09	332	4.88
1KXQ	5	4,63	5	4,63	257	4,72
1M10	71	4,62	71	4,62	2825	3,96
1MAH	1895	3,96	55947	3,96	12175	4,44
1ML0	2385	2,14	4025	2,14	23527	2,87
1MLC	12596	2,87	45104	2,87	23564	2,29
1NCA	12075	2,29	12075	2,29	7047	3.01
1NSN	480	3.00	480	3.00	18	3,48
1PPE	8	2,95	81	1,51	966	4,67
1QA9	16	2,79	16	2,79	449	5.00
1QFW	2654	4,87	2654	4,87	20786	4,36
1RLB	7295	4,61	10796	2,63	42841	3,58
1SBB	159	3,42	9650	2,12	12	2,60
1TMQ	13	2,17	53	1,63	13	3.70
1UDI	326	5.08	4,51	1716	1704	5.08
1VFB	7901	3,28	7901	3,28	16860	3,28
1WEJ	3677	3,96	3677	3,96	118	3,98
1WQ1	150	3,9	1274	3,05	15404	4,25
2BTF	2710	4,25	2710	4,24	32888	4,78
2JEL	3345	4,78	7238	3,49	29826	4,29

	GeoScore + AminoScore				GeoScore	
	Erste n.n. Struktur		beste Struktur		Erste n.n. Struktur	
PDB	Rang	RMSD	Rang	RMSD	Rang	RMSD
2MTA	6211	3.49	22973	3,72	3605	3.49
2QFW	1185	3,35	9342	2,40	18865	4,14
2SIC	4	2,28	17	2,17	61	2,28
2SNI	1	3,57	28	1,62	31	4,12
2VIS	2477	3,93	2600	3,25	13479	4,62
7CEI	279	3,71	1877	3,56	290	4,51

Der Datensatz wurde mit unterschiedlichen Parametersätzen gedockt. In der Tabelle 4.8 finden sich die Daten für einen festen Parametersatz, der einen guten Kompromiss aus Verallgemeinerbarkeit und Qualität darstellt. Konkret wurden folgende Werte benutzt: $\alpha_g = 200$, $\alpha_f = 15$, kritische Punkte zur Komplexbildung im Abstand zwischen 7 und 11Å, bei einer Abweichung bis zu 1,5Å, Mindesttiefe für Vertiefungen 1,5Å.

Es zeigt sich, dass die Ergebnisse mit eingeschalteter Aminosäuren-Gewichtung in den meisten Fällen besser sind, als die der reinen geometrischen Bewertung. Bis auf die beiden Ausnahmen 1KAC und 1WEJ, liegen bei den übrigen Komplexen die Platzierungen in der Region der geometrischen Bewertung. Im Folgenden wird deshalb bei den ungebundenen Komplexen stets mit der Aminosäuren-Gewichtung gerechnet. Die Berechnung der Elektrostatik mit der im Abschnitt 3.3.2 beschriebenen Methode brachte hingegen nur in sehr wenigen Fällen eine leichte Verbesserung und in vielen Fällen eine Verschlechterung der Position guter Komplexe. Sie wird im weiteren Verlauf daher nicht eingesetzt.

4.5.2 Ungebundene Komplexe

Tabelle 4.10 enthält die Ergebnisse des Dockings der ungebundenen Testfälle. Es wurden die gleichen Parameter wie im letzten Absatz bei den gebundenen Fällen verwendet. Gezeigt werden wieder die Position der ersten nahen nativen Lösung und die Position der besten erzeugten Struktur. Es sind auch einige Datensätze aufgeführt, bei denen keine Lösungen mit einem RMSD unter 5Å gefunden werden konnte. Dies dient der Veranschaulichung, in welcher Region die Positionen von gescheiterten Testfällen liegen können.

Tabelle 4.9: Anzahl der Treffer im Datensatz bei unterschiedlichen RMSD Schwellen. Ergebnisse für die *bound* (b) und *unbound* (u) Fälle erzielt mit *AlhpaDock*. Zum Vergleich die Werte für den *unbound* Datensatz gemäß [87].

RMSD Schwelle	AlphaDock (b)	AlhpaDock (u)	ContexShape (u)
5Å	43	37	41
10Å	65	58	65
15Å	73	68	78
20Å	76	73	84

Die Ergebnisse der ungebundenen Komplexe sind wie zu erwarten war in vielen Fällen schlechter, als die der gebundenen Testfälle. Es finden sich aber auch hier eine Reihe guter Lösungen, bei 6 Komplexen sogar eine Struktur auf den ersten 9 Plätzen mit einem RMSD unter 5Å. Um die Qualität der Gesamtheit der Vorhersagen zu überprüfen, eignet sich folgendes Maß. Es wird geschaut, wie viele Treffer im gesamten Datensatz erzielt werden, wobei die Schwelle, welcher Komplex als solcher gezählt wird, variiert wird. Zudem betrachtet man nur eine gewisse Anzahl der produzierten Lösungen. In [87] wird die Grenze bei den ersten 3600 Strukturen gewählt und die Schwelle des RMSD für einen Treffer in Schritten zwischen 5Å und 20Å gesetzt. Wendet man dieses Vorgehen auf die Ergebnisse des Dockings mit *AlphaDock* an, so ergeben sich die in Tabelle 4.9 gezeigten Werte. Zum Vergleich ist die Zahl der Treffer angegeben, die mit dem Programm *ContextShape* aus [87] erzielt werden.

Tabelle 4.10: Ergebnis des Docking-Prozesse mit den ungebundenen Strukturen als Testfälle. Die benutzten Parameter finden sich im Text.

	Erste n.n. Struktur		beste Struktur			Erste n.n. Struktur		beste Struktur	
PDB	Rang	RMSD	Rang	RMSD	PDB	Rang	RMSD	Rang	RMSD
1A2K	995	3,03	3774	1,25	1I4D	316	3,28	316	3,28
1ACB	48	4,66	48	4,66	1IB1	53	4,58	88	3,54
1AHW	15303	4,94	15303	4,94	1IBR	415	4,21	415	4,21
1AK4	788	4,27	10942	3,32	1IJK	1695	4,71	4233	2,64
1AKJ	68	4,55	90452	5,69	1IQD	1971	1,95	3531	1,43
1ATN	9818	4,50	20237	3,97	1JPS	181596	5,91	181596	5,91
1AVX	6	2,41	309	1,70	1K4C	3517	5,89	11024	5,14
1AY7	5	3,69	55	2,32	1K5D	2270	2,61	2270	2,61
1B6C	88	4,73	199	3,32	1KKL	26	3,82	1256	2,57
1BJ1	1977	1,66	2856	1,66	1KLU	6255	4,73	17628	3,27

	Erste n.n. Struktur		beste Struktur			Erste n.n. Struktur		beste Struktur	
PDB	Rang	RMSD	Rang	RMSD	PDB	Rang	RMSD	Rang	RMSD
1BUH	2774	4,87	5697	4,61	1KXQ	13	4,91	13	4,91
1BVK	6295	5,80	13158	5,57	1M10	1164	4,76	4354	3,90
1BVN	4	2,92	4	2,92	1MAH	5845	4,11	8791	3,45
1CGI	24	4,63	24	4,63	1ML0	1708	2,63	1708	2,63
1D6R	415	4,39	5624	2,63	1MLC	20293	4,39	20293	4,39
1DFJ	10	2,50	10	2,50	1NCA	17167	4,20	20122	3,61
1E6E	45	4,45	126	3,81	1NSN	14727	4,45	14727	4,45
1E96	1642	5,99	9972	5,42	1PPE	16	4,87	661	2,45
1EAW	54	3,82	374	2,27	1QFW	5328	4,71	7768	3,75
1EER	80	3,77	337	2,72	1RLB	1174	4,66	3780	2,63
1EWY	386	4,85	1418	4,48	1SBB	80705	4,16	137117	3,61
1EZU	21	3,03	21	3,03	1TMQ	15	3,70	131	2,94
1F34	882	3,71	882	3,71	1UDI	160	4,67	3696	5,04
1F51	85	4,52	153	2,63	1VFB	13310	4,85	13310	4,85
1FAK	1142	4,80	1216	4,20	1WEJ	10139	5,45	10139	5,45
1FC2	8558	4,43	9974	3,41	1WQ1	1126	4,57	9904	1,99
1FQ1	1318	3,92	2491	3,66	2BTF	4316	4,85	4316	4,85
1FQJ	7621	4,11	7621	4,11	2JEL	1814	4,01	1814	4,01
1FSK	2650	4,53	10263	3,66	2MTA	6211	3,50	6211	3,50
1GHQ	3155	5,60	53069	5,04	2PCC	10729	4,95	10729	4,95
1GP2	332527	5,95	332528	5,53	2QFW	2500	4,54	9974	2,62
1GRN	900	3,90	1732	3,60	2SIC	13	4,07	310	2,50
1H1V	6976	4,57	6976	4,57	2SNI	5	4,78	11	1,23
1HE1	152	2,27	152	2,27	2VIS	800	4,43	1400	1,78
1HIA	72	3,83	72	3,83	7CEI	751	3,33	751	3,33
1I2M	9	3,96	14	2,94					

4.6 Diskussion

Die Ergebnisse im letzten Abschnitt zeigen, dass durch die Bewertungsfunktion für viele Fälle nahe native Lösungen in den ersten Prozent der produzierten Komplexe gefunden werden können. Die Resultate sind jedoch stark unterschiedlich, so dass man zum besseren Verständnis des Prozesses gezielt einige Testfälle untersuchen muss. Zu diesem Zweck werden im Folgenden verschiedene Darstellungen der berechneten Komplexe gezeigt, die verdeutlichen, bei welchen Konstellationen gute Ergebnisse erzielt werden können und unter welchen Bedingungen der

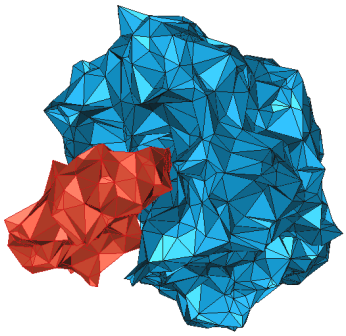
Algorithmus Strukturen bevorzugt, die nicht den in der Natur vorkommenden entsprechen.

4.6.1 Erfolgreiche Testfälle

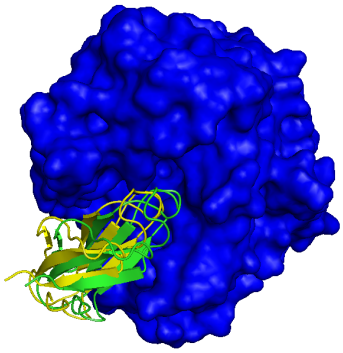
Abbildung 4.6 zeigt die Alpha-Shape und die Oberflächendarstellung von drei erfolgreichen Struktur-Vorhersagen. Es ist zu erkennen, dass sich der Ligand jeweils tief im inneren einer Tasche des Rezeptors befindet. Solche Anordnungen werden von der Bewertungsfunktion besonders hervorgehoben. Es ist aber auch zu sehen, dass die berechnete Position des Liganden jeweils etwas zu dicht an den Rezeptor herangerückt liegt. Der durch die Bewertung zugelassene Überlapp der Bindungspartner ist also etwas zu großzügig gewählt.

4.6.2 Gescheiterte Testfälle

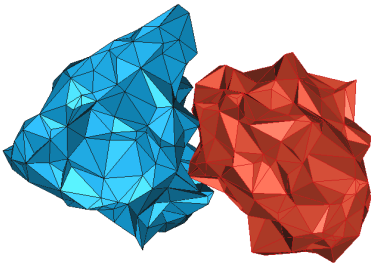
Die Abbildungen 4.7 und 4.8 zeigen prototypische Fälle, in denen der Algorithmus die korrekte Position des Liganden nicht bestimmen kann. Bild (a) zeigt jeweils die Alpha-Shape des richtig gedockten Komplexes. Darin ist bereits deutlich zu erkennen, dass auf der Oberfläche eine größere Vertiefung existiert, in die der Ligand platziert werden kann. Genau diese Stelle wird von der Suche auch gefunden und von der Bewertungsfunktion entsprechend belohnt. Bild (b) zeigt die Alpha-Shape des als beste Lösung berechneten Komplexes. Die Oberflächendarstellung der Bindungspartner in den Bildern (c) und (d) verdeutlicht die Problematik. Die berechnete Lösung zeigt einen Ligaden, der sich in die größtmögliche Vertiefung einpasst. Der zu sehende Überlapp der Oberflächen wird zur impliziten Behandlung der Flexibilität bei der Komplex-Bildung zugelassen. Er ist jedoch wie bei den geglückten Fällen etwas zu groß gewählt. Eine genauere Anpassung der Parameter des Docking-Prozesses kann hier Abhilfe schaffen, wird aber das grundsätzliche Problem der geometrischen Bewertung nicht lösen.



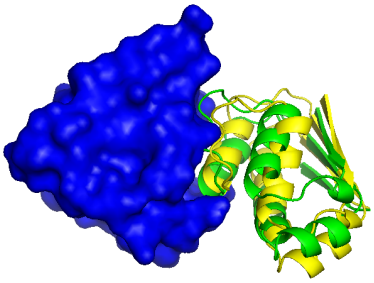
(a) 1BVN



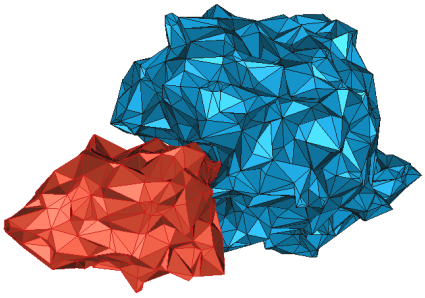
(b) 1BVN



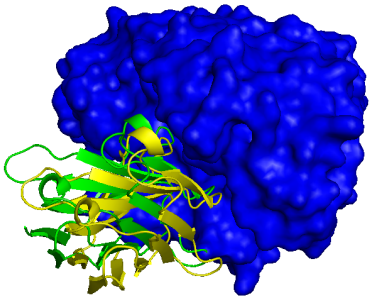
(c) 1AY7



(d) 1AY7

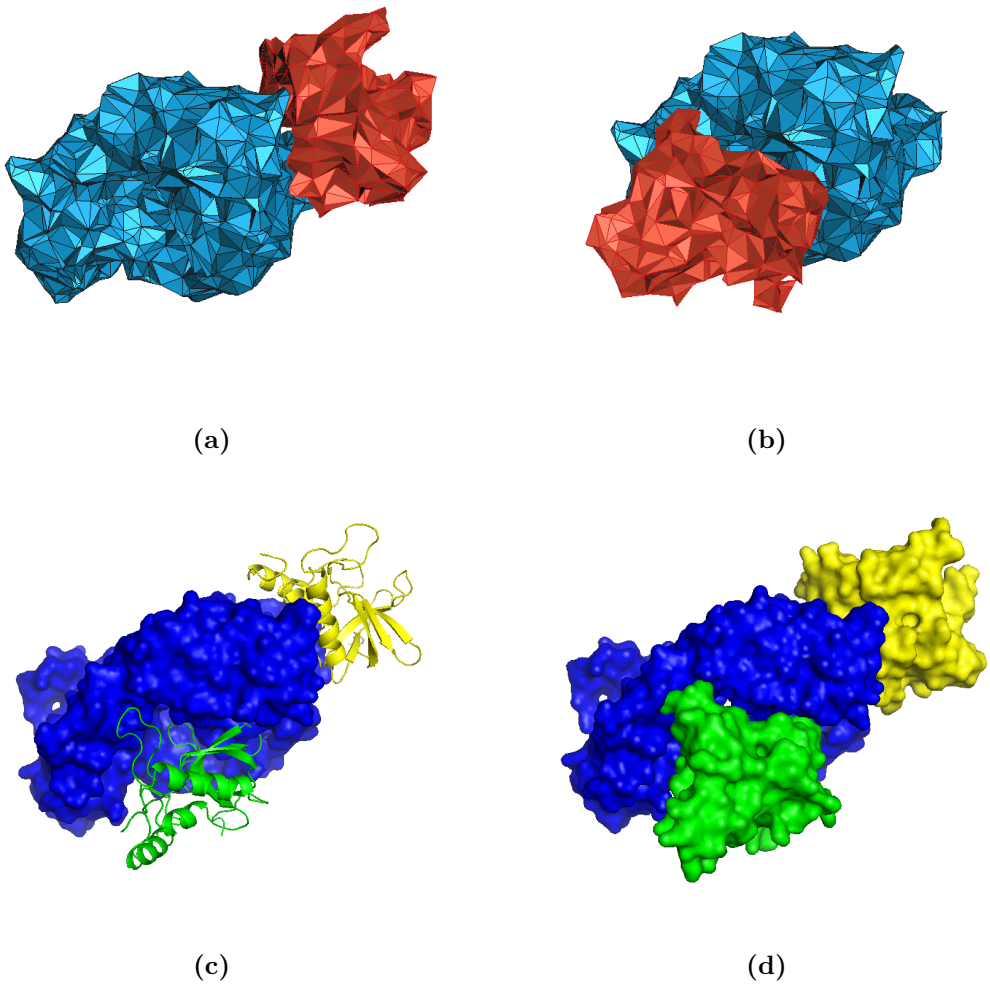


(e) 1KXQ



(f) 1KXQ

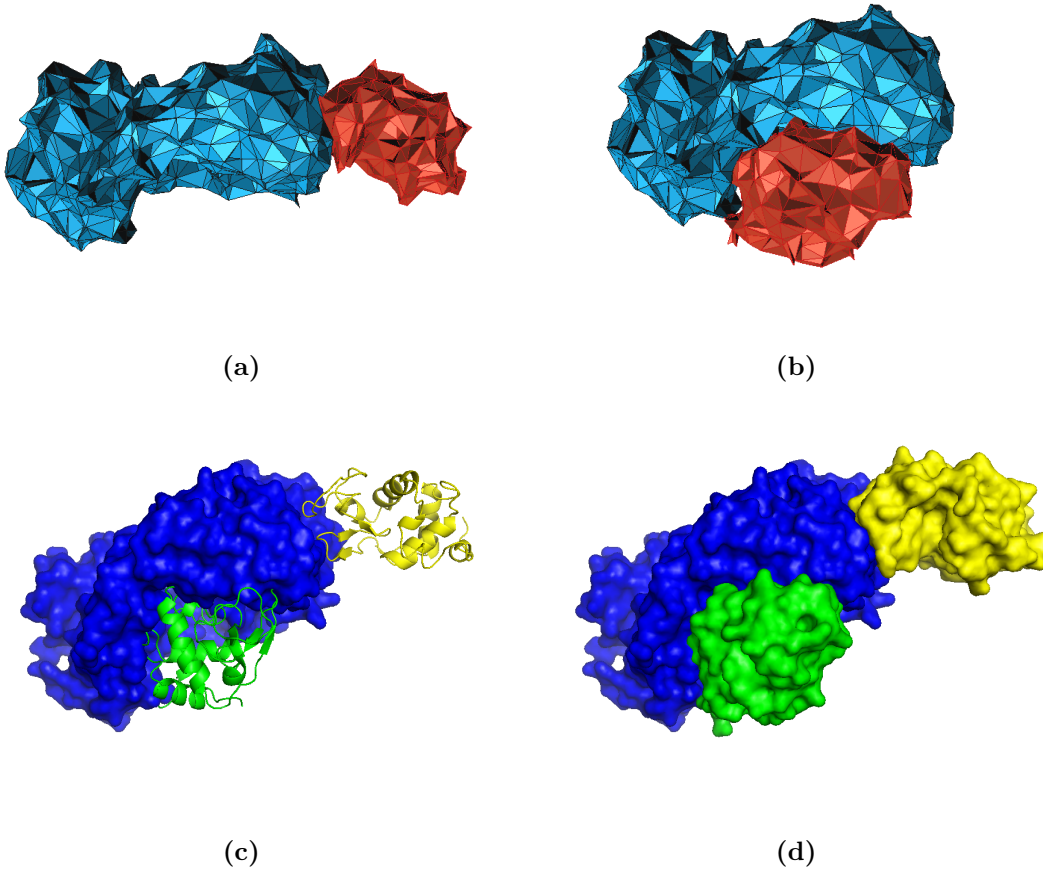
Abbildung 4.6: Beispiele für erfolgreiche Vorhersagen. Neben der Alpha-Shape des gedockten Proteins ist jeweils der Rezeptor in Oberflächendarstellung zu sehen und reduzierte Ansichten des Liganden. In grün ist die beste berechnete Position zu sehen, in gelb die richtige Lösung.



(a) **(b)**

(c) **(d)**

Abbildung 4.7: Beispiel einer fehlgeschlagenen Vorhersage für den Komplex 1NSN. Gezeigt wird die Alpha-Shape der korrekten Anordnung (a) und die der von der Bewertungsfunktion als beste Lösung vorgeschlagene Struktur (b). In der Oberflächendarstellung (c) und (d) ist die richtige Position des Liganden in gelb und die berechnete in grün gezeichnet.



(a) **(b)**

(c) **(d)**

Abbildung 4.8: Beispiel einer fehlgeschlagenen Vorhersage für den Komplex 1MLC. Gezeigt wird die Alpha-Shape der korrekten Anordnung (a) und die der von der Bewertungsfunktion als beste Lösung vorgeschlagene Struktur (b). In der Oberflächendarstellung (c) und (d) ist die richtige Position des Liganden in gelb und die berechnete in grün gezeichnet.

4.7 Fazit

Die bisher mit dem Programm *Alpha-Dock* erzielten Ergebnisse zeigen, dass es mit Hilfe der Alpha-Shape-Darstellung von Proteinen möglich ist, ein Docking-Verfahren zu realisieren. In erster Linie eignet sich die Herangehensweise zur schnellen Beurteilung der geometrischen Passgenauigkeit eines Komplexes. Im Durchschnitt benötigt die Erzeugung und Bewertung von einigen Hunderttausend Anordnungen weniger als fünf Minuten. Die Grenzen der reinen Geometrie-Beurteilung wurden genau wie in anderen Verfahren deutlich. Neben der Passgenauigkeit der Komplexe werden daher weitere Kriterien benötigt. Beispielhaft wurde hier eine Aminosäuren-spezifische Gewichtung in das Alpha-Shape-basierte Docking eingebaut. Erste Versuche, eine Bewertung der Elektrostatik aufzunehmen, sind gescheitert. Die verwendete Abschätzung der Energien erwies sich als nicht auf die Alpha-Shape übertragbar. Die Entwicklung einer angepassten Energiefunktion muss noch erfolgen, ist aber grundsätzlich denkbar. Weitere Ansätze für einen weiteren Ausbau der Software finden sich im Ausblick.

Kapitel 5

Zusammenfassung und Ausblick

5.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde ein neuer Ansatz zur Lösung des Protein-Protein-Docking-Problems verfolgt und dessen Merkmale und Möglichkeiten untersucht. Dazu wurde eine vollständig neue Software, genannt *AlphaDock*, entwickelt, die vom Einlesen der Proteindaten über das Erzeugen und Bewerten von Proteinkomplexen bis hin zur Auswertung der Ergebnisse, alle Schritte eines Docking-Verfahrens realisiert.

Hauptmerkmal der Vorgehensweise ist die durchgängige Verwendung der Alpha-Shape [30] der Proteine. Diese Teilmenge der Delaunay-Triangulation bietet die Möglichkeit einer verlustfreien komprimierten Repräsentation, deren Auflösung mittels eines Parameters eingestellt werden kann. Sie wird hier zum einen dazu benutzt, um charakteristische Erhebungen und Vertiefungen auf den über sie dargestellten Proteinoberflächen zu ermitteln. Über diese werden wiederum mögliche Proteinkomplexe erzeugt, indem die zur Überlagerung von gegensätzlichen Merkmalen benötigten Transformationen bezüglich der Alpha-Shapes berechnet werden. Auch die Beurteilung der so generierten Positionierungen der Bindungspartner wird mit Hilfe der Alpha-Shape der Proteine durchgeführt. Dazu werden sowohl geometrische als auch physiko-chemische Eigenschaften der Proteine auf die Tetraeder der zugehörigen Triangulationen abgebildet und die Kompatibilität

mit den bei einer Anordnung darin enthaltenen Atomen des anderen Proteins überprüft.

Zusätzlich kann auch eine Visualisierung über die Alpha-Shapes erfolgen. Durch die Anbindung des 3D-Viewers *geomview* [3] können sowohl die von *AlphaDock* erzeugten Lösungen aus allen Richtungen betrachtet werden, als auch Analysen der korrekt gedockten Proteine erfolgen.

Ein Vorteil des beschriebenen Ansatzes ist es, dass die Genauigkeit der Ergebnisse von der Auflösung der verwendeten Alpha-Shapes abhängt und die benötigte Rechenzeit entsprechend skaliert. D.h. es kann mit wenig Aufwand eine erste Näherung einer möglichen Struktur bestimmt werden, und je nach vorhandener Rechenleistung das Gesamtergebnis verbessert werden. Die bisher berechneten Resultate zeigen, dass es für zahlreiche Fälle möglich ist, mit dem vorgestellten Algorithmus innerhalb von wenigen Minuten mindestens eine nahe native Lösung unter den ersten Prozent der erzeugten Komplexe zu finden.

Das noch nicht bei allen Testfällen zufriedenstellende Ergebnisse erzielt werden konnten, verwundert nicht, bedenkt man die Kürze der Existens des Algorithmus und die deshalb fehlende vollständige Optimierung der zahlreichen Parameter. Ein Pluspunkt des Verfahrens besteht aber in seiner guten Erweiter- und Anpassbarkeit. Damit sollte es in Zukunft gelingen, die Leistung der Software weiter zu steigern und die Anwendungsmöglichkeiten auszubauen. Der nächste Abschnitt enthält dazu einige Vorschläge, wie die Entwicklung fortgeführt werden kann.

5.2 Ausblick

Nachdem bisher gezeigt wurde, dass der entwickelte Algorithmus prinzipiell in der Lage ist, nahe native Lösungen zu generieren und die Bewertungsroutinen diese auch deutlich über dem Durchschnitt einstufen, gilt es in der weiteren Entwicklung vor allem, die Positionierung guter Lösungen weiter zu verbessern. Daneben sollte es ein Ziel sein, die Anzahl der erzeugten Komplexe zu erhöhen, die der korrekten Lösung nahe kommen. Nicht zuletzt gibt es auch noch einige Möglichkeiten, die Geschwindigkeit der Berechnungen zu erhöhen.

Obwohl während der Ausarbeitung der Ergebnisse schon zahlreiche Werte für die einzustellenden Parameter von *AlphaDock* getestet wurden, besteht noch die Möglichkeit, weitaus bessere Einstellungen zu finden. Eine systematische Optimierung der Einflussgrößen war nicht Ziel der Arbeit, sondern muss im nächsten Schritt erfolgen. Dazu gehört auch eine weitere Verfeinerung der Einstellungen in Abhängigkeit von der behandelten Problemklasse (Antikörper-Antigen, Enzym-Inhibitor/Substrat und sonstige Komplexe).

Um die Chance zu erhöhen, im Bewertungsschritt eine gute Lösung zu finden, ist es hilfreich, wenn der Generator zuvor möglichst viele zu favorisierende Komplexe erzeugt hat, von denen dann einige einen hohen Score erhalten können. Die Berechnung von potentiellen Anordnungen der Proteine kann im Algorithmus einfach durch ein anderes Verfahren ersetzt werden. Denkbar ist es, dass eine weitere Analyse von existierenden Interfaces in Alpha-Shape-Darstellung andere Kriterien als die hier verwendeten hervorbringt, die dann zur Berechnung der Transformationen herangezogen werden können. Alternativ können diese Merkmale als Verfeinerung des vorhandenen Ansatzes dienen. Eine verbesserte Taschensuche könne mit den in [24] und [7] beschriebenen Verfahren erfolgen, die ebenfalls auf Alpha-Shapes basieren.

Auch die Bewertungsfunktionen in *AlphaDock* lassen sich problemlos erweitern oder durch andere ersetzen. Denkbar ist z.B. ein Score, der auf einer Beurteilung des Interfaces des erzeugten Komplexes beruht. Dabei könnte die in [9] vorgestellte Interfacedefinition benutzt werden, welche ebenfalls auf einer Alpha-Shape-Darstellung aufbaut. Zudem können jederzeit weitere physiko-chemische Faktoren in die Bewertung aufgenommen werden, indem die Gewichte der Hashfunktion entsprechend angepasst werden.

Im Gegensatz zu den FFT-basierten Verfahren, finden hier alle Berechnungen im direkten Raum statt. Dadurch hat man jederzeit den Zugriff auf einzelne erzeugte Strukturen, ohne das erste eine Rücktransformation aus dem Fourierraum erfolgen muss. Dies vereinfacht die Analyse und verspricht auch eine Erleichterung bei der Erweiterung des Verfahrens in Richtung einer expliziten Einbeziehung der Verformungen der Proteine beim Docking. D.h. die Flexiblität bei der Anordnung der Bindungspartner würde nicht mehr durch das Zulassen eines gewissen Überlapps der starren Strukturen behandelt, sondern durch die Berechnung der

Bewegungen einzelner Teile der Proteine. Hinweise darauf, wie das zu geschehen hat, kann eine Untersuchung der Veränderung der Alpha-Shape vom ungebundenen zum gebunden Zustand liefern.

Die Bewertung der generierten Komplexe kann unabhängig voneinander erfolgen und damit parallel durchgeführt werden. Dies wird in *AlphaDock* bereits ausgenutzt, indem die Berechnungen auf mehrere Threads aufgeteilt werden, je nachdem wie viele Kerne auf der Maschine vorhanden sind. Eine Parallelisierung ließe sich auch für einen Cluster realisieren. In diesem Fall würde jeder Knoten einen Teil der Transformationen zugewiesen bekommen und die Scores dazu berechnen. Somit ließen sich in der gleichen Zeit noch mehr Komplexe beurteilen oder die gleiche Anzahl Komplexe noch genauer untersuchen.

Abschließend kann man feststellen, dass mit dem vorliegenden Programm eine Grundlage geschaffen wurde, die eine neue Herangehensweise an die Lösung des Protein-Docking-Problems erlaubt und noch das Potential für zahlreiche Verbesserungen und Erweiterungen bietet.

Anhang A

Die Software *AlphaDock*

A.1 Aufbau

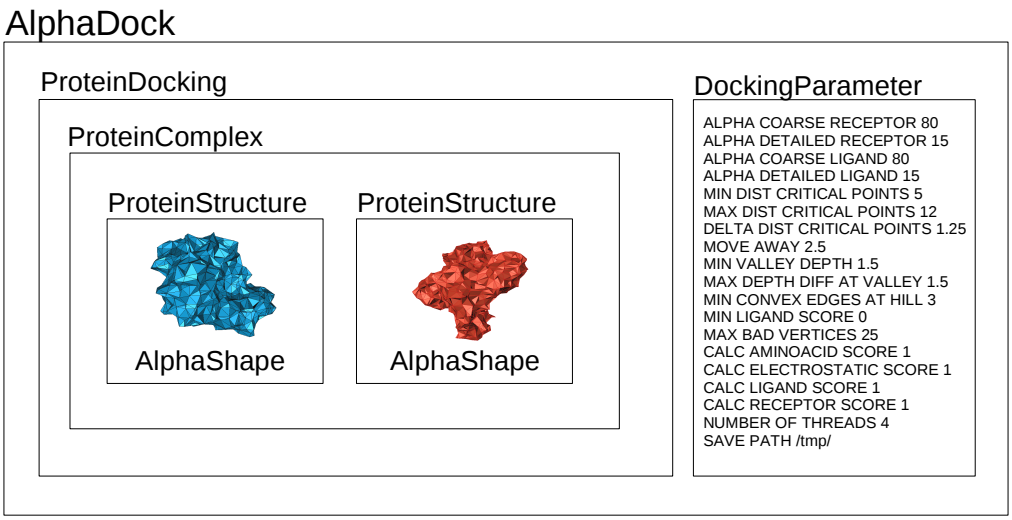


Abbildung A.1: Prizipieller Aufbau der Software AlphaDock

Abbildung A.1 zeigt grob die Struktur des Programms. Die wichtigsten Objekte sind als Rechtecke dargestellt und ihre Verschachtelung ist zu erkennen. Das Objekt *ProteinDocking* steuert den Ablauf des Algorithmus, es läd die Proteine-Daten und erzeugt damit eine Instanz des Objektes *ProteinComplex*. Dieses beinhaltet wiederum zwei Objekte des Typs *ProteinStructure*, innerhalb derer die *AlphaShape*s verwaltet werden.

A.2 Implementierung

Es existieren zwei Versionen des Programms, eine kommandozeilenbasierte Version (CLI) und eine mit grafischem Benutzerinterface (GUI). Beide Varianten sind in C++ unter Linux entwickelt worden. In der folgenden Tabelle werden kurz die Bibliotheken erwähnt, die für wesentliche Berechnungen benutzt wurden. Es wird jeweils angegeben, welche Teile des Algorithmus damit realisiert wurden und welche Version verwendet wurde.

Tabelle A.1: Wesentliche Bibliotheken bei der Implementierung von *AlphaDock*.

Bibliothek	Version	Verwendung
CGAL [2]	3.4	Alpha-Shapes, Hashing, Geomview Anbindung, ...
Wild Magic [6]	4.9	Abstandsberechnung Punkt zu Dreieck
Boost [1]	1.39	Threads
Qt [5]	4.5	GUI

Alle Bibliotheken sind für die nicht kommerzielle Nutzung frei verfügbar. Sie werden für aktuelle Linux-Distributionen als Binärpakete oder im Quelltext angeboten. Für die Verwendung von *AlphaDock* ist zu beachten, dass die Boost Bibliothek in einer Version ab 1.36 verwendet wird, da erst darin die im Programm benutzte Parameterübergabe an Threads funktioniert.

Genannt werden soll an dieser Stelle auch noch einmal das Programm *Geomview* [3], welches für die direkte Visualisierung der Alpha-Shapes in der GUI-Version von *AlphaDock* benutzt wird. Die aktuelle Version 1.9.4 ist ebenfalls Teil gängiger Distributionen oder kann ggf. aus den Quellen gebaut werden. Der Vorteil dieses Viewers liegt in der vorhandenen Integration in CGAL, wodurch eine einfache Steuerung aus dem eigenen Programm heraus möglich ist.

Die in dieser Arbeit vorgestellten Ergebnisse sind auf einem PC mit 2.8 GHz Intel Quad-Core-CPU und 8GB Hauptspeicher berechnet worden, der unter Fedora Linux 10 (64-Bit) lief. *AlphaDock* wurde mit GCC in der Version 4.3.2 erstellt.

A.3 Benutzung und Parameter

CLI

Der Aufruf der Kommandozeilen Version geschieht wie folgt:

```
$ ./AlphaDockCLI <Rezeptor.pdb> <Ligand.pdb> [<Parameter.param>]
```

D.h. die ersten beiden Parameter sind die Dateinamen der PDB-Dateien des Rezeptors und des Liganden, die zwingend angegeben werden müssen. Als dritter Parameter kann optional der Dateiname eine Parameterdatei übergeben werden. Das Format dieser Datei und die Auswirkung der darin enthaltenen Werte zeigt Tabelle A.2. Wird keine Parameterdatei übergeben, werden die im Programm enthaltenen Standardwerte verwendet, die unten aufgeführt sind.

GUI

Die Bedienung der GUI-Version ist weitgehend selbsterklärend. Rezeptor und Liganden können über Auswählknöpfe geladen werden. Die Parameter für den Viewer und für die Berechnung werden jeweils in Reitern eingestellt. Die Berechnung der Transformationen und der Scores erfolgt über den Knopf „Start Docking“. Die Ergebnisse erscheinen in Tabellenform in einem weitern Reiter. Alternativ können zuvor berechnete Lösungen geladen werden. Ist der Viewer gestartet, kann die zu einer Lösung gehörende Transformation durch Doppelklicken der Ergebniszeile auf den Liganden angewendet werden. Die aktuelle Lösung kann über den Menüpunkt „Save PDB“ im File-Menü als PDB-Datei gespeichert werden. Der Screenshot zeigt das GUI mit den Einstellmöglichkeiten für den Viewer.

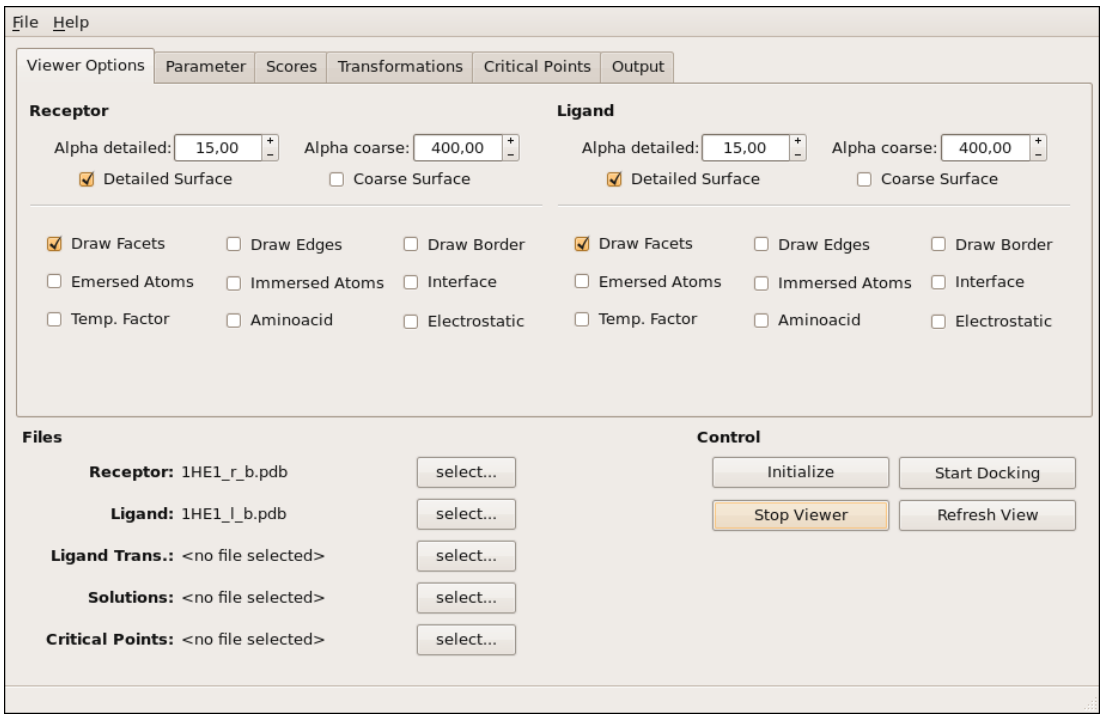


Abbildung A.2: Screenshot von AlphaDockGUI

Tabelle A.2: In *AlphaDock* verwendete Parameter und ihre Standardwerte.

Parameter	Standard	Bedeutung
ALPHA_COARSE_RECEPTOR	80	Alpha_{grob} des Rezeptors
ALPHA_DETAILED_RECEPTOR	15	Alpha_{fein} des Rezeptors
ALPHA_COARSE_LIGAND	80	Alpha_{grob} des Liganden
ALPHA_DETAILED_LIGAND	15	Alpha_{fein} des Liganden
MIN_DIST_CRITICAL_POINTS	5	Mindestabstand zweier kritischer Punkte bei der Berechnung der Ligand-Transformation
MAX_DIST_CRITICAL_POINTS	12	Maximaler Abstand zweier kritischer Punkte bei der Berechnung der Ligand-Transformation
DELTA_DIST_CRITICAL_POINTS	1.25	Erlaubte Abweichung der beiden Abstände der bei der Transformationsberechnung benutzten Punktepaare
MOVE_AWAY	2.5	Größe der Verschiebung des Liganden in Richtung der gewählten Normalen im letzen Schritt der affinen Transformation
MIN_VALLEY_DEPTH	1.5	Minimale mittlere Tiefe eines kritischen Punktes
MAX_DEPTH_DIFF_AT_VALLEY	1.5	Maximale Abweichung von der mittleren Tiefe nach unten, die ein Nachbar eines tiefliegenden kritischen Punktes besitzen darf
MIN_CONVEX_EDGES_AT_HILL	3	Mindestanzahl konvexer Kanten, die in ein hervorstehenden kritischen Punkt enden müssen
MIN_LIGAND_SCORE	0	Minimaler Score des Liganden bei dem der Score des Rezeptors berechnet wird
MAX_BAD_VERTICES	25	Maximal erlaubte Zahl von Atomen im inneren der Alpha-Shape des Bindungspartners bevor die Bewertung abbricht
CALC_AMINOACID_SCORE	1	Schaltet die Berechnung des Aminosäuren Scores ein (1) oder aus (0)
CALC_ELECTROSTATIC_SCORE	1	Schaltet die Berechnung der Elektrostatik ein (1) oder aus (0)
CALC_LIGAND_SCORE	1	Schaltet die Berechnung des Ligand Scores ein (1) oder aus (0)
CALC_RECEPTOR_SCORE	1	Schaltet die Berechnung des Rezeptor Scores ein (1) oder aus (0)
NUMBER_OF_THREADS	4	Anzahl der für die Bewertung zu verwendenden Threads
SAVE_PATH	/tmp/	absoluter Pfad unter dem die Ergebnisse gespeichert werden

Literaturverzeichnis

[1] Boost, *C++ Libraries*. http://www.boost.org/.

[2] Cgal, *Computational Geometry Algorithms Library*. http://www.cgal.org.

[3] Geomview, *An interactive 3D viewer*. http://www.geomview.org.

[4] Gpl, *GNU General Public License*. http://www.gnu.org/licenses/gpl.txt.

[5] Qt, *A cross-platform application and UI framework*. http://qt.nokia.com/.

[6] Wild Magic, *Real-Time 3D Graphics Engine*. http://geometrictools.com.

[7] Albou, L.P., B. Schwarz, O. Poch, J.M. Wurtz und D. Moras: *Defining and characterizing protein surface using alpha shapes*. Proteins: Structure, Function, and Bioinformatics, 76(1):1–12, 2008.

[8] Ausiello, G., G. Cesareni und M. Helmer-Citterich: *ESCHER: A New Docking Procedure Applied to the Reconstruction of Protein Tertiary Structure*. PROTEINS: Structure, Function, and Genetics, 28:556–567, 1997.

[9] Ban, Yih-En Andrew, Herbert Edelsbrunner und Johannes Rudolph: *Interface surfaces for protein-protein complexes*. Jornal of the ACM, 53(3):361–378, 2006.

[10] Bernardini, Fausto und Chandrajit L. Bajaj: *Sampling and Reconstructing Manifolds using Alpha-Shapes*. In: *Proc. 9th Canadian Conf. Computational Geometry*, Seiten 193–198, 1997.

[11] Bernstein, F. C., T. F. Koetzle, G. J. B. Williams, E. F. Meyer Jr, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi und M. Tasumi: *The Protein Data Bank: a computer-based archival file for macromolecular structures*. J. Mol. Biol., 112:535–542, 1977.

[12] Bondi, A.: *van der Waals Volumes and Radii*. Journal of Physical Chemistry, 68(3):441–451, 1964.

[13] Bonvin, Alexandre M. J. J.: *Flexible protein-protein docking.* Current Opinion in Structural Biology, 16(2):194–200, 2006.

[14] Bonvin, A.M.J.J., R. Boelens und R. Kaptein: *NMR analysis of protein interactions.* Current opinion in chemical biology, 9(5):501–508, 2005.

[15] Camacho, Carlos J. und Chao Zhang: *FastContact: rapid estimate of contact and binding free energies.* Bioinformatics, 21(10):2534–2536, 2005.

[16] Carter, P., V.I. Lesk, S.A. Islam und M.J.E. Sternberg: *Protein-protein docking using 3D-Dock in rounds 3, 4, and 5 of CAPRI.* Proteins: Structure, Function, and Bioinformatics, 60(2):281–288, 2005.

[17] Chandler, David: *Interfaces and the driving force of hydrophobic assembly.* Nature, 437:640–647, 2005.

[18] Chen, R. und Z. Weng: *Docking unbound proteins using shape complementarity, desolvation, and electrostatics.* Proteins, 47(3):281–294, 2002.

[19] Chernov, Alexander A.: *Protein crystals and their growth.* Journal of Structural Biology, 142(1):3–21, 2003.

[20] Connolly, M. L.: *Analytical molecular surface calculation.* J. Appl. Crystallogr., 16:548–558, 1983.

[21] Da, Tran Kai Frank und Mariette Yvinec: *3D Alpha Shapes.* In: *CGAL User and Reference Manual.* CGAL Editorial Board, 2007.

[22] Delaunay, Boris N.: *Sur la sphère vide.* Bulletin of Academy of Sciences of the USSR, (6):793–800, 1934.

[23] Duhovny, Dina, Ruth Nussinov und Haim J. Wolfson: *Efficient Unbound Docking of Rigid Molecules.* In: *WABI '02: Proceedings of the Second International Workshop on Algorithms in Bioinformatics*, Seiten 185–200, London, UK, 2002. Springer-Verlag.

[24] Dundas, J., Z. Ouyang, J. Tseng, A. Binkowski, Y. Turpaz und J. Liang: *CASTp: computed atlas of surface topography of proteins with structural and topographical mapping of functionally annotated residues.* Nucleic Acids Res, 34(Web Server issue):W116–8, 2006.

[25] Edelsbrunner, H.: *Algorithms in Combinatorial Geometry.* Springer-Verlag, Berlin, 1987. ISBN 3-540-13722-X.

[26] Edelsbrunner, Herbert: *Weighted alpha shapes.* Technical Report UIUCDCS-R-92-1760, Department of Computer Science, University of Illinois at Urbana-Champaign, USA, 1992.

[27] Edelsbrunner, Herbert: *The Union of Balls and Its Dual Shape*. Discrete & Computational Geometry, 13:415–440, 1995.

[28] Edelsbrunner, Herbert, Michael A. Facello und Jie Liang: *On the Definition and the Construction of Pockets in Macromolecules*. Discrete Applied Mathematics, 88(1-3):83–102, 1998.

[29] Edelsbrunner, Herbert, David G. Kirkpatrick und Raimund Seidel: *On the shape of a set of points in the plane*. IEEE Transactions on Information Theory, 29(4):551–558, 1983.

[30] Edelsbrunner, Herbert und Ernst P. Mücke: *Three-dimensional alpha shapes*. ACM Trans. Graph., 13(1):43–72, 1994.

[31] Emekli, U., D. Schneidman-Duhovny, H.J. Wolfson, R. Nussinov und T. Haliloglu: *HingeProt: Automated prediction of hinges in protein structures*. Proteins: Structure, Function, and Bioinformatics, 70(4):1219–1227, 2007.

[32] Fernández-Recio, J., M. Totrov und R. Abagyan: *ICM-DISCO docking by global energy optimization with fully flexible side-chains*. Proteins Structure Function and Genetics, 52(1):113–117, 2003.

[33] Fischer, Kaspar: *Introduction to Alpha Shapes*, 2000. http://people.inf.ethz.ch/fischerk/pubs/as.pdf.

[34] Gabb, H.A., R.M. Jackson und M.J.E. Sternberg: *Modelling protein docking using shape complementarity, electrostatics and biochemical information*. Journal of Molecular Biology, 272(1):106–120, 1997.

[35] Giesen, Joachim, Frédéric Cazals, Mark Pauly und Afra Zomorodian: *The conformal alpha shape filtration*. The Visual Computer, 22(8):531–540, 2006.

[36] Gray, J.J., S. Moughon, C. Wang, O. Schueler-Furman, B. Kuhlman, C.A. Rohl und D. Baker: *Protein–protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations*. Journal of Molecular Biology, 331(1):281–299, 2003.

[37] Güntert, Peter: *Automated NMR protein structure calculation*. Progress in Nuclear Magnetic Resonance Spectroscopy, 43(3-4):105–125, 2003.

[38] Halperin, I., B. Ma, H. Wolfson und R. Nussinov: *Principles of docking: An overview of search algorithms and a guide to scoring functions*. Proteins: Structure, Function, and Genetics, 47(4):409–443, 2002.

[39] Hein, Lutz: *Entwicklung der Beta-Blocker*. Pharmazie in unserer Zeit, 33(6):434 – 437, 2004.

[40] Heuser, P. und O. Martin: *Unbound Unbound Protein Protein Docking Dataset (UUPPDD)*, 2003. http://biotool.uni-koeln.de/uuppdd/.

[41] Heuser, Philipp: *Optimierte atom- und aminosäurespezifische Gewichtungsfaktoren für Protein-Docking Methoden*. Dissertation, Universität zu Köln, 2006.

[42] Hingerty, BE, RH Ritchie, TL Ferrell und JE Turner: *Dielectric effects in biopolymers: the theory of ionic saturation revisited*. Biopolymers, 24(3), 1985.

[43] Hou, T., J. Wang, L. Chen und X. Xu: *Automated docking of peptides and proteins by using genetic algorithm combined with a tabu search*. Protein Engineering, 12:639–647, 1999.

[44] Jackson, R.M., H.A. Gabb und M.J.E. Sternberg: *Rapid refinement of protein interfaces incorporating solvation: application to the docking problem.* Journal of Molecular Bbiology, 276(1):265, 1998.

[45] Janin, Joel: *Assessing predictions of protein-protein interaction: The CAPRI experiment*. Protein Science, 14(2):278–283, 2005.

[46] Johnson, C.S.: *Diffusion ordered nuclear magnetic resonance spectroscopy: principles and applications*. Progress in Nuclear Magnetic Resonance Spectroscopy, 34(3):203–256, 1999.

[47] Jones, S. und J.M. Thornton: *Protein-protein recognition*, Kapitel 2: Analysis and classification of protein-protein interactions from a structural perspective, Seiten 33–59. Oxford University Press, 2000.

[48] Jones, Susan und Janet M. Thornton: *Principles of protein–protein interactions*. Proc. Natl. Acad. Sci. USA, 93:13–20, 1996.

[49] Katchalski-Katzir, E., I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo und I. A. Vakser: *Molecular Surface Recognition: Determination of Geometric Fit Between Proteins and Their Ligands by Correlation Techniques*. In: *Proceedings of the National Academy of Sciences*, Band 89, Seiten 2195–2199. National Academy of Sciences, 1992.

[50] Knight, Robin D., Stephen J. Freeland und Laura F. Landweber: *Selection, history and chemistry: the three faces of the genetic code*. Trends in Biochemical Sciences, 24(6):241 – 247, 1999.

[51] Knorr, Edwin M., Raymond T. Ng und David L. Shilvock: *Performing boundary shape matching in spatial data*. In: *CASCON '96: Proceedings of the 1996 conference of the Centre for Advanced Studies on Collaborative research*, Seite 20. IBM Press, 1996.

[52] Kowalsman, N. und M. Eisenstein: *Inherent limitations in protein-protein docking procedures*. Bioinformatics, 23(4):421, 2007.

[53] Kozakov, D., R. Brenke, S.R. Comeau und S. Vajda: *PIPER: An FFT-Based Protein Docking Program with Pairwise Potentials*. Proteins: Structure, Function, and Bioinformatics, 65(2):392–406, 2006.

[54] Kozakov, D., O. Schueler-Furman und S. Vajda: *Discrimination of near-native structures in protein-protein docking by testing the stability of local minima*. Proteins: Structure, Function, and Bioinformatics, 72(3):993–1004, 2008.

[55] Kyte, Jack und Russell F. Doolittle: *A simple method for displaying the hydropathic character of a protein*. Journal of Molecular Biology, 157(1):105 – 132, 1982.

[56] Lamdan, Y. und H.J. Wolfson: *Geometric Hashing: A General And Efficient Model-based Recognition Scheme*. Proc. of the Second International Conference on Computer Vision, Seiten 238–249, 1988.

[57] Lattman, E. E.: *Optimal sampling of the rotation function*. Acta Crystallographica Section B, 28(4):1065–1068, 1972.

[58] Lee, B. und F. M. Richards: *The interpretation of protein structures: estimation of static accessibility.* J. Mol. Biol., 55:379–400, 1971.

[59] Lenhof, Hans-Peter: *New Contact Measures for the Protein Docking Problem*. Research Report MPI-I-97-1-004, Max-Planck-Institut für Informatik, Saarbrücken, Germany, 1997.

[60] Liang, Jie: *Computational Methods for Protein Structure Prediction and Modeling*, Band 1, Kapitel 6. Computation of Protein Geometry and Its Applications: Packing and Function Prediction, Seiten 181–206. Springer, New York, 2006.

[61] Liang, Jie, Herbert Edelsbrunner, Ping Fu, Pamidighantam V. Sudhakar und Shankar Subramaniam: *Analytical shape computation of macromolecules: I. molecular area and volume through alpha shape*. Proteins: Structure, Function, and Genetics, 33(1):1–17, 1998.

[62] Liang, Jie, Herbert Edelsbrunner, Ping Fu, Pamidighantam V. Sudhakar und Shankar Subramaniam: *Analytical shape computation of macromolecules: II. Inaccessible cavities in proteins.* Proteins: Structure, Function, and Genetics, 33(1):18–29, 1998.

[63] Lin, S.L. und R. Nussinov: *Molecular recognition via face center representation of a molecular surface.* J Mol Graph, 14(2):78–90, 95–77, 1996.

[64] Lin, S.L., R. Nussinov, D. Fischer und Wolfson H.J.: *Molecular surface representations by sparse critical points.* Proteins, 18(1):94–101, 1994.

[65] Massa, Werner: *Kristallstrukturbestimmung.* Teubner Verlag, Stuttgart, 2002. ISBN 3-519-23527-7.

[66] Merhof, Dorit, Martin Meister, Ezgi Bingöl, Peter Hastreiter, Christopher Nimsky und Günther Greiner: *Bildverarbeitung für die Medizin 2007*, Kapitel Generation of Hulls Encompassing Neuronal Pathways Based on Tetrahedralization and 3D Alpha Shapes, Seiten 308–312. Springer, Berlin Heidelberg, 2007.

[67] Meyer, M., P. Wilson und D. Schomburg: *Hydrogen bonding and molecular surface shape complementarity as a basis for protein docking.* J Mol Biol, 264(1):199–210, 1996.

[68] Mintseris, J., B. Pierce, K. Wiehe, R. Anderson, R. Chen und Z. Weng: *Integrating statistical pair potentials into protein complex prediction.* Proteins, 69:511–520, 2007.

[69] Mintseris, J., K. Wiehe, B. Pierce, R. Anderson, R. Chen, J. Janin und Z. Weng: *Protein-Protein Docking Benchmark 2.0: an Update.* Proteins, 60(2):214–216, 2005.

[70] Moont, G., H.A. Gabb und M.J.E. Sternberg: *Use of pair potentials across protein interfaces in screening predicted docked complexes.* Proteins: structure, function, and genetics, 35(3):364–373, 1999.

[71] Muñoz, Carlos A. Del Carpio, T. Peissker, A. Yoshimori und E. Ichiishi: *Docking Unbound Proteins with MIAX: A Novel Algorithm for Protein-Protein Soft Docking.* Genome Informatics, 14:238–249, 2003.

[72] Mücke, Ernst Peter: *Shapes and implementations in three-dimensional geometry.* Dissertation, University of Illinois at Urbana-Champaign, 1994.

[73] Nooren, I.M.A. und J.M. Thornton: *Diversity of protein–protein interactions.* The EMBO Journal, 22(14):3486–3492, 2003.

[74] Norel, Raquel, Donald Petrey, Haim J. Wolfson und Ruth Nussinov: *Examination of Shape Complementarity in Docking of* Unbound *Proteins*. Proteins: Structure, Function, and Genetics, 36(3):307–317, 1999.

[75] Nussinov, R. und H.J. Wolfson: *Efficient Detection of Three-Dimensional Structural Motifs in Biological Macromolecules by Computer Vision Techniques*. Proceedings of the National Academy of Sciences, 88(23):10495–10499, 1991.

[76] Ohbuchi, Ryutarou und Tsuyoshi Takei: *Shape-Similarity Comparison of 3D Models Using Alpha Shapes*. In: *Pacific Conference on Computer Graphics and Applications*, Seiten 293–302. IEEE Computer Society, 2003.

[77] Park, Si Hyung, Seoung Soo Lee und Jong Hwa Kim: *A Surface Reconstruction Algorithm Using Weighted Alpha Shapes*. In: *Fuzzy Systems and Knowledge Discovery*, Seiten 1141–1150. Springer-Verlag, 2005.

[78] Peters, K.P., J. Fauck und C. Frommel: *The Automatic Search for Ligand Binding Sites in Proteins of Known Three-dimensional Structure Using only Geometric Criteria*. Journal of Molecular Biology, 256:201–213, 1996.

[79] Pierce, B. und Z. Weng: *A combination of rescoring and refinement significantly improves protein docking performance*. Proteins: Structure, Function, and Bioinformatics, 72(1):270–279, 2008.

[80] Pratt, Lawrence R. und David Chandler: *Theory of the hydrophobic effect*. The Journal of Chemical Physics, 67:3683–3704, 1977.

[81] Rance, M., OW Sørensen, G. Bodenhausen, G. Wagner, RR Ernst und K. W
Üthrich: *Improved spectral resolution in COSY 1H NMR spectra of proteins via double quantum filtering*. Biochemical and biophysical research communications, 117(2):479, 1983.

[82] Ritchie, D.W.: *Recent progress and future directions in protein-protein docking*. Current Protein and Peptide Science, 9(1):1–15, 2008.

[83] Sandak, B., H.J. Wolfson und R. Nussinov: *Flexible Docking Allowing Induced Fit in Proteins: Insights From an Open to Closed Conformational Isomers*. Proteins: Structure, Function, and Genetics, 32:159–174, 1998.

[84] Schneidman-Duhovny, D., Y. Inbar, R. Nussinov und H. J. Wolfson: *Geometry-based flexible and symmetric protein docking*. Proteins, 60(2):224–231, 2005.

[85] Schneidman-Duhovny, D., R. Nussinov, H.J. Wolfson und US Binational: *Automatic prediction of protein interactions with large scale motion.* Proteins: Structure, Function, and Bioinformatics, 69(4):764–773, 2007.

[86] Sharf, A und A Shamir: *Feature-sensitive 3D Shape Matching.* In: *Proc. Computer Graphics International*, Seiten 596–599, 2004.

[87] Shentu, Z., M. Al Hasan, C. Bystroff und M.J. Zaki: *Context Shapes: Efficient Complementary Shape Matching for Protein-Protein Docking.* Proteins: Structure, Function, and Bioinformatics, 70(3):1056–1073, 2007.

[88] Shindyalov, I.N. und P.E. Bourne: *Protein structure alignment by incremental combinatorial extension (CE) of the optimal path.* Protein Engineering, 11(9):739–747, 1998.

[89] Taylor, F. J. R. und D. Coates: *The code within the codons.* Biosystems, 22(3):177 – 187, 1989.

[90] Taylor, Garry: *The phase problem.* Acta Crystallographica Section D, 59(11):1881–1890, 2003.

[91] Taylor, William Ramsay: *The classification of amino acid conservation.* Journal of Theoretical Biology, 119(2):205 – 218, 1986.

[92] Teichmann, Marek und Michael Capps: *Surface reconstruction with anisotropic density-scaled alpha shapes.* In: *VIS '98: Proceedings of the conference on Visualization '98*, Seiten 67–72, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press.

[93] Uetz, P. und E. Pohl: *Protein-Protein- und Protein-DNA-Interaktionen.* In: Wink, M. (Herausgeber): *Molekulare Biotchnologie*, Seiten 385–407. Wiley-VCH, 2004.

[94] Vajda, S. und C.J. Camacho: *Protein–protein docking: is the glass half-full or half-empty?* TRENDS in Biotechnology, 22(3):110–116, 2004.

[95] Vajda, Sandor und Dima Kozakov: *Convergence and combination of methods in protein-protein docking.* Current Opinion in Structural Biology, 19(2):164 – 170, 2009. Theory and simulation / Macromolecular assemblages.

[96] Via, A., F. Ferrè, B. Brannetti und M. Helmer-Citterich: *Protein surface similarities: a survey of methods to describe and compare protein surfaces.* Cellular and Molecular Life Sciences (CMLS), 57:1970 – 1977, 2000.

[97] Voet, D. und J. G. Voet: *Biochemistry*. John Wiley & Sons, Inc., 3. Auflage, 2004.

[98] Voronoi, G.F.: *Nouvelles applications des paramètres continus à la théorie de formes quadratiques. Premier Mémoire: Sur quelques propriétés de formes quadratiques positives parfaites.* Journal für die reine und angewandte Mathematik, (133):97–178, 1907.

[99] Voronoi, G.F.: *Nouvelles applications des paramètres continus à la théorie de formes quadratiques. Deuxième Mémorire: Recherches sur les parallélloèdres primitifs.* Journal für die reine und angewandte Mathematik, (134):198–287, 1908.

[100] Wang, C., P. Bradley und D. Baker: *Protein-Protein Docking with Backbone Flexibility*. Journal of Molecular Biology, 373(2):503–519, 2007.

[101] Wiehe, K., M.W. Peterson, B. Pierce, J. Mintseris und Z. Weng: *Protein Structure Prediction*, Band 413 der Reihe *Methods in Molecular Biology*, Kapitel 11 Protein-Protein Docking: Overview and Performance Analysis, Seiten 283–314. Humana Press, 2008.

[102] Wikipedia: *Aminosäuren — Wikipedia, Die freie Enzyklopädie*, 2009. [Online; Stand 2. September 2009].

[103] Wikipedia: *Genetischer Code — Wikipedia, Die freie Enzyklopädie*, 2009. [Online; Stand 2. September 2009].

[104] Wolfson, H.J. und R. Nussinov: *Computational Methods in Molecular Biology*, Band 32, Kapitel 14: From computer vision to protein structure and association, Seiten 313–334. Elsevier Science, 1999.

[105] Xu, Xiaolong und Koichi Harada: *Automatic surface reconstruction with alpha-shape method*. The Visual Computer, 19(7-8):431–443, 2003.

[106] Zimmermann, O.: *Untersuchungen zur Vorhersage der nativen Orientierung von Protein-Komplexen mit Fourier-Korrelationsmethoden*. Dissertation, Universität zu Köln, 2002.

[107] Zsoldos, Z., D. Reid, A. Simon, B. S. Sadjad und A. P. Johnson: *eHiTS: An Innovative Approach to the Docking and Scoring Function Problems*. Current Protein and Peptide Science, 7:421–435, October 2006.

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich bei der Erstellung dieser Arbeit unterstüzt haben. Ohne den Zuspruch, die Anteilnahme und den Rückhalt der Menschen in meinem Umfeld wäre ich nicht bis an diesen Punkt gekommen.

Ein besonderer Dank gilt Prof. Schrader, der mich während meiner Promotionszeit betreut hat. Er hat mich auf die Spur dieser interessanten Themenstellung gebracht und war stets offen für Fragen und Diskussionen. Zudem hat er mir durch sein Vertrauen in das Gelingen meines Vorhabens die nötige Selbstsicherheit gegeben.

Ebenfalls hervorzuheben ist die Unterstützung durch meinen Arbeitgeber, das Rechenzentrum der Universität zu Köln. Insbesondere meinem Abteilungsleiter Claus Kalle gilt mein Dank, da er die richtigen Worte zur Ermutigung finden konnte und mir die nötige Freiheit gab, die Promotion abzuschließen.

Ein großer Dank gilt meiner Familie und meinen Freunden, die mir stets die Kraft gegeben haben, um diese Aufgabe zu bewältigen. Speziell meiner Freundin Alke gebührt ein riesiges Dankeschön, da Sie mich durch alle Höhen und Tiefen der Ausarbeitung begleitet hat.

Abschließend möchte ich mich bei allen Korrekturlesern bedanken, vor allem bei Mario Mech, der jederzeit für das Aufspüren von Fehlern zu motivieren war.

Erklärung

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie noch nicht veröffentlicht worden ist sowie, dass ich eine solche Veröffentlichung vor Abschluß des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen dieser Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Prof. Dr. Rainer Schrader betreut worden.

Köln, den 3. November 2009

Niels Lange

Lebenslauf

Dipl. Physiker
Niels Christian Lange
Geb.: 14. November 1975 in Leverkusen
Staatsangehörigkeit: deutsch

Thürmchenswall 51
D-50668 Köln

seit **2004**	**Doktorand** bei Prof. Schrader, Universität zu Köln Thema: Protein-Protein-Docking
seit **2003**	**Wissenschaftlicher Mitarbeiter** der Universität zu Köln, RRZK, Abteilung Systeme
2002	**Diplomarbeit** über maschinelle Spracherkennung am Lehrstuhl für BioMolekulare Optik, LMU München Arbeitsgruppe Prof. Tavan
1996-2001	Studium der **Physik**, LMU München
1995-1996	**Zivildienst** in der Jugendherberge Morsbach
1986-1995	**Abitur** am Bodelschwingh Gymnaisum in Herchen
1982-1986	Fontane Grundschule Leverkusen

Köln, den 3. November 2009