

**Modellierung von Prenyltransferasen und Terpensynthesen
und Untersuchungen zum molekularen *Docking*
von Reaktionsintermediaten**

Dissertation

zur Erlangung des Doktorgrades der Naturwissenschaften (Dr. rer. nat.)

der

Naturwissenschaftlichen Fakultät II – Chemie, Physik und Mathematik

der Martin-Luther-Universität Halle-Wittenberg,

vorgelegt

von Herr Robert Klein

geb. am 02.07.1981 in Schkeuditz

Datum der Verteidigung: 24. Juni 2014

Gutachter:

1. Prof. Dr. Ludger A. Wessjohann, Martin-Luther-Universität Halle-Wittenberg
2. PD Dr. Thomas E. Exner, Eberhard-Karls-Universität Tübingen

"You don't use science to show you're right, you use science to become right."
– Randall Munroe

DANKSAGUNG

Ich bedanke mich bei Prof. Dr. Ludger A. Wessjohann für die freundliche Aufnahme in die Abteilung Natur- und Wirkstoffchemie am Leibniz-Institut für Pflanzenbiochemie (IPB) in Halle (Saale). Die vielseitigen Erfahrungen, die ich als Bioinformatiker in diesem spannenden Umfeld sammeln konnte, haben mich fachlich und menschlich sehr bereichert.

PD Dr. Wolfgang Brandt möchte ich danken für sein stets offenes Ohr, viele fachliche Diskussionen und die mir gelassene Experimentierfreiheit.

Prof. Dr. Alain Tissier und Romy Töpfer aus der Abteilung Stoffwechsel- und Zellbiologie des IPB haben mich auf das spannende Thema Kaurensynthase-ähnliche Enzyme gebracht. Ich danke ihnen herzlich für die bereitwillige Kooperation und die prompte Kommunikation experimenteller Daten aus dem Labor.

Bei PD Dr. Thomas E. Exner (Pharmazeutisches Institut der Eberhard-Karls-Universität Tübingen) und Dr. Oliver Korb (Cambridge Crystallographic Data Centre) bedanke ich mich für die Zurverfügungstellung des Quellcodes des *Docking*-Programms PLANTS. Herrn Exner gebührt ferner mein Dank für die Erarbeitung des externen Gutachtens über diese Arbeit.

Ich bedanke mich weiterhin bei unseren Kooperationspartnern von der Palacký Universität, Olomouc (Tschechische Republik), Dr. Lukáš Spíchal, Václav Mik und Markéta Gemrotová, die experimentelle Ergebnisse zu Isopentenyltransferasen bereitstellten. Václav Mik danke ich besonders für den freundlichen und effizienten Austausch während seines Aufenthaltes am IPB in Halle.

Mein Kollege Dimitar Vasilev hat projektbezogen zahlreiche chemische Synthesen durchgeführt. Dafür möchte ich mich herzlich bedanken. Ich schätze auch unsere zielgerichteten fachlichen Diskussionen sehr.

Michael Dressel hat im Rahmen seiner wissenschaftlichen Tätigkeit am IPB Modifikationen am PLANTS-Quellcode vorgenommen, die mir in dieser Arbeit von großem Nutzen waren. Ich bedanke mich für diese wertvolle Vorarbeit.

Dr. Frank Broda bot mir am IPB auf verschiedenen Ebenen Hilfe und Diskussionsbereitschaft und für seine wichtigen Einsätze danke ich ihm sehr.

Meine Kolleginnen und Kollegen Dr. Susanne Aust, Juliane Fischer, Julia Kufka, Dr. Silke Pienkny, Dr. Diana Schulze, Eva Schulze, Jennifer Szczesny, Dr. Stephanie Tennstedt, Tobias Heintz, Thomas Herberg, Peter-Paul Heym und Felix Rausch waren immer für mich da und haben mir durch ihre Ausgestaltung des wissenschaftlichen und zwischenmenschlichen Miteinanders am IPB sehr viel Freude bereitet. Dafür danke ich ihnen von ganzem Herzen.

Ich bedanke mich weiterhin bei allen Doktoranden und anderen Diskussionspartnern am IPB und im Umfeld der Martin-Luther-Universität Halle-Wittenberg, die sich mit mir austauschten und indirekt auch zum Gelingen dieser Arbeit beitrugen.

Als finanzierenden Institutionen danke ich dem Land-Sachsen-Anhalt, der Leibniz-Gemeinschaft, der Deutschen Bundesstiftung Umwelt und der Max-Buchner-Forschungstiftung.

Schließlich möchte ich mich bei meiner Familie bedanken, die mich stets ermuntert und motiviert hat, obwohl sie viel auf mich verzichten musste. Viel Kraft schöpfte ich auch aus meinen „Hallunken“, die abseits der Arbeit meine Inspiration waren. Schließlich gilt all das besonders für meine Freundin Nicole. Dafür sage ich von ganzem Herzen: „Danke!“

INHALTSVERZEICHNIS

INHALTSVERZEICHNIS	I
SUMMARY	VI
ABKÜRZUNGSVERZEICHNIS	VIII
HINWEISE ZUR NOTATION	IX
ABBILDUNGSVERZEICHNIS.....	X
TABELLENVERZEICHNIS	XIII
GLEICHUNGSVERZEICHNIS	XIV
GLOSSAR.....	XV
1 EINLEITUNG.....	1
1.1 Prenyltransferasen und Terpensynthasen	1
1.1.1 Klassifikation	1
1.1.1.1 Funktion	1
1.1.1.2 Struktur	3
1.1.1.3 Reaktionsmechanismus.....	6
1.1.1.4 Phylogenie	9
1.1.2 Ökologische Bedeutung	10
1.1.3 Bedeutung für den Menschen	11
1.1.4 Isoprenoidbiosynthese.....	12
1.1.5 Inhibierung.....	14
1.1.6 Theoretische Untersuchungen	15
1.1.7 Beispiele für prenylierende Enzyme	16
1.1.7.1 Limonensynthase aus <i>Mentha spicata</i>	16
1.1.7.2 Isopentenyltransferasen aus <i>Arabidopsis thaliana</i>	18
1.1.7.3 Kaurensynthase-ähnliche Enzyme.....	20
1.2 Molekulares <i>Docking</i>	22
1.2.1 Definition	23
1.2.2 Das <i>Docking</i> -Problem.....	24
1.2.2.1 Komplexität des <i>Docking</i> -Problems	24
1.2.2.2 Posengenerierung – <i>Sampling</i>	25
1.2.2.3 Posenbewertung – <i>Scoring</i>	25
1.2.3 Historische Entwicklung.....	26
2 MOTIVATION UND ZIELSTELLUNG	28
2.1 Das wissenschaftliche Interesse an prenylierenden Enzymen.....	28

2.2 Molekulares <i>Docking</i>	28
2.3 Zielstellungen dieser Arbeit.....	29
3 METHODEN	31
3.1 Homologiemodellierung	31
3.1.1 Homologiemodellierung mit YASARA.....	31
3.1.2 Isopentenyltransferasen aus <i>Arabidopsis thaliana</i>	32
3.1.2.1 Voruntersuchungen.....	33
3.1.2.2 Vorbereitungen für die Homologiemodellierungen	33
3.1.2.3 Bewertung der erstellten Proteinmodelle	34
3.1.2.4 Vergleich der Homologiemodelle	34
3.1.3 Kaurensynthase-ähnliche Enzyme	35
3.1.3.1 Voruntersuchungen.....	35
3.1.3.2 Bewertung der erstellten Proteinmodelle	35
3.1.3.3 <i>Docking</i> von Substratstrukturen und strukturelle Verfeinerung	35
3.1.3.4 Mutationsvorschläge und Aktivitätsuntersuchungen	36
3.2 Suche nach Inhibitoren für <i>Arabidopsis thaliana</i> Isopentenyltransferasen	36
3.2.1 Pharmakophorsuche	36
3.2.2 Vorschlag und Verifikation potenzieller AtIPT-Liganden.....	37
3.3 Molekulares <i>Docking</i> von Reaktionsintermediaten in <i>Mentha spicata</i> Limonensynthase und <i>Solanum habrochaites</i> Santalen- und Bergamotensynthase	38
3.3.1 Das <i>Docking</i> -Programm PLANTS	38
3.3.1.1 Erzeugung von <i>Docking</i> -Posen.....	39
3.3.1.2 Diversifikationsheuristik	40
3.3.1.3 <i>Clustering</i> der <i>Docking</i> -Lösungen.....	40
3.3.1.4 <i>Scoring</i> in PLANTS	40
3.3.1.5 Parametrisierung von PLANTS	43
3.3.2 Versuch und Definitionen zur <i>Cluster</i> -Option	43
3.3.3 Modifikation der CHEMPLP- <i>Scoring</i> -Funktion.....	43
3.3.4 Informationsfluss	45
3.3.4.1 Informationsverwaltung	51
3.3.4.2 Verwendete Parameterkombinationen	51
3.3.4.3 Auswertung der <i>Docking</i> -Untersuchungen	52
3.3.5 Vorbereitung von Ligand- und Rezeptorstrukturen.....	52
3.3.6 Das Geranylkation in verschiedenen Kraftfeldern	54
3.4 Quantenchemische Berechnungen	54

3.4.1	Bildung von Santalen und Bergamoten	54
3.4.2	Wechselwirkung zwischen Karbokationen und Aromaten bzw. Ethylmethylsulfid	55
4	ERGEBNISSE UND DISKUSSION	56
4.1	Isopentenyltransferasen aus <i>Arabidopsis thaliana</i>	56
4.1.1	Homologiemodellierung	56
4.1.1.1	<i>Template</i> -Wahl aus Ergebnissen der Voruntersuchungen	56
4.1.1.2	Qualität der erstellten Modelle	58
4.1.1.3	Analyse der Substratbindung	59
4.1.1.4	Diskussion der Proteinmodelle und der experimentellen Ergebnisse	67
4.1.2	Suche nach Inhibitoren	69
4.1.2.1	Auswertung und Diskussion der Pharmakophorsuche	69
4.1.2.2	Ergebnisse der Inhibitionstests im Labor	70
4.1.2.3	Diskussion der Inhibitorsuche	71
4.2	Kauren-Synthase-ähnliche Enzyme	71
4.2.1	Voruntersuchungen	71
4.2.2	Homologiemodellierung	73
4.2.2.1	Qualität der erstellten Modelle	73
4.2.2.2	Diskussion der modellierten Enzym-Ligand-Komplexe	73
4.2.2.3	Mutationsvorschläge für NtABS und ShSBS	81
4.2.3	Ergebnisse der experimentellen Umsetzung der Mutationsvorschläge	82
4.2.4	Quantenchemische Berechnungen zur Bildung von Santalen und Bergamoten	84
4.2.4.1	Erste Zyklisierung	84
4.2.4.2	Folgezyklisierungen	85
4.2.4.3	Abschließende Deprotonierung	89
4.2.5	Integrierte Zusammenfassung und Diskussion der theoretischen und experimentellen Ergebnisse zu KSL-Enzymen	90
4.3	Molekulares <i>Docking</i> von Reaktionsintermediaten in <i>Mentha spicata</i> Limonensynthase und <i>Solanum habrochaites</i> Santalen- und Bergamotensynthase	93
4.3.1	Das Geranylkatation in verschiedenen Kraftfeldern	93
4.3.2	Quantenchemische Berechnungen zur Wechselwirkung zwischen Karbokationen und Aromaten bzw. Sulfiden	94
4.3.3	Versuch zur Abhängigkeit bei <i>Cluster-Docking</i>	96
4.3.4	Modifikation der <i>Scoring</i> -Funktion in PLANTS	97
4.3.5	Auswertung der <i>Docking</i> -Experimente	97
4.3.5.1	Bezeichnungen und Terminologie	98

4.3.5.2 <i>Docking</i> des α -Terpinylkations in das Limonensynthesemodell.....	98
4.3.5.3 <i>Docking</i> des (Z,Z)-Farnesylokations in das ShSBS-Modell	104
4.3.6 Diskussion der <i>Docking</i> -Experimente und ihrer Ergebnisse.....	126
4.4 Schlussfolgerungen und Stellungnahme zu den in der Zielstellung aufgeworfenen Fragen	129
5 ZUSAMMENFASSUNG	132
6 LITERATURVERZEICHNIS	134
ANHANG	144
A 1 Aminosäuresequenzen der AtIPT	144
A 2 Paarweise Sequenzidentität der AtIPT und sechs potenzieller <i>Template</i> - Strukturen	145
A 3 ProSA-web: z-Score-Darstellungen und empirisches Potenzial für AtIPT	145
A 4 Ramachandran-Diagramme der AtIPT-Modelle.....	148
A 5 Sequenz- <i>Alignment</i> der AtIPT-Modelle und ihrer beiden <i>Template</i> -Sequenzen	150
A 6 Aminosäuresequenzen der vier KSL-Enzyme	151
A 7 ProSA-web: z-Score-Darstellungen und empirisches Potenzial für KSL	152
A 8 Pharmakophorbeschreibung für die Suche nach AtIPT-Liganden	154
A 9 Formale Definition des ACO-Algorithmus	155
A 10 PLANTS-Konfigurationsdateien für Versuche zu <i>Cluster</i> -Option.....	156
A 11 Bash-Shell-Skript <code>transformer</code>	158
A 12 Bash-Shell-Skript <code>runPLANTS</code>	162
A 13 Bash-Shell-Skript <code>evaPLANTS</code>	165
A 14 Bash-Shell-Skript <code>metaPLANTS</code>	170
A 15 Bash-Shell-Skript <code>corDistScores</code>	176
A 16 Bash-Shell-Skript <code>violinPLANTS</code>	180
A 17 Bash-Shell-Skript <code>violinOverlay</code>	185
A 18 Bash-Shell-Skript <code>violinTops</code>	190
A 19 Bash-Shell-Skript <code>drawPLPs</code>	192
A 20 Bash-Shell-Skript <code>partPLANTS</code>	195
A 21 Bash-Shell-Skript <code>collectFeats</code>	198
A 22 Bash-Shell-Skript <code>calcDistances</code>	200
A 23 Bash-Shell-Skript <code>evaTransform_c</code>	204
A 24 Bash-Shell-Skript <code>evaTransform_e</code>	208
A 25 Bash-Shell-Skript <code>PPQdepict</code>	219
A 26 Liganddefinition α -Terpinylkation (.mol2-Datei)	225

A 27 Liganddefinition (<i>Z,Z</i>)-Farnesylierung (.mol2-Datei).....	226
A 28 Analysedateien für <i>Docking</i> -Komplexe	227
A 29 Beispiel für Ergebnisse der <i>Cluster</i> -Analysen.....	227
A 30 Korrelationen der Parametervariation (Experiment <i>R</i>)	229
A 31 Korrelationen der Parametervariation (Experiment <i>S</i>)	230
Anschließend:	
CURRICULUM VITAE	
VERÖFFENTLICHUNGEN	
EIDESSTATTLICHE ERKLÄRUNG	

SUMMARY

Terpenoids are a highly diverse class of natural compounds. Among other properties, they are interesting for applications in the pharmaceutical industry or as flavoring or odorant substances. Terpene synthases are involved in the biosynthesis of terpenoids and are often able to catalyze the formation of complex carbon skeletons. On the other hand, prenyl transferases are responsible for the modification of other molecules with prenyl moieties. Together, both enzyme classes form the foundation of a vast chemical diversity. A deeper understanding of the mechanisms steering the underlying catalyses will allow for modification and biotechnological use of the enzymes. This work is concerned with the modeling of prenyl transferases and terpene synthases and molecular docking studies of reaction intermediates. It is divided into three parts.

First, homology models for seven DMAPP:AMP/ADP/ATP transferases, AtIPT1 and AtIPT3 to AtIPT8, from *Arabidopsis thaliana* were created. The catalytic mechanism known for a related enzyme is transferred to the produced models and discussed in this context. A virtual screening yielded numerous suggestions for potential inhibitors of the modeled enzymes. Based on these results, 17 potential inhibitors of the modeled enzymes were selected for further investigation. The substances were chemically synthesized and subsequently tested for their activity on AtIPT1 by cooperation partners. The enzyme tests identified six mildly inhibitory compounds with IC₅₀ values between 300 μM and 1000 μM.

In the second part, homology models for the four kaurene synthase-like enzymes

- *ent*-kaurene synthase from *Arabidopsis thaliana*
- *cis*-abienol synthase from *Nicotiana tabacum*
- santalene and bergamotene synthase from *Solanum habrochaites*
- β-phellandrene synthase from *Solanum lycopersicum*

were produced. Using molecular docking, respective substrate structures were positioned in the enzymes' active sites. The resulting receptor-ligand complexes are the basis for a detailed and comparative discussion of aspects of enzyme catalysis. In addition to the modeling of the kaurene synthase-like enzymes, quantum chemical gas-phase calculations were conducted. The results of these calculations illustrate that intermediate carbocations can interact strongly with the free electrons of sulfur atoms of methionine residues during enzymatic catalysis. Moreover, aromatic systems of active site amino acids are also able to stabilize cationic intermediates.

Further quantum chemical calculations in gas-phase address the product formation in the multi-product enzyme santalene and bergamotene synthase. The results provide the picture of a very fine enzymatic modulation of product determination. Differing positions of reaction intermediates in the binding pocket lead to the formation of different products. Based on these modelings, suggestions for mutations of santalene and bergamotene synthase and *cis*-abienol synthase were deduced with the goals to gain insight into specific amino acids' functions and a stepwise functional exchange of both enzymes. Site-directed mutagenesis and subsequent activity tests were performed by cooperation partners. The outcomes supply a detailed impression of the product spectra of single mutants of santalene and bergamotene synthase and *cis*-abienol synthase. The product spectra of both enzymes are discussed within the context of the protein models.

The third part of this work deals with the modulation of the product spectrum of santalene and bergamotene synthase by interactions between ligands and the binding pocket. Molecular docking is investigated as a method to place cationic reaction intermediates in the respective enzyme active sites. For this purpose, the docking program PLANTS was extended by the explicit modeling of interactions between intermediary cations and special receptor atoms. These receptor atoms comprise sulfur atoms of methionine side chains and aromatic carbon atoms. By extensive docking experiments into the known structure of *Mentha spicata* limonene synthase and the modeled structure of santalene and bergamotene synthase, the newly modeled interactions are investigated. The consequences of using different potentials to describe the newly implemented interactions are explored using statistical approaches. Geometric quantities as well as terms of the scoring function for docking poses are analysed. These analyses imply that the randomized pose generation in PLANTS has a larger influence on pose scoring than the variation of parameters of the used interaction potentials. As a consequence, it is insensible to optimize single force field parameters for molecular docking without addressing other important aspects of intermolecular interactions. Such features include extended receptor flexibility, entropy and solvation.

The theoretical investigations in this work exclude that product determination in terpene synthases is explicable by molecular docking or quantum chemical calculations only. Rather, an integrated treatment of enzyme-ligand interactions is necessary that models molecular orbitals and represents the enormous conformational space realistically.

ABKÜRZUNGSVERZEICHNIS

Abkürzung	Bedeutung
ACO	<i>Ant colony optimization</i> (engl. Ameisenkolonieoptimierung)
AMP, ADP, ATP	Adenosinmono-/ di-/ triphosphat
AtIPT	<i>Arabidopsis thaliana</i> Isopentenyltransferase
AtKS	<i>Arabidopsis thaliana ent</i> -Kaurensynthase
CPP	Copolyldiphosphat, näher zu bezeichnen als <i>ent</i> - bzw. <i>syn</i> -Form
DMAPP	Dimethylallyldiphosphat
DMASPP	Dimethylallyl-S-thiolodiphosphat
DNA	<i>Deoxyribonucleic acid</i> (engl. Desoxyribonukleinsäure)
FPP	Farnesyldiphosphat
GPP	Geranyldiphosphat
GGPP	Geranylgeranyldiphosphat
HIIPT	<i>Humulus lupulus</i> Isopentenyltransferase
HMBPP	1-Hydroxy-2-methyl-2-(<i>E</i>)-butenyl-4-diphosphat
IPP	Isopentenylidiphosphat
IPT	Isopentenyltransferase
NMR	<i>Nuclear magnetic resonance</i> (engl. magnetische Kernspinresonanz)
NPP	Neryldiphosphat (<i>Z</i> -Geranyldiphosphat)
NtABS	<i>Nicotiana tabacum cis</i> -Abienolsynthase
PDB	<i>Protein Data Bank</i> (engl. Proteindatenbank)
PLANTS	<i>Protein-ligand ant system</i> (engl. Protein-Ligand-Ameisensystem)
PLP	<i>piecewise linear potential</i> (engl. stückweise lineares Potenzial)
RMSD	<i>Root-mean-square deviation</i> (engl. Wurzel der mittleren quadratischen Abweichung), siehe Gleichung 3, Seite 34
RNA	<i>Ribonucleic acid</i> (engl. Ribonukleinsäure)
ShSBS	<i>Solanum habrochaites</i> Santalen- und Bergamotensynthase
SIPHS	<i>Solanum lycopersicum</i> β -Phellandrensynthase
TbTS	<i>Taxus brevifolia</i> Taxadiensynthase
tRNA	Transfer-RNA

HINWEISE ZUR NOTATION

Für Aminosäuren werden sowohl die üblichen Ein- als auch Dreibuchstabenabkürzungen verwendet. Die Einbuchstabenkodierung „X“ bedeutet eine nicht näher spezifizierte Aminosäure. Um in dieser Arbeit eine inhaltliche Verwechslung zwischen den Modellierungen verschiedener Enzymklassen zu vermeiden, werden die unterschiedlichen Kodierungen abschnittsweise bzw. thematisch verknüpft gebraucht:

Einbuchstabenabkürzung:

- 4.1 Isopentenyltransferasen aus *Arabidopsis thaliana*







Dreibuchstabenabkürzung:

- 4.2 Kauren-Synthase-ähnliche Enzyme
- 4.3 Molekulares *Docking* von Reaktionsintermediaten in *Mentha spicata* Limonensynthase und *Solanum habrochaites* Santalen- und Bergamotensynthase

Wenn für Prenyldiphosphate Abkürzungen verwendet werden, sind im Allgemeinen die entsprechenden *E*-Isomere gemeint. Eine explizite Erwähnung der *E/Z*-Isomerie erfolgt nötigenfalls.

Farbverzeichnis für Atome

Im Bereich der Molekülmodellierung ist folgende farbliche Kodierung für Atomtypen üblich:

 Wasserstoff (H)	 Kohlenstoff (C)	 Stickstoff (N)
 Sauerstoff (O)	 Phosphor (P)	 Schwefel (S)

Abweichungen von diesem Farbschema in dieser Arbeit werden explizit angegeben.

ABBILDUNGSVERZEICHNIS

Abbildung 1: a) Isopren und b) die C ₅ -Elementarbausteine für Terpenoide und Prenylmodifikationen	1
Abbildung 2: Einteilung von Terpensynthasen und Prenyltransferasen	2
Abbildung 3: Modell der Taxadiensynthese aus <i>Taxus brevifolia</i>	3
Abbildung 4: Klasse-II- und Klasse-I-Reaktion von Abietadiensynthese aus <i>Abies grandis</i> . 7	
Abbildung 5: Reaktionsschema der Biosynthese von Lanosterol.....	8
Abbildung 6: Funktion und Lokalisation von Isoprensynthase, Mono-, Sesqui- und Diterpensynthasen.....	13
Abbildung 7: Inhibitoren diverser Monoterpensynthasen	14
Abbildung 8: Pharmazeutisch relevante Bisphosphonate, die FPP-Synthasen inhibieren. .15	
Abbildung 9: Vorgeschlagene Reaktionsmechanismen in <i>Mentha spicata</i> Limonensynthese	17
Abbildung 10: Strukturformeln von Cytokininen	18
Abbildung 11: IPT-katalysierte Reaktionen.....	19
Abbildung 12: Produkte der Kaurensynthese-ähnlichen Enzyme aus Reis.....	20
Abbildung 13: Zyklisierung von GGPP und Folgeschritte in der Biosynthese diverser Gibberelline	21
Abbildung 14: DMAPP als Beispiel für Freiheitsgrade eines Liganden bei molekularem <i>Docking</i>	25
Abbildung 15: Substrate und Hauptprodukte der vier betrachteten KSL-Enzyme	30
Abbildung 16: Relative Lage der Liganden aus HIPT und <i>Agrobacterium tumefaciens</i> IPT .33	
Abbildung 17: Grafische Darstellung (MOE) der verwendeten Pharmakophordefinition.....	37
Abbildung 18: PLP- <i>Scoring</i> -Funktion für die abstandsabhängige Bewertung atomarer Interaktionen.....	41
Abbildung 19: Für die <i>Docking</i> -Untersuchungen verwendete Kationen	52
Abbildung 20: Bindetasche der Limonensynthese	53
Abbildung 21: Bindetasche der ShSBS	54
Abbildung 22: Flexible Koordinaten- <i>Scans</i> zwischen α -Terpinylkation und Modellen für Aminosäureseitenketten	55
Abbildung 23: Schema einer Wechselwirkungsgeometrie zwischen α -Terpinylkation und zwei nukleophilen Gruppen.....	55
Abbildung 24: Sequenzähnlichkeitsbaum der AtIPT und potenzieller <i>Template</i> -Strukturen ..	57
Abbildung 25: Tertiärstrukturdarstellung der strukturell superpositionierten AtIPT-Modelle...58	
Abbildung 26: Segment A des Sequenz- <i>Alignment</i> der AtIPT und der beiden <i>Template</i> -Strukturen.....	60
Abbildung 27: Wechselwirkungen des Diphosphatrestes mit dem p-Loop in den AtIPT-Modellen.....	61
Abbildung 28: DMASPP-Komplexierung in der Kristallstruktur 2ZE7	62
Abbildung 29: Segmente B und C des Sequenz- <i>Alignment</i> der AtIPT und der <i>Template</i> -Strukturen.....	63

Abbildung 30: Ausschnitt aus der Bindetasche der Röntgenkristallstruktur der <i>A. tumefaciens</i> IPT.....	64
Abbildung 31: Segment D des Sequenz- <i>Alignment</i> der AtIPT und der <i>Template</i> -Strukturen	64
Abbildung 32: Segment E des Sequenz- <i>Alignment</i> der AtIPT und der <i>Template</i> -Strukturen	64
Abbildung 33: Ausschnitt aus der Bindetasche des Modells von AtIPT5.....	65
Abbildung 34: Segment F des Sequenz- <i>Alignment</i> der AtIPT und der <i>Template</i> -Strukturen	66
Abbildung 35: Segment G des Sequenz- <i>Alignment</i> der AtIPT und der <i>Template</i> -Strukturen	66
Abbildung 36: Ausschnitt aus der Bindetasche des Modells von AtIPT5.....	67
Abbildung 37: Zwanzig Beispiele für Strukturen von Treffern der Pharmakophorsuche.	69
Abbildung 38: Baumdarstellung der Sequenzähnlichkeit zwischen den zu modellierenden KSL-Enzymen und potenziellen <i>Templates</i>	72
Abbildung 39: Baumdarstellung der Sequenzähnlichkeit	72
Abbildung 40: Sequenz- <i>Alignment</i> -Ausschnitt mit dem metallbindenden DDXXD-Motiv.	74
Abbildung 41: <i>Alignment</i> -Abschnitt mit einem konservierten Glutamin	74
Abbildung 42: <i>Alignment</i> -Abschnitte mit dem zweiten Metallbindemotiv und weiteren Aminosäuren des hydrophoben Teils der Bindetasche.	75
Abbildung 43: Substratbindetasche im AtKS-Modell.....	76
Abbildung 44: Bindetasche im NtABS-Modell	77
Abbildung 45: Bindetasche im ShSBS-Modell	78
Abbildung 46: Bindetasche im SIPHS-Modell	80
Abbildung 47: Ausschnittsdarstellung eines Struktur- <i>Alignment</i> zwischen NtABS- und ShSBS-Modell.	81
Abbildung 48: Mögliche erste Schritte in der enzymatischen Produktion von Santalenen und Bergamotenen.	85
Abbildung 49: Theoretische formale Reaktionswege zu α - und β -Santalenen	86
Abbildung 50: Theoretische formale Reaktionswege zu α - und β -Bergamotenen.....	87
Abbildung 51: A) Relative quantenmechanische Energien der betrachteten Santalene und Bergamotene, B) Epimer von 4.....	88
Abbildung 52: Geranylkation mit Partialladungen nach Struktur-Optimierung mit TAFF-Kraftfeld	93
Abbildung 53: Ergebnisse flexibler Koordinaten- <i>Scans</i>	94
Abbildung 54: Quantenmechanisch optimierte Wechselwirkungsgeometrie des α -Terpinylkations zwischen 3-Methylindol und Ethylmethylsulfid.....	95
Abbildung 55: Einfluss der <i>Cluster</i> -Option auf die Variabilität von Ergebnissen	96
Abbildung 56: verwendete CHEMPLP-Varianten (im Experiment <i>L</i>).....	97
Abbildung 57: Dichte-Diagramm und <i>Boxplot</i> sowie p-Wert-Matrix der Verteilungen der Gesamt- <i>Scores</i> in ausgewählten (aro-) Potenzialvarianten zum Experiment <i>A</i>	99
Abbildung 58: Dichte-Diagramm und <i>Boxplot</i> sowie p-Wert-Matrix der Verteilungen der PLPpartsteric- <i>Score</i> -Anteile in ausgewählten (aro-) Potenzialvarianten zum Experiment <i>A</i>	100

Abbildung 59: Farbliche Veranschaulichung des jeweiligen Posenanteils im Experiment <i>A</i> , bei dem die betrachteten Wechselwirkungen innerhalb des <i>Score</i> -relevanten Distanzbereichs liegen.....	101
Abbildung 60: Rangkorrelation mit $PLP_{partsteric}$ und gemessenen Distanzen ausgewählter Wechselwirkungen im Experiment <i>A</i> (Bezeichnungen siehe Abbildung 59)	102
Abbildung 61: Korrelationen der Parametervariationen im Experiment <i>A</i>	103
Abbildung 62: Farbliche Veranschaulichung des jeweiligen Posenanteils in den Experimenten <i>A</i> , <i>B</i> und <i>E</i> , bei dem die betrachteten Wechselwirkungen innerhalb des <i>Score</i> -relevanten Distanzbereichs liegen.....	104
Abbildung 63: Dichte-Diagramm und <i>Boxplot</i> sowie p-Wert-Matrix der Verteilungen der Gesamt-Scores in ausgewählten (met-) Potenzialvarianten zum Experiment <i>C</i>	105
Abbildung 64: Dichte-Diagramm und <i>Boxplot</i> sowie p-Wert-Matrix der Verteilungen der $PLP_{partsteric}$ - <i>Score</i> -Anteile in ausgewählten (met-) Potenzialvarianten zum Experiment <i>C</i>	106
Abbildung 65: Auftragung der <i>Score</i> -Zusammensetzung für Posen, die mit dem Potenzial 0_-0.6 erzeugt wurden.....	107
Abbildung 66: Farbliche Veranschaulichung des jeweiligen Posenanteils im Experiment <i>C</i> , bei dem die betrachteten Wechselwirkungen innerhalb des <i>Score</i> -relevanten Distanzbereichs liegen.....	108
Abbildung 67: Rangkorrelation mit $PLP_{partsteric}$ und gemessenen Distanzen ausgewählter Wechselwirkungen im Experiment <i>C</i>	109
Abbildung 68: Räumliche Verteilung ausgewählter Repräsentanzpunkte in der ShSBS-Bindetasche.....	111
Abbildung 69: Korrelation der Parametervariation im Experiment <i>C</i>	112
Abbildung 70: Distanzen zwischen C.cat1 und dem α -Kohlenstoffatom von Gly634 und $PLP_{partsteric}$ - <i>Score</i> -Anteile entsprechend der Rangsortierung.....	114
Abbildung 71: Distanzen zwischen C.cat2 und Trp746 und $PLP_{partsteric}$ - <i>Score</i> -Anteile entsprechend der Rangsortierung.....	115
Abbildung 72: Auftragung der <i>Score</i> -Zusammensetzung.....	116
Abbildung 73: Farbliche Veranschaulichung des jeweiligen Posenanteils im Experiment <i>E</i> , bei dem die betrachteten Wechselwirkungen innerhalb des <i>Score</i> -relevanten Distanzbereichs liegen.....	117
Abbildung 74: Rangkorrelation mit $PLP_{partsteric}$ - <i>Score</i> -Anteil und Distanzen ausgewählter Wechselwirkungen im Experiment <i>R</i> (Bezeichnungen siehe Abbildung 59, Seite 101 und Abbildung 66, Seite 108)	119
Abbildung 75: Rangkorrelation mit $PLP_{partsteric}$ - <i>Score</i> -Anteil und Distanzen ausgewählter Wechselwirkungen im Experiment <i>S</i>	120
Abbildung 76: Ausschnitt aus der Top- <i>Docking</i> -Pose für das Potenzial -0.4_-0.4 im Experiment <i>S</i>	121
Abbildung 77: Vergleich von Distanzverteilungen über ausgewählte (met-) Potenzialvarianten	123
Abbildung 78: Vergleich von Distanzverteilungen über ausgewählte (aro-) Potenzialvarianten	125

TABELLENVERZEICHNIS

Tabelle 1:	Auswahl von Terpensynthasen mit veröffentlichter Röntgenkristallstruktur	4
Tabelle 2:	Ausgewählte Prenyltransferasen mit veröffentlichter Röntgenkristallstruktur ..	5
Tabelle 3:	Michaeliskonstanten für ausgewählte Substrate verschiedener IPT.....	19
Tabelle 4:	Genloкус-Identifikatoren der AtIPT für die NCBI-Datenbank.....	32
Tabelle 5:	Verwendete Einstellungen für das <i>Docking</i> mit GOLD	35
Tabelle 6:	Kurzcharakterisierung der Strukturdatenbanken für die Pharmakophorsuche	36
Tabelle 7:	Eingabevariablen für das Hauptskript <code>transformer</code>	44
Tabelle 8:	Verwendete Bash-Shell-Skripte und die jeweils erzeugten Dateien.	45
Tabelle 9:	Bezeichner für die Bildung von in Tabelle 8 angegebenen Dateinamen.....	50
Tabelle 10:	Parameterkombinationen für die <i>Docking</i> -Versuche	51
Tabelle 11:	Homologiemodellierungs- <i>Templates</i> für AtIPT	56
Tabelle 12:	Übersicht über zu <i>Arabidopsis thaliana</i> DMAPP:ATP/ADP-Transferasen homologen Proteinstrukturen.....	57
Tabelle 13:	YASARA-z-Scores (1.-4. Zeile) und ProSA-web-z-Scores.....	58
Tabelle 14:	RMSD in Å zwischen den AtIPT-Modellen	59
Tabelle 15:	YASARA-z-Scores (1.-4. Zeile) und ProSA-web-z-Scores.....	73
Tabelle 16:	Im <i>Alignment</i> einander entsprechende Aminosäuren der Bindetaschen von NtABS und ShSBS.	81
Tabelle 17:	Mutationsvorschläge für ShSBS und hypothetische Konsequenzen	82
Tabelle 18:	Ergebnisse von Umsetzungsversuchen an ShSBS und ShSBS-Mutanten mit verschiedenen Substraten	82
Tabelle 19:	Ergebnisse von Umsetzungsversuchen an NtABS und NtABS-Mutanten mit verschiedenen Substraten	84
Tabelle 20:	Die Nomenklatur von Santalenen und Bergamotenen	89
Tabelle 21:	Anzahl der bei Termination des PLANTS-Algorithmus zurückgegebenen <i>Docking</i> -Posen (Experiment <i>H</i> bzw. Experiment <i>J</i>).....	113
Tabelle 22:	Gesamt-Scores in diversitätsorientierten vs. unabhängigen <i>Docking</i> -Experimenten	118

GLEICHUNGSVERZEICHNIS

Gleichung 1: Berechnung eines <i>z-Score</i>	32
Gleichung 2: Berechnung von <i>z-Score</i> „Insgesamt“ in YASARA.....	32
Gleichung 3: Berechnung des RMSD für zwei Atommengen <i>A</i> und <i>B</i>	34
Gleichung 4: Optimierungsproblem der <i>Scoring</i> -Funktion.....	39
Gleichung 5: Aufschlüsselung der CHEMPLP- <i>Scoring</i> -Funktion aus PLANTS	42
Gleichung 6: Definition der Wahrscheinlichkeit der Realisierung eines bestimmten Wertes für einen Freiheitsgrad.....	155
Gleichung 7: Aufdatierung der Pheromonvektoren	156
Gleichung 8: Berechnung der Iterationsanzahl für den PLANTS-Algorithmus.....	156
Gleichung 9: Definition des Gültigkeitsintervalls für Pheromonwerte	156

GLOSSAR

Die wissenschaftliche Literatur im Bereich der Molekülmodellierung, virtuellen Inhibitorsuche und Quantenchemie ist von einigen englischsprachigen Begriffen geprägt, die sich nur schwer bedeutungsgleich ins Deutsche übertragen lassen. Solche Begriffe werden hier kurz erläutert, da sie in der Arbeit verwendet werden.

Alignment	Hier im engeren Sinne: Das algorithmische Aufeinanderpassen von Aminosäuresequenzen (<i>Sequenz-Alignment</i>) bzw. Sekundär- oder Tertiärstrukturen von Proteinen, dann <i>Struktur-Alignment</i>
Boxplot	Grafische Darstellung der Verteilung einer oder mehrerer Variablen als Rechteck über einer Skala. Das Rechteck begrenzt jeweils oberes und unteres Quartil der Variablen. Der Median der aufgetragenen Größe wird mit einem Strich markiert. Vom Rechteck ausgehend werden jeweils durch zwei mit einem Strich abgeschlossene „Antennen“ Variablenwerte innerhalb des (hier) anderthalbfachen Quartilabstandes dargestellt. Werte jenseits dieser Grenzen werden als Punkte dargestellt und gelten als Ausreißer im Sinne des <i>Boxplot</i> .
Clustering	Auch <i>Cluster-Analyse</i> , Verfahren zur Entdeckung und Beschreibung von Ähnlichkeiten in Daten
Docking	Hier im engeren Sinne des molekularen <i>Docking</i> : die Modellierung von Wechselwirkungen zwischen Rezeptor- und Ligandmolekülen unter Benutzung einschlägiger Software
Heatmap	Zweidimensionale Darstellung von Daten, die farblich repräsentiert werden
Sampling	Hier im engeren Sinne: Auswahl von Stichproben konkreter Konformationen eines Molekülensembles aus einem Konformationsraum
Scan	Abtastung, hier im engeren Sinne einer schrittweisen Untersuchung von Energiehyperflächen
Scoring	Hier im engeren Sinne: Bewertung der Konformation eines Liganden in einem Rezeptor; <i>Score</i> entsprechend: Zahlenwert einer solchen Bewertung
Screening	Hier im engeren Sinne: Das mehr oder weniger systematische Untersuchen von Hypothesen zu molekularen Interaktionen durch entweder theoretische Methoden (<i>virtual screening</i>) oder Labormethoden, wie Bindungs- und Umsetzungsstudien
Template	Vorlage, hier im engeren Sinne eine experimentell ermittelte Proteinstruktur

1 EINLEITUNG

Terpenoide bilden mit mehreren zehntausend Vertretern die größte Naturstoffklasse.¹ Sie sind im Bereich des Lebens ubiquitär verbreitet und spielen wichtige Rollen in grundlegenden metabolischen Prozessen.² Auch in komplexe ökologische Zusammenhänge sind sie involviert.¹ Prenyltransferasen und Terpensynthasen sind die biokatalytischen Eltern dieser Chemodiversität.

1.1 Prenyltransferasen und Terpensynthasen

Prenyltransferasen sind Enzyme, die Übertragungen von Prenylresten – aktivierten Isoprenen – katalysieren. Terpensynthasen sind ein Spezialfall der Prenyltransferasen, wenn eine intramolekulare Prenylrestverknüpfung erfolgt.³ Der wesentliche Unterschied zwischen beiden Enzymklassen besteht darin, dass echte Prenyltransferasen ein Akzeptormolekül prenylieren, während Terpensynthasen Ein-Substrat-Enzyme sind und nach enzymatischer Aktivierung eine formal intramolekulare Reaktion katalysieren.

Prenyltransferasen und Terpensynthasen haben ein gemeinsames Substratspektrum: Beide Enzymklassen setzen Prenyldiphosphate als chemisch aktivierte Prenylketten unterschiedlicher Länge um. Es gibt jedoch zusätzlich Substrate für Terpensynthasen, die über keinen Diphosphatrest verfügen.⁴ Die Elementareinheiten für die Biosynthese längerer Prenyldiphosphatketten sind in Abbildung 1 dargestellt.

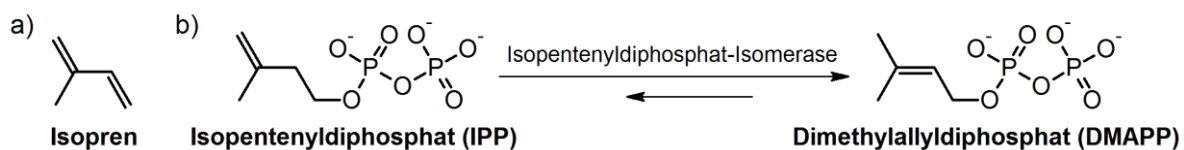


Abbildung 1: a) Isopren und b) die C₅-Elementarbausteine für Terpenoide und Prenylmodifikationen, Isopentenyl diphosphat (IPP) und Dimethylallyl diphosphat (DMAPP)⁵. Der terminale Phosphatrest wird als β-Phosphatrest und der Phosphatrest zwischen β-Phosphat und organischem Rest als α-Phosphatrest bezeichnet.

Modifikationen an Terpenen erweitern die Chemodiversität dieser Substanzklasse und erlauben feine Modulationen der molekular vermittelten Informationen.⁶

1.1.1 Klassifikation

Enzyme sind nach verschiedenen Gesichtspunkten klassifizierbar. Der folgende Abschnitt ordnet Prenyltransferasen und Terpensynthasen bezüglich Funktion (EC-Nomenklatur), Struktur, chemischer Mechanistik und phylogenetischer Aspekte.

1.1.1.1 Funktion

Prenyltransferasen werden nach dem Nomenklaturschema für Transferasen benannt, so dass eine Donor: Akzeptor-Transferase die Übertragung der Donor-Gruppe auf die Akzeptor-Gruppe bewerkstelligt.⁷ In der Praxis wird jedoch oft vereinfachend zu Donor-Transferase abgekürzt, zum Beispiel Isopentenyl-Transferase und Geranyl-Transferase. Auch andere Trivialnamen bzw. Schreibweisen werden verwendet, beispielsweise Farnesylproteintransferase für ein Enzym, das Farnesylgruppen auf Proteine überträgt (eigentlich Farnesyl:Protein-Transferase). In der EC-Nomenklatur gehören Prenyltransferasen zur

Klasse „EC 2.5.1 – Enzyme, welche Alkyl- oder Arylreste übertragen, die keine Methylgruppen sind“.⁷

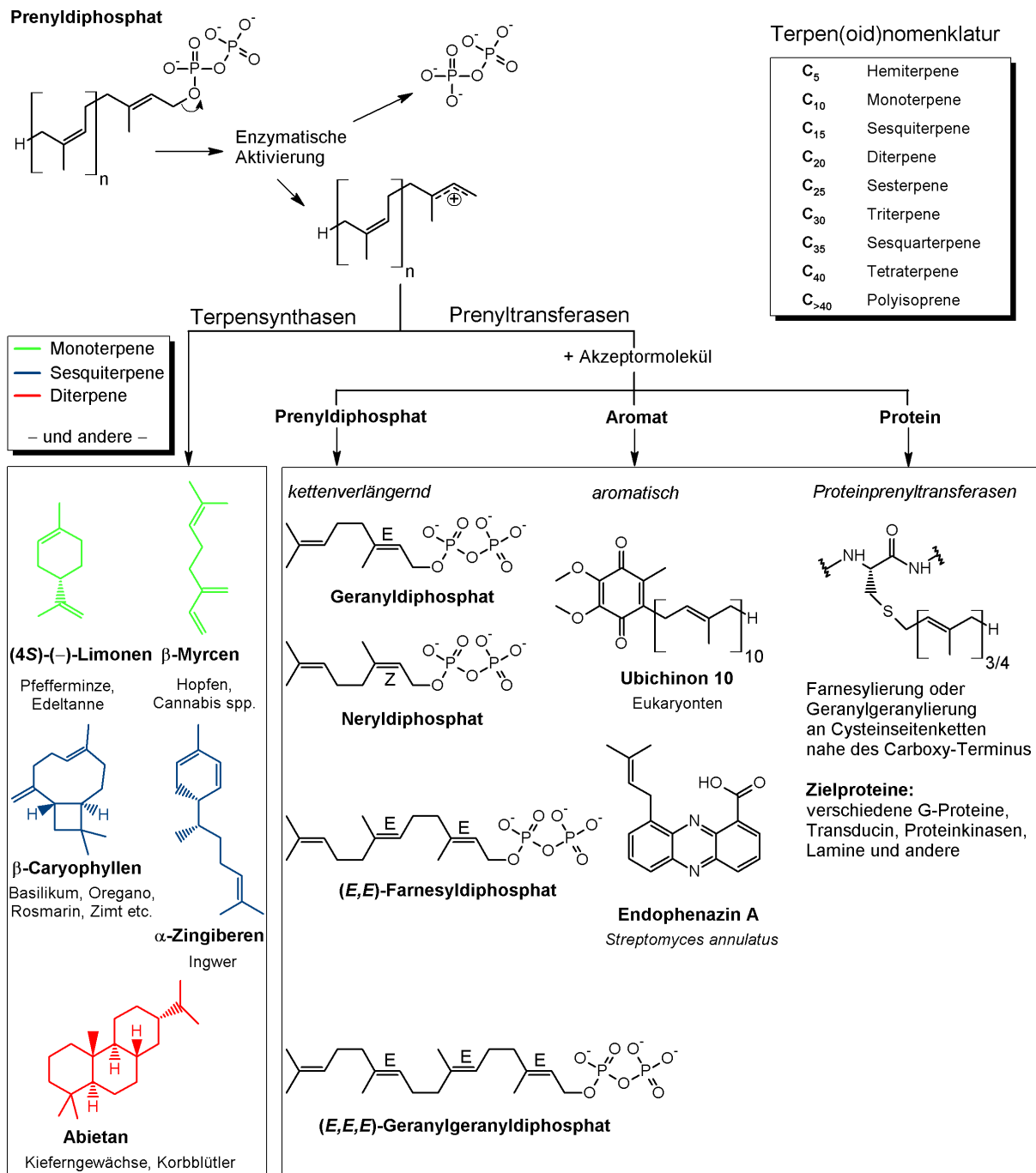


Abbildung 2: Einteilung von Terpensynthesen und Prenyltransferasen mit jeweiligen Beispielen für Reaktionsprodukte (und natürliche Quellen,^{8–20}) sowie Nomenklatur der Terpenoide nach der Anzahl der Isopreneinheiten

Prenyldiphosphate mit mehreren Prenyleinheiten werden enzymatisch durch Kondensationsreaktionen gebildet.^{21,22} Dabei werden C₅-Einheiten verknüpft, um länger-kettige Prenyldiphosphate oder Terpene zu erhalten, was in Abbildung 2 veranschaulicht wird. Die Anzahl der Kohlenstoffatome ist dabei jeweils ein Vielfaches von fünf, was als sogenannte C₅-Regel oder Isoprenregel (Wallach 1914, siehe dazu²³) bezeichnet wird. Diese Regel kann dabei helfen, biosynthetische Ursprünge von neu identifizierten Naturstoffen aufzuklären. Die Verknüpfung der Isopreneinheiten kann sowohl in *E*- als auch in *Z*-Form erfolgen, was ein

zusätzliches Klassifizierungsmerkmal für kettenverlängernde Prenyltransferasen darstellt, weil diese ihre Substrate in der Regel isomerenspezifisch akzeptieren und umsetzen. Kettenverlängernde *trans*-Prenyltransferasen bilden im Allgemeinen Produkte bis C₅₀. *Cis*-Prenyltransferasen katalysieren hingegen den Aufbau längerer Polymere.²⁰ Weiterhin wird bei der Isoprenkettenverlängerung nach dem Ort der Bindungsknüpfung zwischen den beiden Prenyleinheiten zwischen Kopf-zu-Kopf- und Kopf-zu-Schwanz-Verknüpfungen (1-4-Kondensationen) unterschieden.²¹

Terpensynthasen sind oft Multiproduktenzyme und werden meistens nach dem Produkt mit der höchsten Ausbeute *in vitro* bzw. dem zuerst identifizierten Reaktionsprodukt *in vivo* benannt, wie Limonensynthase und Kaurensynthase. Abbildung 2 bietet nur ansatzweise einen Eindruck davon, welche Diversität Kohlenstoffgerüste mit wachsender Zahl von Isopreneinheiten annehmen können. Entsprechend der EC-Nomenklatur gehören viele Terpensynthasen in die Klassen „EC 4.2.3 – phosphatumsetzende Enzyme“ und „EC 5.4.99 – intramolekulare Transferasen von weder Acyl-Gruppen, noch Phosphat-, Amino- oder Hydroxylgruppen“. Es gibt Terpensynthasen, die Nichtzyklisierungsreaktionen katalysieren (siehe beispielsweise^{24–26}). Daher sind Terpenzyklasen als eine Untergruppe der Terpensynthasen zu verstehen.

1.1.1.2 Struktur

Über strukturelle Domänen in Terpensynthasen und ihre katalytische Bedeutung bieten Köksal *et al.* einen hervorragenden Überblick.²⁷

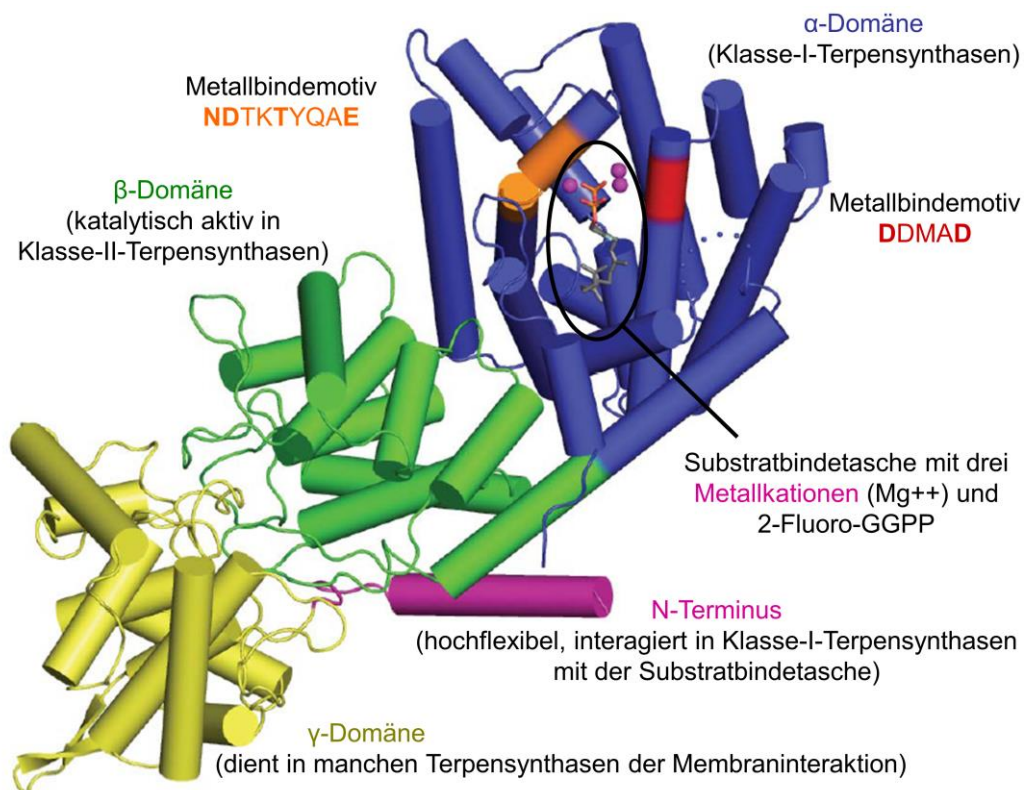


Abbildung 3: Modell der Taxadiensynthase aus *Taxus brevifolia* als Beispiel für den modularen Aufbau von Terpenzyklasen (modifiziert von Köksal *et al.*,²⁷). Fettgedruckte Aminosäuren sind direkt an der Metallkomplexierung beteiligt. Verschiedene Terpensynthasen besitzen nur Teile dieser Tertiärstruktur, siehe auch Abschnitt 1.1.1.4 Phylogenie, Seite 9 ff.

Es wird zwischen Klasse-I- und Klasse-II-Terpensynthasen unterschieden, die verschiedene Reaktionsmechanismen bedingen (siehe Abschnitt 1.1.1.3 Reaktionsmechanismus, Seite 6 ff.).^{28,29} Die strukturellen Grundlagen der diversen Eigenschaften schildert Abbildung 3. Klasse-I-Terpensynthasen verfügen über zwei konservierte Metallbindemotive mit den Konsensussequenzen **DDXXD** und **(N,D)DXX(S,T)XXXE**, die in der Regel drei divalente Metallkationen, meist Magnesium, komplexieren (fettgedruckt: die typischerweise metallkomplexierenden Aminosäuren).²⁷ Viele der bisher charakterisierten Terpensynthasen aus Pflanzen und Pilzen sind als Homodimere aktiv (^{25,30–33}), es gibt aber offenbar Ausnahmen.³⁴

So ist die große Mehrheit der kurze *cis*-Prenyldiphosphate synthetisierenden Enzyme als Homodimer aktiv, während einige Geranyldiphosphatsynthasen als Heterodimere aktiv sind.^{35–37} Es ist bemerkenswert, dass auch diese kettenverlängernden Prenyltransferasen (aber nicht polymerisierenden, sondern sog. *short chain prenyl transferases*, engl., Kurzketten-Prenyltransferasen) wie Farnesylidiphosphatsynthase und Geranylgeranyldiphosphatsynthase eine katalytisch aktive Domäne besitzen, die der α -Domäne der Klasse-I-Terpensynthasen ähnelt.³⁸ Diese Domäne besteht aus 12 α -Helices, die ein sogenanntes α_6/α_6 -*Barrel*-Faltungsmotiv bilden. Einen Überblick zur Domänenstruktur in Terpensynthasen bietet Tabelle 1.

Tabelle 1: Auswahl von Terpensynthasen mit veröffentlichter Röntgenkristallstruktur. Verwendete Farbcodierung: ■ Mono-, ■ Sesqui-, ■ Di- und □ Triterpensynthasen. Spalte "D" beinhaltet die jeweils vorhandenen Terpensynthasedomänen entsprechend Köksal *et al.*²⁷ und Abbildung 3, Seite 3. PDB (engl. *Protein Data Bank*, Proteindatenbank) steht für den Identifikator in der Proteindatenbank.³⁹ Mit * markierte Enzyme katalysieren die Bildung von Prenyldiphosphaten. Die aufgeführten Triterpensynthasen sind Transmembranproteine (1EZF) oder membranassoziiert (1SQC, 1W6K) und verfügen über *all*- α -Domänen mit entfernter Ähnlichkeit zur entsprechenden Domäne in Mono-, Sesqui- und Diterpensynthasen.

Terpensynthase	Organismus	D	PDB
4S-Limonensynthase	<i>Mentha spicata</i>	$\alpha\beta$	2ONH ¹²
Bornyldiphosphatsynthase*	<i>Salvia officinalis</i>	$\alpha\beta$	1N1B ³¹
1,8-Cineolsynthase	<i>Salvia fruticosa</i>	$\alpha\beta$	2J5C ⁴⁰
<i>epi</i> -Isozizaensynthase	<i>Streptomyces coelicolor A3</i>	α	3KB9 ⁴¹
(+)- δ -Cadinensynthase	<i>Gossypium arboreum</i>	$\alpha\beta$	3G4F ⁴²
Aristolochensynthase	<i>Aspergillus terreus</i>	α	3BNX ⁴³
5- <i>epi</i> -Aristolochensynthase	<i>Nicotiana tabacum</i>	$\alpha\beta$	5EAU ^{44,45}
Abietadiensynthase	<i>Abies grandis</i>	$\alpha\beta\gamma$	3S9V ⁴⁶
Taxadiensynthase	<i>Taxus brevifolia</i>	$\alpha\beta\gamma$	3P5R ²⁷
<i>ent</i> -Copalyldiphosphatsynthase*	<i>Arabidopsis thaliana</i>	$\alpha\beta\gamma$	3PYA ⁴⁷
Squalenzyklase	<i>Alicyclobacillus acidocaldarius</i>	-	1SQC ⁴⁸
Squalensynthase	<i>Homo sapiens</i>	-	1EZF ⁴⁹
Lanosterolsynthase	<i>Homo sapiens</i>	-	1W6K ⁵⁰

Interessanterweise ist die Abietadiensynthese aus *Abies grandis* durch katalytisch aktive Klasse-I- und Klasse-II-Terpensynthasedomänen bifunktional.^{46,51} In Klasse-II-Terpensynthasen befindet sich das katalytische Zentrum in der β -Domäne, am Rande zur γ -Domäne.²⁷ Es beinhaltet ein DXDD-Motiv, welches direkt zur Katalyse beiträgt.⁵² Tri- und Tetraterpensynthasen wurden von Wendt und Schulz mechanistisch als Klasse-II-Terpensynthasen eingeordnet.²⁸ Die Biosynthese größerer Isoprenpolymere erfolgt durch Deka-, Undeka- und weitere *cis*-Prenyldiphosphatsynthasen, die als Transmembranproteine charakterisiert wurden und nur noch entfernte Ähnlichkeit mit den Terpensynthasen kleinerer Produkte aufweisen.^{20,53} Es sei darauf hingewiesen, dass namentliche Terpenzyklen in ihrem Produktspektrum auch nichtzyklisierte Terpene oder Terpenalkohole wie Geraniol oder Farnesol aufweisen können.^{24–26}

Tabelle 2: Ausgewählte Prenyltransferasen mit veröffentlichter Röntgenkristallstruktur. Verwendete Farbkodierung: ■ Mono-, ■ Sesqui- und ■ Diterpenttransferasen. Das mit * markierte Enzym katalysiert die Übertragung von Triterpenen und Flavonoiden auf Glycosylreste. PDB steht für den Identifikator in der Proteindatenbank.³⁹

Prenyltransferase	Organismus	PDB
tRNA-Isopentenyltransferase	<i>Saccharomyces cerevisiae</i>	3EPH ⁵⁴
Adenylatisopentenyltransferase	<i>Humulus lupulus</i>	3A8T ⁵⁵
Farnesylproteintransferase	<i>Rattus norvegicus</i>	1QBQ ⁵⁶
Farnesyldiphosphatsynthase	<i>Gallus gallus</i>	1FPS ⁵⁷
Rab-Geranylgeranyltransferase	<i>Rattus norvegicus</i>	1DCE ⁵⁸
Geranylgeranyldiphosphatsynthase	<i>Saccharomyces cerevisiae</i>	2DH4 ⁵⁹
multifunktionsell*	<i>Medicago truncatula</i>	2ACV ⁶⁰

Die strukturellen Merkmale von Prenyltransferasen sind weniger konserviert als bei Terpensynthasen. Allerdings katalysiert diese Enzymklasse auch ein breiteres Reaktionsspektrum.

Aromatische Prenyltransferasen die an der Biosynthese von Ubichinon, Menachinon und Plastochinon beteiligt sind wurden als Membranproteine beschrieben.^{61,62} Sie verfügen über konservierte (N,D)DXXD-Motive zur Komplexierung von Prenyldiphosphaten über Magnesiumionen.⁶² Erste Strukturvorschläge wurden von Bräuer *et al.* beschrieben.^{63,64}

An der Biosynthese mikrobieller Sekundärstoffe beteiligte aromatische Prenyltransferasen sind lösliche Enzyme ohne die bekannten aspartatreichen Motive.⁶² Während sich deren Vertreter CloQ und NphB von zum Beispiel pflanzlichen DMAPP:Indol-Transferasen in der Sequenz stark unterscheiden, verfügen alle über eine *barrel*-Architektur mit zehn innen liegenden antiparallelen β -Faltblättern, die von zehn α -Helices umgeben sind (ABBA- oder $(\alpha\beta\beta)_5$ -Faltung).^{62,65}

Die DMAPP:tRNA-Transferase aus *Saccharomyces cerevisiae* überträgt einen Prenylrest auf Nukleotide von tRNA-Molekülen.⁵⁴ Das Enzym besteht aus einer Kerndomäne und verfügt über ein flexibel verknüpftes Zinkfinger-Motiv zur Erkennung der tRNA.⁵⁴ Dieser Kerndomäne ähnelt die Struktur der Adenosinphosphat-Isopentenyltransferase aus *Agrobacterium tumefaciens*.⁶⁶ Entfernte Ähnlichkeiten weist auch die Adenylatisopentenyltransferase aus *Humulus lupulus* (HIPT) auf.⁶⁶ Strukturelle Ähnlichkeiten zu experimentellen Enzym-

strukturen wurden untersucht: Ein ähnliches *p-Loop*-Motiv (engl. *loop*, Schleife – eine Sekundärstruktur mit flexibler räumlicher Ordnung; ein *p-Loop* dient der Phosphaterkennung) wie in der HIPT ist in der Familie der Nukleosidtriphosphat-Hydrolasen beschrieben. Auch C-terminale Struktur motive der HIPT finden sich in dieser Familie wieder.⁶⁶

Proteinprenylierung wird durch drei verschiedene Gruppen von Prenyltransferasen katalysiert: Farnesyl:Protein-Transferase, Geranylgeranyl:Protein-Transferase I und Geranylgeranyl:Protein-Transferase II.⁶⁷ Trotz sehr ähnlicher Faltungen unterscheiden sie sich in der Akzeptorspezifität.⁶⁷ Ein Übersichtsartikel von Maurer-Stroh *et al.* zeigt ferner strukturelle Verwandtschaften zu Terpensynthasen auf, obwohl die Sequenzähnlichkeit zwischen diesen Enzymklassen gering ist. Auch strukturelle Bedingungen zur Wechselwirkung mit Akzeptorproteinen werden beschrieben.⁶⁷

1.1.1.3 Reaktionsmechanismus

Auf welche Weise Enzyme chemische Reaktionen katalysieren, ist eng mit der Frage danach verbunden, wie sie mit ihren Substraten und Kofaktoren wechselwirken. Zielgerichtete Mutagenesen bestimmter Aminosäuren von Enzymen und anschließende Umsatzuntersuchungen können die Involvierung dieser Aminosäuren am Reaktionsmechanismus nachweisen. Untersuchungen mit (zum Teil radiomarkierten) Substrat- und Intermediat analoge liefern potenziell Hinweise auf die elektronischen Prozesse vom Substrat zum Produkt. Als Reaktionsmechanismus einer Enzymkatalyse ist im weiteren Sinne die Gesamtheit von Wechselwirkungen und tatsächlichen Umsetzungen im Enzym-Ligand-Komplex zu verstehen. Degenhardt und Kollegen verfassten 2009 einen Übersichtsartikel über Mono- und Sesquiterpensynthasen, worin zahlreiche Untersuchungen zu den Reaktionsmechanismen dieser Enzyme zusammengefasst sind.⁶

Die Enzymkatalyse von Monoterpensynthasen wird durch die metallionenabhängige Ionisierung des Substrates eingeleitet. Das resultierende kationische Intermediat kann verschiedene Zyklisierungs- und Umlagerungsschritte durchlaufen, bis die Reaktion durch Deprotonierung oder die Addition eines Nucleophils – beispielsweise Wasser – formal endet. Diese Funktionsweise wurde in den 1980er Jahren durch Croteau mit Hilfe von Substrat- und Intermediat analoge sowie Inhibitoren an mehreren Monoterpensynthasen belegt.⁶⁸⁻⁷⁴ Eine Umlagerung des Diphosphatrestes vom Geranyl- zum Linalyldiphosphat und die Isomerisierung von der transoiden zur cisoiden Form führen zur Bildung des Linalylkations. Erst dieses Intermediat kann zum α -Terpinylkation zyklisieren, welches Ausgangspunkt für zyklische Monoterpene ist. Azyklische Monoterpene können mechanistisch sowohl über das Geranyl- als auch das Linalylkation gebildet werden. Für eine Veranschaulichung dieser formalen Abläufe sei auf die entsprechenden Darstellungen in Degenhardt *et al.*⁶ verwiesen.

Ähnlich dieser Mechanismen verläuft auch die Enzymkatalyse in Sesquiterpensynthasen, wobei zwischen der anfänglichen Bildung des Nerolidylkations und des (*E,E*)-Germacradienylkations unterschieden wird. Durch die zusätzliche Isopreneinheit kommt jedoch insgesamt eine deutlich größere Produktvielfalt zu Stande.⁶ Ringkontraktionen, 1,2-Methyl- und Wagner-Meerwein-Umlagerungen erweitern die grundsätzliche Diversität.⁷⁵⁻⁷⁷ Bemerkenswert ist die Enzymkatalyse mancher Sesquiterpensynthasen wie 5-*epi*-Aristolochensynthase, die über neutrale Intermediate verläuft.^{6,78} Durch eine Reprotonierung wird bei solchen Mechanismen erneut ein Kation gebildet wonach weitere Zyklisierungs- und Umlagerungsschritte ablaufen können, wie in den Katalysen von Vetaspiradiensynthase⁷⁵, Valencensynthase⁷⁹ und β -Selinensynthase⁸⁰. Köllner und Kollegen zeigten für zwei

Sesquiterpensynthesen aus Mais, dass für die Reptonierung Wassermoleküle als Protonendonoren fungieren.⁸¹ Diese Säure/Base-katalysierten Enzymreaktionen werden auch von Miller und Allemann diskutiert, die über eine Beteiligung von Tyrosinseitenketten oder des Diphosphatrestes als Protonendonoren spekulieren.⁸² Sie führen ferner aus, dass die Komplexierung des Substratmoleküls Farnesylidiphosphat (FPP) mit einem Magnesiumion vermutlich eine Vororientierung des Diphosphatrestes für die Bindetasche bewirkt, die zur anschließenden Rekrutierung eines zweiten Magnesiumions beiträgt. Dieser Prozess geht – in der Modellvorstellung – davon aus, dass anschließend die Substratbindetasche durch Konformationsänderungen weitgehend vom Lösungsmittel abgeschirmt wird, bis die Rekrutierung eines dritten Magnesiumions weitere Konformationsänderungen auslöst, die zur Reaktionsinitiation führen. Miller und Allemann gehen von einer relativ starren Positionierung des Diphosphatrestes in der Bindetasche während des Reaktionsverlaufs aus, während im hydrophoben Teil der Bindetasche die Transformation des kationischen Intermediats verläuft.⁸²

Mono- und Sesquiterpensynthesen sind als Klasse-I-Terpensynthesen definiert, weil ihr Reaktionsmechanismus durch die ionische Abspaltung des Diphosphatrestes beginnt.²⁸ Auch für Diterpensynthesen werden analoge Reaktionswege beschrieben.^{82–84} Allerdings erlaubt die um die γ -Domäne erweiterte Tertiärstruktur (siehe Tabelle 1, Seite 4) zusätzliche Funktionalität, wie sie bei *Abies grandis* Abietadiensynthese nachgewiesen wurde.⁴⁶ Die bifunktionale Abietadiensynthese katalysiert zweistufig die Bildung allylischer Abietadien-Doppelbindungsisomere aus Geranylgeranylidiphosphat (GGPP), wie in Abbildung 4 dargestellt. Dabei treten folgende Teilschritte auf (siehe auch Peters und Croteau,^{85,86}):

- Klasse-II-Reaktion: In einer Bindetasche zwischen β - und γ -Domäne findet die protonierungsinitiierte Zyklisierung des GGPP zu Copolyldiphosphat (CPP) statt.
- Klasse-I-Reaktion: In der α -Domäne wird der Diphosphatrest abgespalten und nach einer weiteren Zyklisierung und Deprotonierung entstehen Abietadiene.

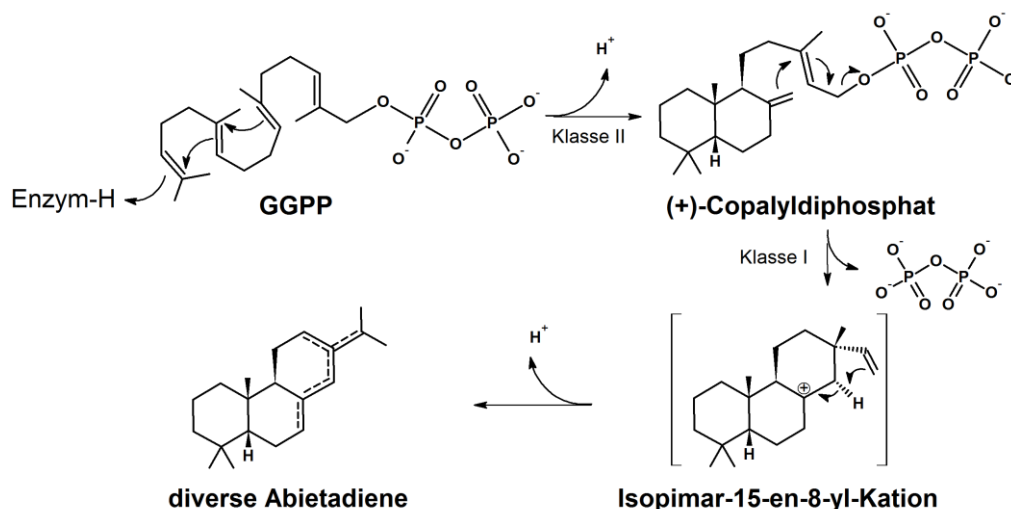


Abbildung 4: Klasse-II- und Klasse-I-Reaktion von Abietadiensynthese aus *Abies grandis*. Abbildung modifiziert nach Zhou *et al.*⁴⁶

Einige in die Katalyse involvierte Aminosäureseitenketten wurden mit zielgerichteter Mutagenese identifiziert.⁴⁶

Sehr komplexe Kohlenstoffgerüste entstehen bei Umsetzungen von GGPP mit Phomactatriensynthase und Taxadiensynthase.⁸⁷ Die Mechanismen wurden mit Hilfe von Deuterium-markiertem GGPP und quantenchemischen Berechnungen aufgeklärt.⁸⁷

Triterpensynthasen benutzen Squalen oder Oxidosqualen als Substrat für Zyklisierungsreaktionen.^{82,83,88,89} Bei diesen Klasse-II-Terpensynthasen erfolgt die Aktivierung des Substrates durch Protonierung, wie in Abbildung 5 veranschaulicht wird.

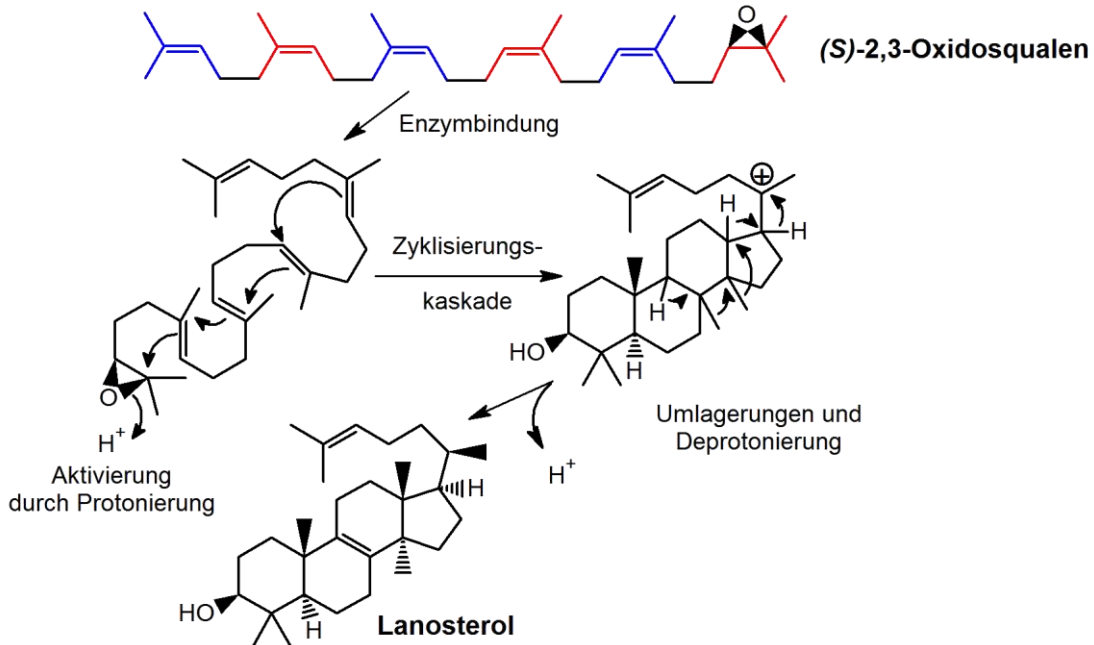


Abbildung 5: Reaktionsschema der Biosynthese von Lanosterol aus dem Triterpenoid Oxidosqualen durch Lanosterolsynthase.⁴ Dieses Beispiel zeigt eine Aktivierung des Prenylrestes ohne Diphosphatabspaltung aber mit zahlreichen Zyklisierungen und Umlagerungen, wie sie für Terpensynthasen typisch sind. Die Färbung des Substratmoleküls hebt die sechs Prenylreste hervor.

Ein Übersichtsartikel von Liang et al. aus dem Jahr 2002 bietet einen grundlegenden Überblick über mechanistische Aspekte von Prenyltransferasen.³

Bei kettenverlängernden Prenyltransferasen wird grundsätzlich zwischen den *trans*- (oder *E*-) und den *cis*- (oder *Z*-) Prenyltransferasen differenziert, die sich schon in ihrer Aminosäuresequenz stark unterscheiden.⁹⁰ Bereits frühe Untersuchungen zeigten eine Metallionenabhängigkeit dieser Enzyme.⁹¹ Die jeweiligen Mechanismen unterscheiden sich jedoch nicht nur in der bevorzugten Abstraktionsposition eines Protons.^{90,92} Unter Benutzung reaktionsverlangsamender Substratanaloga wurde an Oktaprenyltransferase gezeigt, dass in *trans*-Prenyltransferasen ein sequentieller Ablauf von Ionisierung, Kondensation und Eliminierung auftritt.²⁷ Für *cis*-Prenyltransferasen konnte dieser Ansatz am Beispiel der Undekaprenyltransferase nicht nachvollzogen werden, so dass Lu *et al.* hier vom Vorliegen eines konzertierten Ablaufs der Reaktionsschritte ausgehen.^{93,94} Die Polymerisierung von Isopren zum klassischen Naturkautschuk im Latex von *Hevea brasiliensis* (Kautschukbaum) erfolgt durch *cis*-Prenyltransferasen.⁹⁵ Das natürliche Isoprenpolymer Guttapercha des Baumes *Palaquium gutta* ist jedoch *trans*-verknüpft. Sehr bemerkenswert sind aktuelle Forschungsergebnisse von Frick *et al.*, die am Beispiel einer Prenyldiphosphatsynthase aus *Phaedon cochleariae* beweisen, dass der metallische Kofaktor – in diesem Falle Cobalt-, Mangan- bzw. Magnesiumionen – die Kettenlänge des Katalyseproduktes beeinflusst. Somit

kann die Art des vorliegenden Metallions zu Diversifizierung und Regulation des Terpenoidstoffwechsels beitragen.⁹⁶

Als ein Beispiel für aromatische Prenyltransferasen soll das Enzym CloQ dienen. Zu CloQ publizierten Metzger *et al.* 2010 eine Enzym-Ligand-Struktur und interessante mechanistische Studien, entsprechend derer in der Bindetasche dieses Enzyms keine Metallionen, sondern positiv geladene Aminosäureseitenketten für die entsprechende Polarisierung der C-O-Bindung zwischen Diphosphat und Prenylrest sorgen.⁶² Der anschließende Angriff des Prenylkations am aromatischen Kosubstrat entspricht dem Formalismus einer Friedel-Crafts-Alkylierung.

Proteinprenyltransferasen bilden Thioetherverknüpfungen zwischen Cysteinseitenketten des Proteins und dem entsprechenden Prenylkation.⁹⁷ Bei der Ras-Farnesyltransferase wird hierzu die Cysteinseitenkette durch ein Metallion zum Thiolat aktiviert, so dass sie als Nukleophil für die Verknüpfung mit dem Farnesylkation dient, welches sich durch Ionisierung des FPP bildet. Diese Ionisierung wird durch positiv geladene Aminosäureseitenketten in der Nähe des Diphosphatrestes begünstigt.

1.1.1.4 Phylogenie

Terpensynthasen und Prenyltransferasen sind die Biokatalysatoren für die Bildung von Molekülen im gesamten Spektrum zwischen Primärmetabolismus und ökologisch hochspezialisierten Substanzen wie Boten-, Lock- und Abwehrstoffen. Die Phylogenetik dieser Enzymklassen ist besonders in Pflanzen und Mikroorganismen sehr facettenreich.

Für Terpensynthasen gilt allgemein, dass es zwar geringe Sequenzhomologie, aber starke mechanistische und strukturelle Ähnlichkeiten gibt.^{27,82} Ein umfangreicher Übersichtsartikel von Chen *et al.* charakterisiert die mutmaßliche Geschichte pflanzlicher Terpensynthasen und ihrer Diversifikation.⁹⁸ Die Klassifikation entsprechend des Reaktionsmechanismus in Klassen I und II hat ihre Ursache in strukturellen Gegebenheiten, die wiederum auf der Aminosäuresequenz der entsprechenden Enzyme beruhen. Die meisten bekannten pflanzlichen Terpensynthasen gehören zur Klasse I, in welcher die enzymatische Reaktion durch die Ionisierung des Prenyldiphosphates eingeleitet wird. Das Moos *Physcomitrella patens* verfügt über eine bifunktionale Terpensynthase, deren N-terminale Domäne Ähnlichkeiten zu Klasse-II-Terpensynthasen aufweist, während der C-terminale Bereich Klasse-I-Terpensynthasen ähnelt (vgl. Abbildung 3, Seite 3). Es wird heute davon ausgegangen, dass solche Enzyme evolutionäre Vorläufer für später herausgebildete Klasse-I- und Klasse-II-Terpensynthasen sind. Sequenzanalysen in Gymnospermen und Angiospermen weisen darauf hin, dass es zu einer Duplikation des Gens für ein solches bifunktionales Enzym gekommen ist, dessen zwei Ausführungen sich anschließend evolutionär unterschiedlich spezialisierten, und sich so die beiden Funktionalitäten trennten.⁹⁸ Während in Pilzen auch die bifunktionalen Enzyme nachgewiesen wurden, verfügt das Leguminosen-Symbiose-Bakterium *Bradyrhizobium japonicum* über zwei Terpensynthasen mit jeweils nur einer der beiden Funktionen.^{99,100} Einige Gymnospermen verfügen über bifunktionale Diterpensynthasen, in denen sowohl die Klasse-I-(α -) als auch die Klasse-II-(β -) Domäne aktiv sind. In Angiospermen wurden bisher nur monofunktionale Diterpensynthasen gefunden, die entweder der Klasse I oder der Klasse II angehören. Bisher wurden nur Mono- und Sesquiterpensynthasen entdeckt, die ebenfalls dieser Monofunktionalität unterliegen. In diesen fehlt in vielen Fällen die β -Domäne ganz oder teilweise, was Chen *et al.* als mögliche evolutionäre Anpassungen deuten.^{98,101,102}

Cao und Kollegen⁽⁵²⁾ entwickelten die $\alpha/\beta/\gamma$ -Nomenklatur, welche auch Köksal *et al.* aufgriffen (vgl. Abbildung 3, Seite 3 und²⁷). Sie fanden ferner für bakterielle Klasse-II-Diterpensynthasen in den Sequenzen von Farnesyl- und Geranylproteintransferasen und Triterpensynthasen Motiventprechungen, die auf eine verwandte β/γ -Struktur hindeuten.⁵² Für pflanzliche Diterpensynthasen schlugen sie allgemein eine $\alpha/\beta/\gamma$ -Architektur vor, basierend auf der Theorie, dass Mono- und Sesquiterpensynthasen (oft mit α/β -Struktur) in Pflanzen durch Exonverlust und Rekombination aus ursprünglichen Diterpenzyklasen hervorgehen, vgl. Tabelle 1 (Seite 4). Mit der Aufklärung der Struktur der bifunktionalen Abietadiensynthase aus *Abies grandis* durch Zhou *et al.* konnte diese Theorie nachvollzogen werden.⁴⁶ Der „Verlust“ der γ -Domäne in Mono- und Sesquiterpensynthasen fand nachweislich mehrfach und unabhängig voneinander statt, was ein Anhaltspunkt für die Komplexität der Mechanismen ist, die der Selektion dieser Enzyme zu Grunde liegen.¹⁰³

Über phylogenetische Zusammenhänge kettenverlängernder Prenyltransferasen mit Terpenzyklasen oder Proteinprenyltransferasen ist wenig bekannt. Proteinfarnesyl- und Proteingeranylgeranyl-Transferasen sind sich grundsätzlich sehr ähnlich.⁶⁷ Sie bestehen aus einer α - und einer β -Untereinheit, wobei Letztere der α_6/α_6 -*Barrel*-Domäne aus Terpensynthasen strukturell stark ähnelt.⁶⁷

1.1.2 Ökologische Bedeutung

Die hohe Komplexität der biochemischen Information in Terpenoiden spiegelt sich in vielfältigen ökologischen Zusammenhängen wider. Gershenzon und Dudareva beleuchten die ökologische Bedeutung von Terpenoiden in einem Übersichtsartikel anhand einiger Beispiele.¹

- Pathogenabwehr bei Pflanzen:
 - Insektenlarven der Spezies *Trichoplusia ni* vergiften sich an den Cardenoliden von *Asclepias curassavica* (Indianer-Seidenpflanze)¹⁰⁴, siehe auch^{105,106}.
 - *Arabidopsis thaliana*-Pflanzen, die durch eine genetische Transformation übernatürlich viel Linalool, einen Monoterpenalkohol, produzieren, werden signifikant weniger von *Myzus persicae* (Grüne Pfirsichblattlaus) als Nahrungsquelle benutzt.¹⁰⁷
 - Mit Methyljasmonat, das in Pflanzen Pathogenreaktionen auslöst, behandelte *Picea abies* (Gemeine Fichte) steigern den Gehalt von Mono- und Diterpenen im Harz des Stammes, was einen geringeren Befall durch *Ips typographus* (Buchdrucker-Käfer) zur Folge hat.¹⁰⁸
 - Pflanzlich produzierte, triterpenoide Saponine zerstören die Integrität von Pilzmembranen und schützen so vor Pilzinfektionen¹⁰⁹, während *Avena strigosa*-Pflanzen, bei denen die Fähigkeit zur Produktion solcher Substanzen fehlt, eine deutlich schlechtere Immunität aufweisen.¹¹⁰
- Abwehrsekrete von Insekten:
 - Iridoide, eine Klasse von Monoterpenoiden, sind wichtige Abwehrstoffe verschiedener Ameisen- und Käferarten gegen Fraßfeinde. Die biosynthetische Herkunft der Abwehrsekrete beinhaltet in der Regel die pflanzliche Nahrung.^{111–115}

- Chemische Wehrstoffe in Meeresorganismen (siehe auch ¹¹⁶):
 - Grünalgen (*Caulerpa* spp.) werden von eigentlichen Fraßfeinden, verschiedenen Seesternarten, verschmäht. Verantwortlich dafür ist u.a. Caulerpenin, ein von den Grünalgen produziertes Sesquiterpenoid.¹¹⁷
 - Andere Grünalgen (*Halimeda* spp.) zeigen eine schnelle, verwundungsaktivierte Bereitstellung von Halimedatrial, einem Diterpenoid, das Fraßfeinde abschreckt.^{118,119}
 - Viele sesshafte Meeresorganismen überziehen ihre Oberfläche mit komplexen Stoffgemischen. Diese beinhalten verschiedene Terpene mit antifungaler und antibakterieller Wirkung und verhindern auch Algenbewuchs ¹²⁰, siehe auch ¹²¹.

1.1.3 Bedeutung für den Menschen

Über diese erst wenig erforschte Vielfalt an ökologischen Interaktionen hinaus haben Terpene auch für den Menschen eine große Bedeutung. Sie stellen einen wesentlichen Anteil der Naturstoffe, die vom Menschen gustatorisch oder olfaktorisch wahrgenommen werden.¹²² Dies zeigt sich beispielsweise in unterschiedlichen Geschmackswahrnehmungen agrarwirtschaftlich zum Teil sehr bedeutsamer Pflanzenarten und -unterarten mit verschiedenen Terpenoidprofilen wie Wein^{123,124}, Möhren¹²⁵ oder Süßkartoffeln¹²⁶. Ferner kommen Terpene häufig in großen Mengen in Pflanzen vor, die als Gewürz verwendet werden. Beispiele dafür sind:

- Sesquiterpene und Sesquiterpenlaktone aus verschiedenen *Illicium*-Arten (Sternanis)^{127–131}
- Zahlreiche Mono- und Sesquiterpene aus *Piper nigrum* (Schwarzer Pfeffer)^{8,132}
- Monoterpene und Monoterpenalkohole aus *Mentha* spp. (Minze)^{9,133,134}
- Das Monoterpene-Keton Carvon aus *Carum carvi* (Kümmel), *Anethum graveolens* (Dill) und *Mentha spicata* (Grüne Minze)¹³⁵
- Viele Mono- und Sesquiterpene und ihre Alkohole aus *Zingiber officinale* (Ingwer)^{10,136,137}
- Über 30 eindeutig identifizierte Sesquiterpene aus *Ocimum basilicum* (Basilikum)¹¹
- Das ätherische Öl aus *Oreganum vulgare* (Oregano) wird von Monoterpenen dominiert (zum Teil Linien mit über 90% Gewichtsanteil von Monoterpenen), wobei die anteilige Zusammensetzung zwischen verschiedenen Linien stark variiert.¹³⁸

Bei der Extraktion von ätherischen Ölen aus Pflanzen sind Terpene oft wesentliche Bestandteile. Seit Jahrhunderten werden diese Öle wegen ihrer besonderen Eigenschaften verwendet.¹³⁸ Mit modernen wissenschaftlichen Methoden sind ätherischen Ölen zum Beispiel antibakterielle, antifungale, antiinflammatorische oder spasmolytische Wirkungen nachgewiesen worden.^{138–146} Zunehmend gibt es auch Untersuchungen, die Terpenbiosynthese gezielt zu beeinflussen, um Eigenschaften beispielsweise in Pflanzen zu modifizieren oder bestimmte Terpene anzureichern.^{147–151} Ein Beispiel hierfür ist der genetische Eingriff in die Biosynthese des toxischen Terpenoid-Aldehyds Gossypol in Baumwollpflanzen, so dass transgene Pflanzen durch eine Reduktion des Gossypol-Gehalts potenziell auch als Tierfutter verwendet werden könnten.¹⁵² Einen aktuellen Überblick über das sog. *pathway engineering* (engl., Konstruktion von Synthesewegen) zur biotechnologischen Produktion verschiedener Terpenklassen bietet Misawa.¹⁵³

Mehrere aktuelle Übersichtsartikel beschäftigen sich mit der Bedeutung von Proteinprenyltransferasen in pharmazeutischen Kontexten. Die posttranslationalen Modifikationen der Farnesylierung und Geranylgeranylierung von Substratproteinen wie zum Beispiel G-Proteinen kann Signalwirkung für ganze Biosynthesewege haben.^{154,155} Dies ist nicht auf den Säugetiermetabolismus beschränkt, sondern im Pflanzen- und Tierreich weit verbreitet. Für die Blockade von Proteinprenyltransferasen werden oft Inhibitoren eingesetzt, die den Substraten Farnesyldiphosphat und Geranylgeranyldiphosphat ähneln.^{156–160} Die Prenylierung von Proteinen hat oft Einfluss auf ihre Lokalisation, weil hydrophobe Prenylreste als Membrananker dienen können.¹⁶¹

Der wohl berühmteste Wirkstoff gegen Krebs – Paclitaxel – wurde 1967 erstmals isoliert und ist ein hochsubstituierter Naturstoff, der sich von dem Diterpen Taxadien ableitet.¹⁶² Die natürliche Quelle ist die sehr langsam wachsende Pazifische Eibe (*Taxus brevifolia*), was die Erforschung alternativer Anreicherungsverfahren interessant und im Laufe der Zeit auch lukrativ machte.¹⁶² So entwickelten sich für Paclitaxel nach semisynthetischen Produktionsverfahren moderne Fermentationsprozesse mit Pflanzenzell- und Pilzkulturen.^{163–165} Die enorme Bedeutung von Paclitaxel war Motivation für detaillierte Untersuchungen über den molekularen Wirkmechanismus, der darauf beruht, dass die Substanz die Mikrotubuli in sich teilenden Zellen so sehr stabilisiert, dass die Chromosomen nicht die für die Mitose notwendige räumliche Umverteilung erfahren können und die Zelle entweder Apoptose einleitet oder zurück in ein vegetatives Stadium verfällt.^{166–171} Auch zahlreiche Osteoporosewirkstoffe beruhen auf der Inhibition prenylierender Enzyme, siehe Abbildung 8 (Seite 15).¹⁷²

Abgesehen von vielfältigen Wirkweisen exogener Terpenoide, sind Terpensynthasen und Prenyltransferasen wichtige Enzyme im Primärmetabolismus fast aller Lebensformen. Terpenoide und spezifisch verknüpfte Prenylreste sind unabdingbare Informationsträger des Lebens und die weitere Entschlüsselung ihrer Produktion, Perzeption und Modulation biochemischer Prozesse ist eine andauernde Herausforderung für die Wissenschaft.

1.1.4 Isoprenoidbiosynthese

Die Biosynthese von Isoprenoiden beginnt mit dem Anabolismus der grundlegenden aktivierten Isoprene IPP und DMAPP aus Stoffen des Primärmetabolismus. Eukaryonten und Archaea benutzen hierfür den Mevalonatweg (MVA-Weg), der auch einigen Bakterien zur Verfügung steht.^{173,174} In den meisten Bakterien wird die Bereitstellung von IPP und DMAPP über den Methylerythritolphosphatweg (MEP-Weg) erreicht, der auch in Pflanzen und Algen existiert und im Gegensatz zum zytosolischen MVA-Weg in den Plastiden stattfindet.¹⁷⁵ Untersuchungen von Sapir-Mir *et al.* zeigten, dass Teile des MVA-Weges auch in den Peroxisomen vonstattengehen.¹⁷⁶ Auf diese Weise werden IPP und DMAPP in diesen Organellen gebildet, so dass eine Farnesyldiphosphatsynthase (zumindest bei Säugetieren) die Kondensation von IPP und DMAPP zu FPP im Peroxisom bewerkstelligt.¹⁷⁶ In einem Übersichtsartikel hinterfragen Hemmerlin *et al.* die Notwendigkeit für Pflanzen, zwei verschieden lokalisierte Isoprenoidbiosynthesewege zu besitzen und diskutieren anhand zahlreicher Beispiele die Bedeutung dieser Tatsache für die Möglichkeiten von Pflanzen, mit ihrer Umgebung zu interagieren.¹⁷⁷

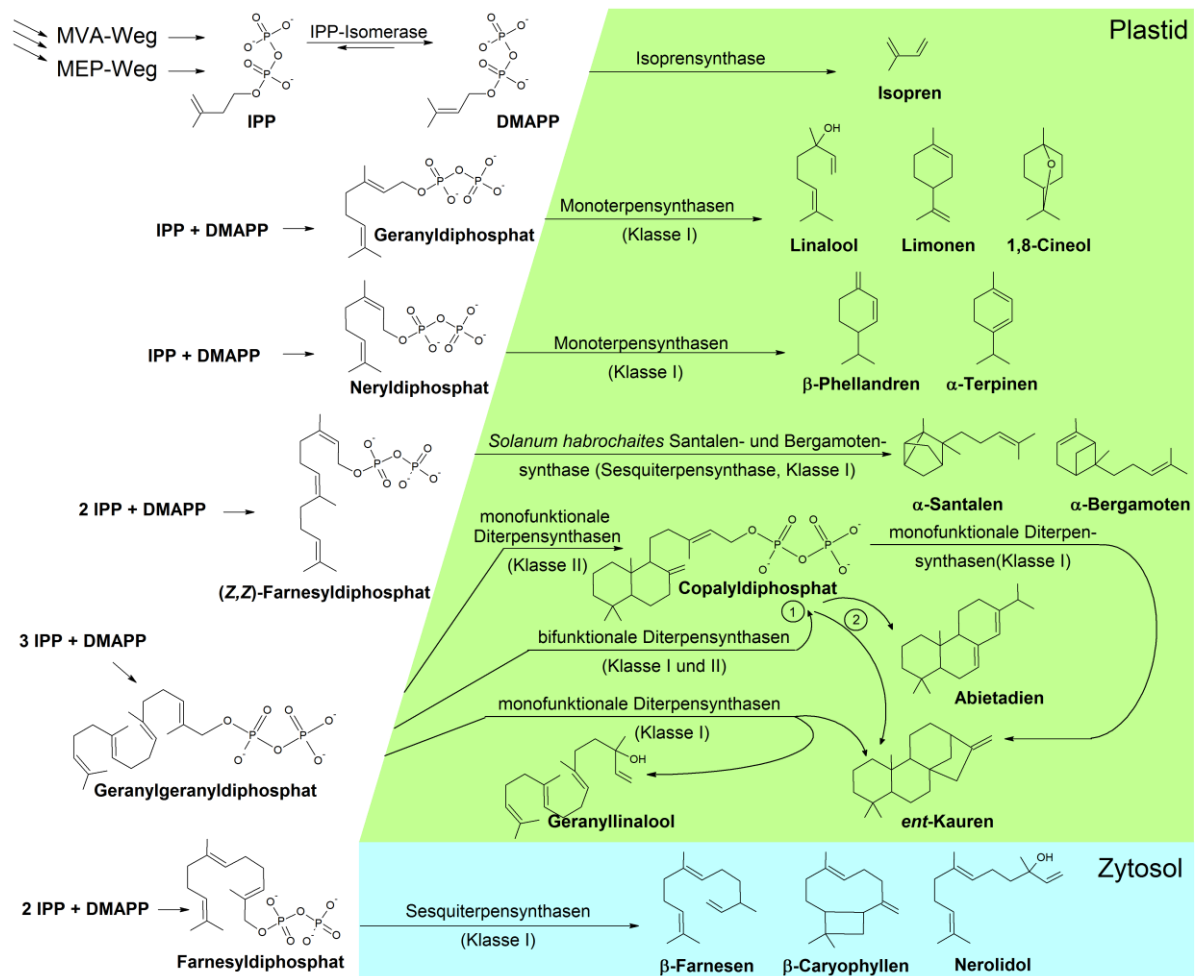


Abbildung 6: Funktion und Lokalisation von Isoprensynthase, Mono-, Sesqui- und Diterpensynthasen, Abbildung nach Fig. 1 in Chen *et al.*,⁹⁸. Die Kondensationsreaktionen zu Prenyldiphosphaten (links) werden durch entsprechende Synthasen sukzessiv katalysiert. Die explizit angeführte *S. habrochaites* Santalen- und Bergamotensynthase stellt insofern eine Ausnahme dar, als dass Sesquiterpensynthasen normalerweise zytosolisch vorliegen.¹⁰¹

Die Terpen-Elementarbausteine IPP und DMAPP werden durch Prenyldiphosphatsynthasen wie FPP-Synthase und GGPP-Synthase um jeweils eine C₅-Einheit verlängert, wie in Abbildung 6 angedeutet. Dabei kann es zu Interferenzen zwischen MVA- und MEP-Weg kommen, deren Natur erst zum Teil erforscht wurde.^{178–180} Die sukzessive enzymatisch bereitgestellten Prenyldiphosphate werden durch zahlreiche Terpensynthasen umgesetzt. Das Schema verdeutlicht, dass die *E/Z*-Isomerie der verschiedenen Substrate von Bedeutung für die Umsetzung durch isomerenspezialisierte Enzyme ist. Die Santalen- und Bergamotensynthase aus *Solanum habrochaites* ist bisher die einzige bekannte Sesquiterpensynthase, die spezifisch (*Z,Z*)-FPP umsetzt und im Plastiden vorkommt.

Squalen ist im Allgemeinen eine Zwischenstufe für die Bildung von Triterpenen und wird durch die Squalensynthase aus zwei FPP-Molekülen gebildet.⁴⁹ Die Bildung längererkettiger Prenyldiphosphate wird von weiteren Enzymen, zum Beispiel Dekaprenyldiphosphatsynthase und Solanesyldiphosphatsynthase, bewerkstelligt, wobei diese Enzyme in der Regel die Kondensation an Prenylketten bestimmter Länge bevorzugen, aber eine gewisse Substratpromiskuität besitzen.^{90,181,182}

Die Bildung pflanzlicher Polyisoprene wie Kautschuk oder Guttapercha wird von spezialisierten *cis*- bzw. *trans*-Prenyltransferasen katalysiert, die IPP als Prenyldonor benutzen, um Polyisopren um jeweils eine weitere C₅-Einheit zu verlängern.^{183,184} Dabei bilden sich Polymerpartikel in zur Latexproduktion spezialisierten Zellen der entsprechenden Pflanzen.^{185,186}

1.1.5 Inhibierung

Die Inhibierung von Enzymen kann in verschiedenen Kontexten interessant sein. Während sie in der Grundlagenforschung zur Ermittlung der biologischen Funktion oder biokatalytischen Funktionsweise von Enzymen eingesetzt wird, bieten sich auch pharmazeutische Anwendungsmöglichkeiten.

Für die Inhibierung von Terpensynthasen wurden bisher hauptsächlich Substanzen benutzt, die durch Ähnlichkeit zu natürlichen Liganden Aufschluss über verschiedene Aspekte der Substratbindetasche geben sollten, so zum Beispiel Phenyl-substituiertes Farnesyl-diphosphat an Aristolochensynthase¹⁸⁷, S-Thiolo-Farnesyldiphosphat sowie Chinuclidine und Chinuclidinole an Dehydrosqualen- und Squalensynthase¹⁸⁸, S-Alkyl-Thiodiphosphate an Farnesyldiphosphatsynthase und verschiedene Substratanaloga von Monoterpensynthasen an Selbigen.^{74,189–191} Einige Beispiele für Inhibitoren von Monoterpensynthasen sind in Abbildung 7 dargestellt.

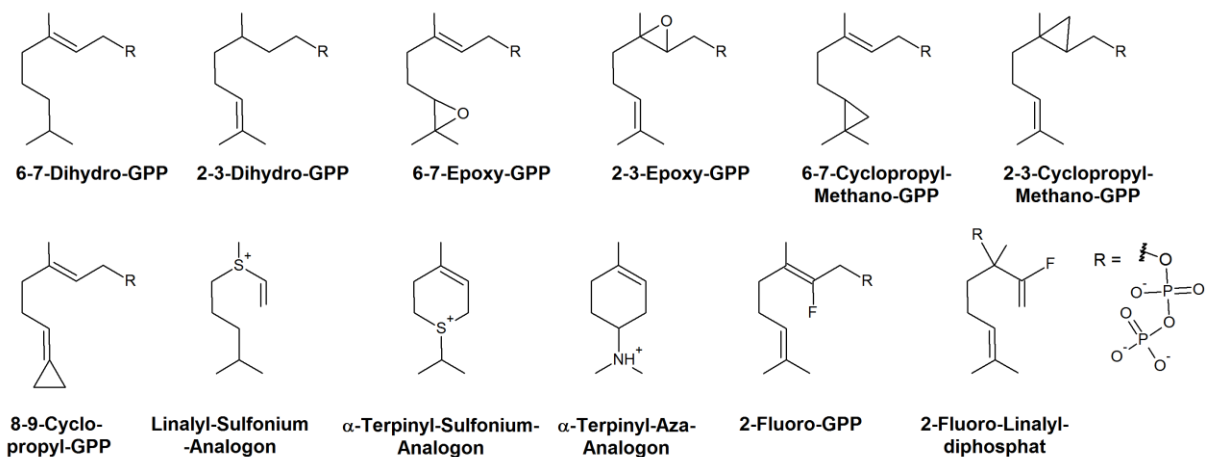


Abbildung 7: Inhibitoren diverser Monoterpensynthasen, Abbildung nachvollzogen aus⁸³

Pflanzliche Diterpensynthasen wie *ent*-Kaurensynthase, die bereits zyklisierte Prenyldiphosphate umsetzen, können mechanismusbasiert inhibiert werden. Dies zeigten für *Arabidopsis thaliana* *ent*-Kaurensynthase (AtKS) Roy und Kollegen mit 16-Aza-*ent*-Beyeran und 16-Aza-*ent*-Trachyloban.¹⁹²

Von pharmazeutischem Interesse ist die Inhibierung bestimmter prenylierender Enzyme, die in Knochenerkrankungen wie Knochenkrebs und Osteoporose eine Rolle spielen.¹⁹³ Eine Reihe stickstoffsubstituierter Bisphosphonate (siehe Abbildung 8) inhibiert Farnesyldiphosphatsynthase. Durch ihre substratanaloge Struktur binden sie mit hoher Affinität an das Enzym, können aber nicht durch das Enzym ionisiert werden. Das verhindert die Bildung von FPP als Substrat für spezielle Proteinprenylierungen, die für Signaltransduktionen in Knochenkrankheitsbildern mitverantwortlich sind (Russell,¹⁷²):

- nach Behandlung mit Zoledronat ist das Risiko neuer Knochenfrakturen geringer¹⁹⁴
- Alendronat beschleunigt die Heilung von Knochenbrüchen in Mäusen¹⁹⁵

- Pamidronat wirkt gegen verschiedene Knochenkrebsarten *in vitro*^{196,197}
- Minodronat hemmt in Kombination mit anderen Wirkstoffen Knochenkrebs¹⁹⁸

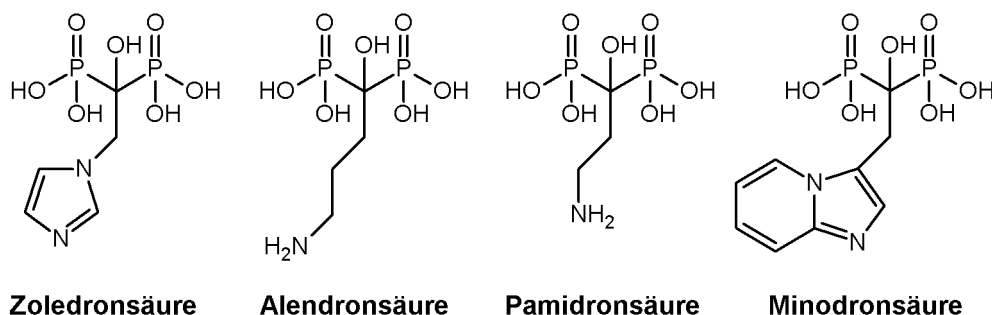


Abbildung 8: Pharmazeutisch relevante Bisphosphonate, die FPP-Synthasen inhibieren.¹⁷²

1.1.6 Theoretische Untersuchungen

Theoretische Modellierungen bedienen sich unterschiedlicher Methoden, um hypothetische Reaktionsmechanismen zu untermauern bzw. zu widerlegen und können *in vivo*- und *in vitro*-Experimente für weiteres Verständnis leiten. Einerseits dienen bioinformatische Methoden dem Erkennen von Ähnlichkeiten zwischen Aminosäuresequenzen und der genomischen Organisation oder der vorläufigen Zuweisung von Funktionen an uncharakterisierte Genprodukte. Andererseits geben computerchemische Untersuchungen Einblick in molekulare Verhältnisse und Prozesse. Die formalen Reaktionsmechanismen zahlreicher Prenyltransferasen und Terpenzyklasen wurden mit der Identifikation ihrer Funktion postuliert.^{26,44,51,87,199,200}

Der Übersichtsartikel von Degenhardt *et al.* fasst den Wissensstand zu den Reaktionsmechanismen vieler Mono- und Sesquiterpensynthasen zusammen, die überwiegend biochemisch charakterisiert wurden.⁶ Um die theoretische Untersuchung der Reaktionsmechanismen solcher Enzyme (ohne expliziten Einbezug der Enzyme) mit quantenchemischen Methoden haben sich in den letzten Jahren besonders Tantillo und Kollegen verdient gemacht:

- Zyklisierungswege des Farnesylkations zu Pentalenen²⁰¹
- Studie zur intramolekularen Mobilität eines Protons in der von Taxadiensynthase katalysierten Reaktion²⁰²
- Diskussion von Bifurkationen der Energiehyperfläche in Reaktionsmechanismen am Beispiel der Abietadiensynthasereaktion²⁰³
- Umfangreiche Berechnungen zur Theorie, dass konkrete Konformationen des Bisabolykations in Sesquiterpensynthasen die Produktbildung maßgeblich bestimmen²⁰⁴
- Übersichtsartikel zur intramolekularen Stabilisierung von kationischen Zwischenzuständen in der Terpenbiosynthese²⁰⁵
- Theoretische Ableitung formaler Reaktionsmechanismen für diverse Diterpene, welche die Bildung sekundärer Karbokationen vermeiden²⁰⁶
- Berechnungen zu feinen Unterschieden bei kationischen Zwischenzuständen in der Bildung von Bergamotenen und verwandten Terpenen²⁰⁷

- Berechnung der kompletten karbokationischen Reaktionskaskade von GGPP zum Taxadien²⁰⁸

Folgender Umstand macht quantenchemische Berechnungen zu formalen Reaktionsmechanismen (das heißt ohne Einbezug des Enzyms) interessant: Teilweise kann abgeleitet werden, welche Aspekte des Reaktionsverlaufs das katalysierende Enzym adressieren muss, um das entsprechende Produkt zu bilden, wie die Bereitstellung eines Intermediates in einer gewissen Konformation oder die Stabilisierung eines bestimmten Zwischenzustandes. Ebenfalls von Tantillo stammt ein Übersichtsartikel zu theoretischen Untersuchungen der Rolle von Karbokationen in der Terpenbiosynthese.²⁰⁹

Auch kombinierte Quanten- und Molekülmechanikansätze (QM/MM) wurden bereits für die Simulation von Terpensynthesereaktionen benutzt.⁸⁹ Zhou *et al.* adressierten mit Hilfe von Moleküldynamiksimulationen mit verschiedenen Kraftfeldern die Rolle einer Loop-Substruktur in der Klasse-II-Domäne der *Abies grandis* Abietadiensynthase.⁴⁶ In einer Publikation, in der Experimente mit radiomarkiertem Substrat und quantenmechanische Berechnungen kombiniert wurden, werden Reaktionsmechanismen für die Bildung der Diterpene Phomactatrien und Taxadien vorgeschlagen.⁸⁷ Die Funktionsweise von cisoiden und transoiden Zyklisierungsmechanismen in Sesquiterpenen wurde ebenfalls mit quantenchemischen Methoden und experimentell mit Substratanaloga untersucht.⁴⁵ Ferner existieren auch allgemeine Betrachtungen zu den Voraussetzungen für eine aktivierende Bindung von Diphosphaten in Substratbindetaschen von Enzymen.²¹⁰

Zu diesen computerchemischen Ansätzen kommen zahlreiche bioinformatische Methoden, die vor allem Einblicke in die Phylogenetik liefern bzw. für den Vergleich wichtiger Sequenzmotive verwendet werden – siehe hierzu Phylogenie, Seite 9 f. und ^{27,52,211}. Im Bereich der Prenyltransferasen gibt es weniger Diversität bezüglich der molekularen Abläufe, so dass häufig Homologieschlüsse geführt werden. Nur wenige computerchemische Untersuchungen befassten sich bisher mit Prenyltransferasen, so zum Beispiel die Modellierungen membrangebundener Enzyme, die *p*-Hydroxybenzoesäure prenylieren.^{63,64,212,213} Durch die wachsende Zahl verfügbarer Gensequenzen und experimenteller Strukturen wächst auch das Wissen über Phylogenetik und Struktureigenschaften von Proteinen im Allgemeinen. Solche theoretischen Methoden dienen heutzutage im gesamten naturwissenschaftlichen Bereich als wichtige Ergänzungen zur empirischen Wissenschaft.

1.1.7 Beispiele für prenylierende Enzyme

Um einerseits auf die in dieser Arbeit untersuchten Enzyme hinzuweisen und andererseits die Vielfalt der prenylierenden Enzyme zu demonstrieren, beinhaltet der folgende Abschnitt kurze Charakterisierungen ausgewählter Prenyltransferasen und Terpensynthasen.

1.1.7.1 Limonensynthase aus *Mentha spicata*

Eine gut untersuchte Monoterpenzyklase ist die (4S)-Limonensynthase aus der Grünen Minze (*Mentha spicata*). Die Aufklärung von Strukturen dieses homodimeren, löslichen Enzyms im Komplex mit Substratanaloga bzw. Intermediatanaloga erlaubt eine detaillierte Beschreibung des enzymatischen Reaktionsprozesses.¹² Die Enzymkatalyse folgt dem Klasse-I-Mechanismus. Hyatt *et al.* diskutieren leicht unterschiedliche Mechanismen für die Umwandlung intermediärer Linalyldiphosphat-Konformationen, je nachdem, ob ein Linalyldiphosphat-Molekül aus dem Lösungsmittel stammt oder in der Substratbindetasche entstand.¹² Die diskutierten Unterschiede werden in Abbildung 9 dargestellt.

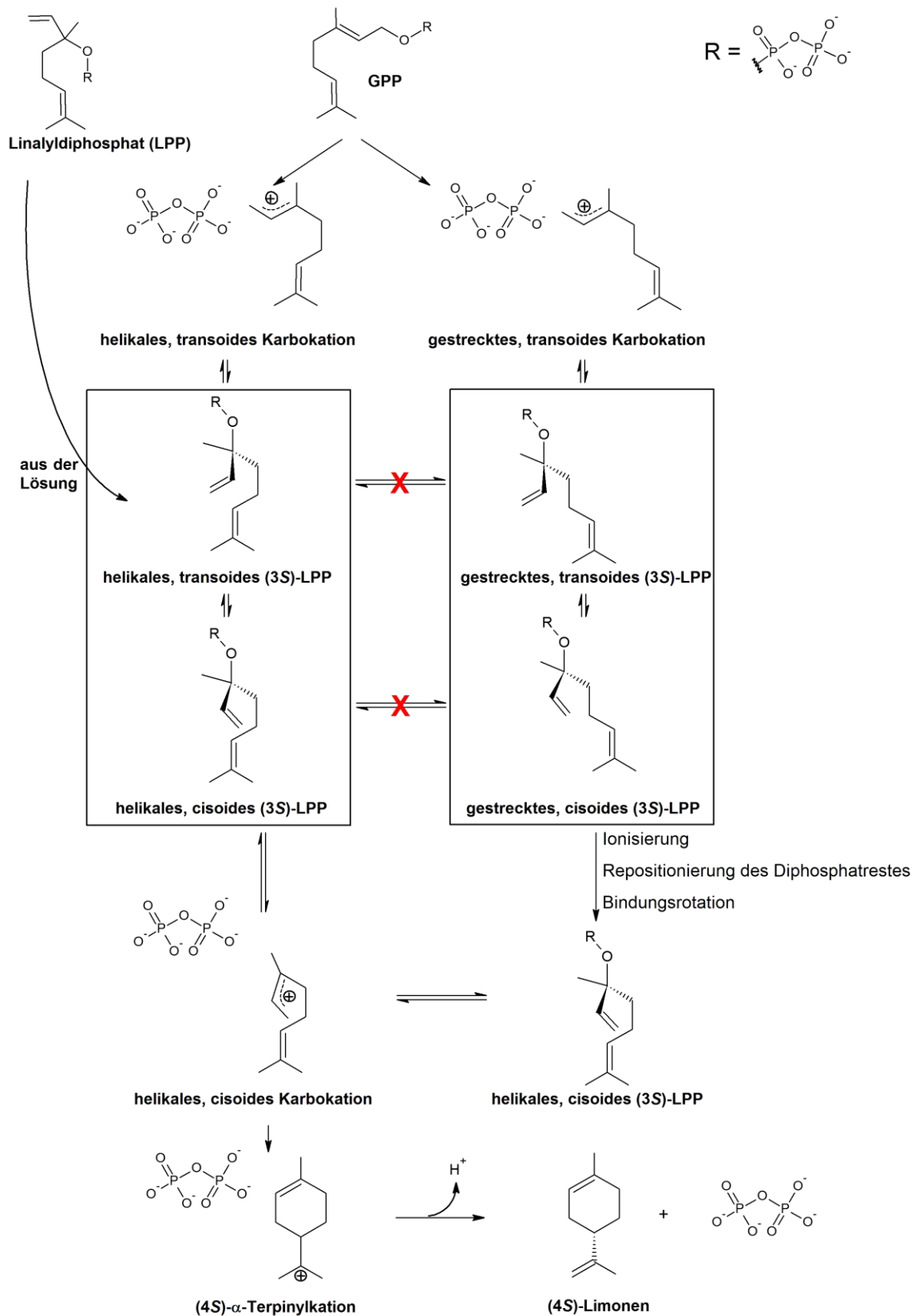


Abbildung 9: Vorgeschlagene Reaktionsmechanismen in *Mentha spicata* Limonensynthese, nach Hyatt *et al.*,¹². Links ist die traditionelle Vorstellung von der gekoppelten Isomerisierung-Zyklisierung dargestellt, rechts zusätzliche Konformationsumwandlungen. Die in Kästen dargestellten Zwischenzustände sind anhand der Röntgenkristallstrukturmodelle plausibilisiert.

Die detaillierte Darstellung von Hyatt *et al.* verdeutlicht, dass in Terpensynthesen geringfügige Konformationsunterschiede wesentlichen Einfluss auf den Reaktionsverlauf haben können.

1.1.7.2 Isopentenyltransferasen aus *Arabidopsis thaliana*

In *A. thaliana* gibt es neun Gene, welche als putative Isopentenyltransferasen annotiert sind.^{214,215} Sie kodieren für die Proteine AtIPT1 bis AtIPT9. Diese sind eingeteilt in zwei Gruppen, welche sich strukturell und funktionell unterscheiden:

AtIPT2 und AtIPT9 sind DMAPP:tRNA-Transferasen (EC-Klasse 2.5.1.8).

AtIPT1 und AtIPT3 bis AtIPT8 sind DMAPP:ATP/ADP-Transferasen (EC-Klasse 2.5.1.27).

Beide Enzymunterklassen spielen wichtige Rollen in der Cytokininbiosynthese. Cytokinine sind Phytohormone von zentraler Bedeutung für das Wachstum und die Entwicklung von Pflanzen.²¹⁶ Prozesse, die durch Cytokinine beeinflusst werden, sind u. a. Zellteilung, Photosynthese und Zelldifferenzierung.²¹⁷ Strukturell sind natürliche Cytokinine Adeninderivate, die an N⁶-Position entweder durch Prenylierung oder aromatisch modifiziert sind. Die Positionen N³, N⁷ und N⁹ werden in Pflanzen oft glykosyliert, womit die Bindung an Cytokininrezeptoren unterbunden wird.^{218,219} Die resultierenden Nukleoside gelten als Speicherform für Cytokinine während die freien Basen als aktive Form der Cytokinine betrachtet werden.²¹⁹

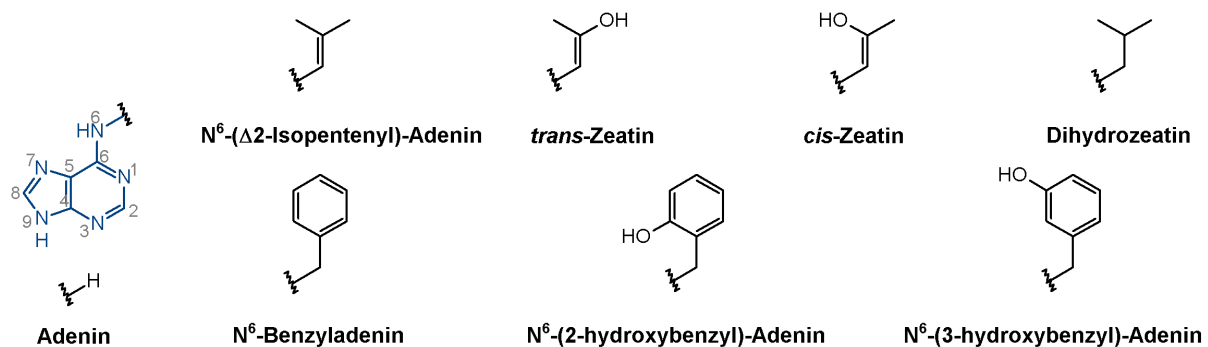
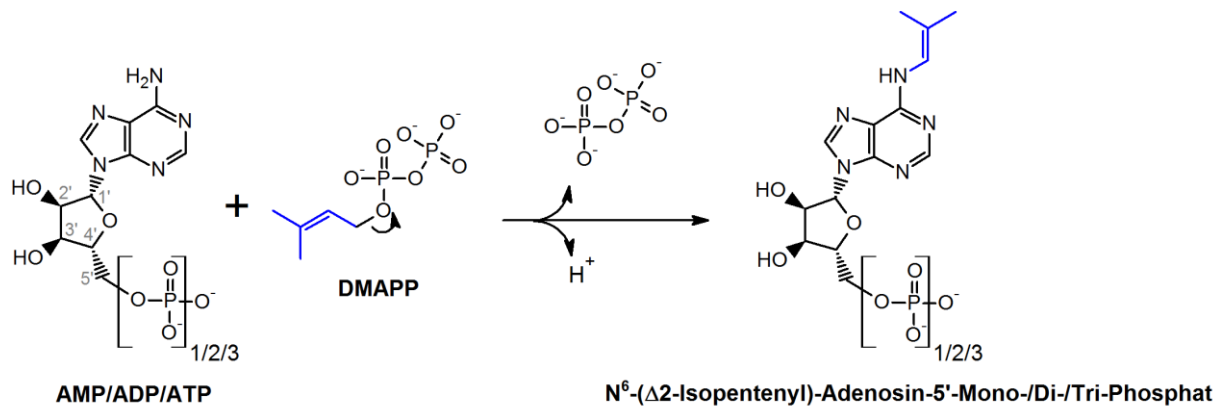


Abbildung 10: Strukturformeln von Cytokininen als freie Basen des Adeningrundkörpers mit Substitutionen, Abbildung nach ²¹⁸

Die jeweilige Modifikation an N⁶ hat wesentlichen Einfluss auf die physiologische Wirkung. Bereits kleine Unterschiede in der Substitution resultieren in drastischen Veränderungen der Hormonantwort.²¹⁸ Isopentenyltransferasen bilden hierbei den Ausgangspunkt für die Biosynthese prenylierter Cytokinine. Sie übertragen den Prenylrest auf das N⁶-Atom von Adenosinresten, wie in Abbildung 11 veranschaulicht. Verschiedene Enzyme katalysieren die in Abbildung 10 gezeigten Modifikationen an N⁶-(Δ 2-Isopentenyl)-Adenin zu Zeatinen.^{218,220} Isopentenyl:tRNA-Transferasen übertragen die Prenylgruppe auf Adenosine nahe der Anticodonregion von tRNA-Molekülen, woraus über enzymatische Folgeschritte Cytokinine regeneriert werden.²¹⁴ Aus diesem Biosyntheseweg stammt jedoch nur ein geringer Teil des physiologischen Cytokininhaushalts von Pflanzen.²¹⁵ Größere Bedeutung kommt der Prenylierung von Adenosinphosphaten zu.²²¹ Takei *et al.* identifizierten 2001 erstmals die Gene für AtIPT1-8 mit der Spekulation, dass AtIPT2 zur Klasse der Isopentenyl:tRNA-Transferasen gehört und führten erste Funktionsanalysen nach heterologer Expression in *Escherichia coli* an AtIPT1, 3 und 8 durch.²¹⁴ Im gleichen Jahr berichtete Kakimoto über das neunte AtIPT-Gen. Er zeigte eine Bevorzugung von AtIPT4 für ATP und ADP gegenüber AMP als Prenylakzeptor.²¹⁵ Eine tRNA-modifizierende Isopentenyltransferase (IPT) aus *A. thaliana* wurde 2002 charakterisiert.²²²


Abbildung 11: IPT-katalysierte Reaktionen

Bakterielle Isopentenyltransferasen nutzen AMP als Prenylakzeptor und als Donormoleküle sowohl DMAPP als auch 1-Hydroxy-2-methyl-2-(*E*)-butenyl-4-diphosphat (HMBPP). Damit kürzen sie den herkömmlichen Cytokininbiosyntheseweg um die Involvierung einer der Prenylierung folgenden Hydroxylierung ab.^{55,223} Tabelle 3 fasst Ergebnisse von Untersuchungen der Prenylakzeptorspezifität einiger pflanzlicher IPT zusammen. Diese Resultate zeigen, dass verschiedene IPT einige Nukleotide bevorzugen.

Tabelle 3: Michaeliskonstanten für ausgewählte Substrate verschiedener IPT. Tmr und Tzs sind zwei IPT-Enzyme aus *A. tumefaciens*. n. b.: nicht bestimmt

Enzym	DMAPP	AMP	ADP	ATP
Tmr ²²⁴	8,5 μM	85 nM	n. b.	n. b.
AtIPT1 ²¹⁴	~50 μM	~185 μM	14,6 μM	11,4 μM
AtIPT4 ²¹⁵	11,6 μM	Kein Umsatz	9,1 μM	3,4 μM
Tzs ²²⁵	11,1 μM	8,2 mM	n. b.	n. b.

Für AtIPT4 wurde außerdem eine Abhängigkeit der Enzymaktivität von Magnesiumionen als Kofaktor nachgewiesen – ohne Magnesium zeigte sich eine deutlich verringerte Enzymaktivität.²¹⁵ Untersuchungen zur räumlich differenzierten Expression der AtIPT *in vivo* untermauern, dass die verschiedenen Isoenzyme zur Modulation gewebespezifischer Cytokininbedürfnisse notwendig sind.^{226,227} Das AtIPT6-Gen enthält in einigen Ökotypen eine Verschiebung des Leserahmens und wird nicht exprimiert.²¹⁹

Anhand einer Röntgenkristallstrukturanalyse der DMAPP:tRNA-Transferase aus *Saccharomyces cerevisiae* wurde 2008 ein detailliertes Modell für den Reaktionsmechanismus dieser Enzymklasse entworfen.⁵⁴ Für die *A. tumefaciens* IPT Tzs existiert seit 2008 ein röntgenkristallographiebasiertes Strukturmodell, das ebenfalls Rückschlüsse auf den Reaktionsmechanismus bietet.⁶⁶ Das erste Modell einer pflanzlichen DMAPP:ATP/ADP-Transferase wurde 2010 veröffentlicht und stammt von Röntgenkristallstrukturanalysen des Enzyms aus *Humulus lupulus* (Hopfen).⁵⁵

Interessanterweise sind IPT selbst Ziel von Prenylierungen. So konnte nachgewiesen werden, dass eine Farnesylierung von AtIPT3 die Lokalisation des Enzyms von Zellkern und Zytoplasma in die Plastiden verschiebt und somit die Cytokininbiosynthese moduliert.²²⁸

1.1.7.3 Kaurensynthese-ähnliche Enzyme

Die Gruppe der Kaurensynthese-ähnlichen Enzyme ist über Sequenzähnlichkeit zum Enzym Kauren-Synthase definiert. Der Begriff *kaurene synthase-like* (engl., Kaurensynthese-ähnlich, nachfolgend KSL) wurde zum ersten Mal im Jahr 2004 bei Untersuchungen am Genom von *Oryza sativa* verwendet. Sieben Gensequenzen wurden entsprechend als Kaurensynthese-ähnlich annotiert.²⁰⁰ Diese Annotation beruht auf einem Gibberellinmangel-Phänotyp, der unter Defizienz eines bestimmten Gens beobachtet wurde.²²⁹ Kaurensynthase ist ein wichtiges Enzym für die Bereitstellung von Gibberellin *in vivo*, so dass für das entsprechende Gen die Kaurensynthasefunktion vorhergesagt wurde.²²⁹ Nachfolgende Sequenzvergleiche zeigten, dass viele Pflanzen über homologe Kaurensynthasegene verfügen. Experimente zur Funktion der Genprodukte belegen unterschiedliche Diterpensynthaseaktivitäten.^{230,231} Eine funktionelle Charakterisierung der als Kaurensynthase annotierten Gene in Reis wurde 2007 von Xu *et al.* durchgeführt und beschreibt die Bildung der Diterpene in Abbildung 12.²³²

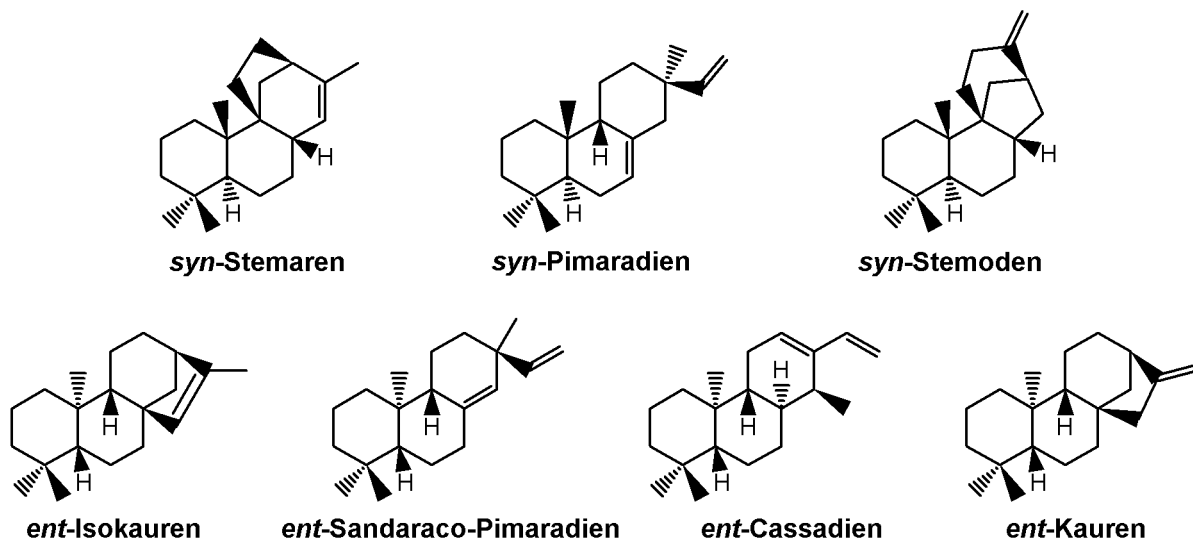


Abbildung 12: Produkte der Kaurensynthese-ähnlichen Enzyme aus Reis (*Oryza sativa* subsp. *indica*) nach ²³²

Die KSL-Enzyme aus Reis wurden umfänglich untersucht und bieten einen Einblick in die Produktvielfalt dieser Gruppe von pflanzlichen Enzymen, die als Klasse-I-Terpensynthasen eingestuft werden. Die Reaktionsprodukte der Reis-KSL-Enzyme sind Vorstufen für die Biosynthese diverser in Pflanzen wirksamer Stoffe wie Momilaktone, Phytoalexine und Gibberelline.²³² Gibberelline sind eine Klasse von Phytohormonen, die zahlreiche metabolische Prozesse in Pflanzen modulieren, beispielsweise Keimung, Streckungswachstum, Blütenbildung und Samenentwicklung.^{233–235} Der Pilz *Gibberella fujikuroi* nutzt Gibberellin zur Interferenz mit dem Wachstum von Wirtspflanzen und ist gleichzeitig Namensgeber der Substanzklasse.^{99,236} In *Arabidopsis thaliana* wurde eine enge Verknüpfung zwischen Gibberellinen und dem zirkadianen Rhythmus festgestellt.^{237,238}

Eine Publikation aus dem Jahr 2011 beschreibt die Synthese von *ent*-Kauren-Derivaten, die an Maus-Makrophagenzellen antiinflammatorisch wirkten.²³⁹ Eine mögliche Weiterentwicklung zur Behandlung von Entzündungen wird angeregt. Eine andere Studie zeigte, dass in Linien von *Streptomyces platensis* Kaurensynthasen und verwandte Enzyme eine tragende Rolle in der Biosynthese von Platensimycin und Platencin spielen – zwei vielversprechenden Leitstrukturen in der Diabetestherapie.²⁴⁰

Ein Übersichtsartikel von Hedden und Kamiya beschreibt wichtige Schritte der Biosynthese von Gibberellinen und einige Aspekte ihrer Regulation.²³³ Gibberelline sind modifizierte Diterpene und leiten sich als solche von Geranylgeranyldiphosphat ab.²³³ Für die Biosynthese von *ent*-Kauren sind in höheren Pflanzen die Enzyme *ent*-Copolyldiphosphat-synthase und *ent*-Kaurensynthase verantwortlich (Schema in Abbildung 13) während die fungale Biosynthese ein bifunktionales Enzym verwendet.^{99,231} Auch in dem Moos *Physcomitrella patens* wurde ein multifunktionales Enzym nachgewiesen, das *in vitro* zusätzlich die Umsetzung von *ent*-Copolyldiphosphat zu *ent*-16 α -Hydroxykauren katalysiert.²³¹ Die singulären Enzymfunktionen wurden beispielsweise bei *A. thaliana* und *Cucurbita maxima* gefunden und verteilen sich auf jeweils zwei Enzyme, die in den Plastiden lokalisiert sind.^{241–243} Der weitere Metabolismus von Gibberellinen beinhaltet diverse Modifikationen an unterschiedlichen Positionen, die von Hedden und Kamiya übersichtlich dargestellt werden.²³³

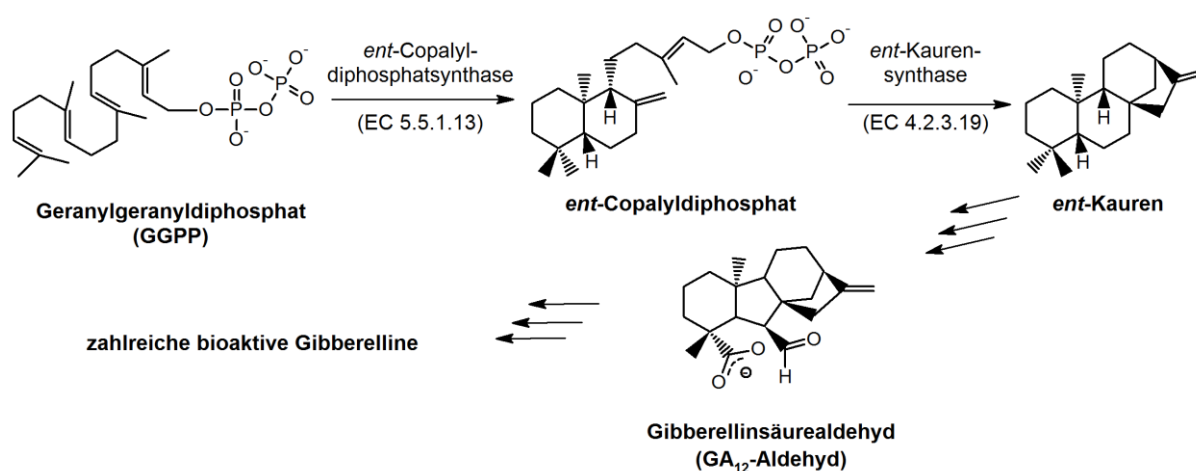


Abbildung 13: Zyklisierung von GGPP und Folgeschritte in der Biosynthese diverser Gibberelline. Drei Reaktionspfeile bedeuten jeweils mehrere Modifikationsschritte. Für eine Übersicht zu produzierten Gibberellinen siehe Hedden und Kamiya,²³³.

Die ExPASy-ENZYME-Datenbank (²⁴⁴) verzeichnet UniProt-Einträge (²⁴⁵) für fünf verschiedene *ent*-Kaurensynthasen (URL: <http://enzyme.expasy.org/EC/4.2.3.19> - Recherche vom 01.03.2012) mit den folgenden Identifikatoren und Herkunftsorganismen:

- Q9UVY5: *Gibberella fujikuroi* – das bereits erwähnte bifunktionale Enzym, das in der Lage ist, GGPP in zwei Schritten zu *ent*-Kauren umzusetzen (Klasse-I/II-Terpensynthase: Bifunktionalität)⁹⁹
- O13284: *Phaeosphaeria* sp. (strain L487) – ebenfalls bifunktional^{246,247}
- Q0JA82: *Oryza sativa* subsp. *japonica* – auch OsKSL1 genannt, katalysiert die Reaktion von *ent*-Copolyldiphosphat zu *ent*-Kauren (Klasse-I-Terpensynthase)²³²
- Q39548: *Cucurbita maxima* – katalysiert die Reaktion von *ent*-Copolyldiphosphat zu *ent*-Kauren (Klasse-I-Terpensynthase)²⁴¹
- Q9SAK2: *Arabidopsis thaliana* – katalysiert die Reaktion von *ent*-Copolyldiphosphat zu *ent*-Kauren (Klasse-I-Terpensynthase)

Alle entsprechenden Proteine enthalten ein Metallionenbindemotiv, die beiden fungalen Proteine zeigen die Ausprägung DEXXE während die pflanzlichen Proteine DDXXD aufweisen.

Die folgenden Beispiele sollen verdeutlichen, wie geringe Unterschiede in den Substratbindetaschen der Enzyme die Produktbildung beeinflussen: Anhand von Vergleichen der Primärstrukturen und Produktspektren verschiedener KSL identifizierten Wilderman *et al.* eine einzelne Aminosäure in der Substratbindetasche eines Reis-KSL-Enzyms, deren Austausch die Produktbildung moduliert.²⁴⁸ Dabei korrespondiert ein Wechsel eines Threonins zu Isoleucin mit einer Änderung des enzymatisch gebildeten Produktes von *ent*-Kauren bzw. *ent*-Atiseren zu *ent*-Pimaradien.²⁴⁸ Anschließend zielgerichtete Mutagenesen in der Abietadiensynthese aus *Abies grandis* konnten einen ähnlichen Effekt an einer Aminosäure in der Nachbarschaft der entsprechenden beschriebenen Mutationsstelle hervorbringen: Hier führt die Substitution eines Alanin durch Serin auch zur Produktion von Pimaradien statt Abietadienen.²⁴⁹

Sallaud *et al.* untersuchten 2009 zwei Kandidatengene für die Biosynthese von Sesquiterpenen in *Solanum habrochaites*.¹⁰¹ Die Auswahl der Gene basiert auf Genkartierungen und festgestellten Unterschieden zur Kulturtomate *Solanum lycopersicum*. Während die glandulären Trichome („Drüsenhaare“) der Wildtomate *S. habrochaites* eine Reihe von Phytochemikalien produzieren, die zum Teil toxische Wirkung auf Insekten haben, ist der Kulturtomate die Fähigkeit zur Bildung dieser Stoffe verloren gegangen.^{101,250–252} Introgressionslinien zwischen beiden Tomatenarten erlauben die funktionelle Charakterisierung der für diesen Umstand verantwortlichen Gene. Mit Hilfe solcher Introgressionslinien konnten zwei DNA-Abschnitte identifiziert werden, die mit der Bildung verschiedener Klassen von Sesquiterpenen in Verbindung gebracht wurden: *Sst1* für die Bildung von Germacren B und *Sst2* für α -Santalene, α - und β -Bergamotene (und jeweils eine Reihe nicht identifizierter Sesquiterpene).²⁵³ Sallaud und Kollegen charakterisierten die zwei Gene des *Sst2*-Lokus, die für die Bildung der entsprechenden Sesquiterpene verantwortlich sind und machten interessante Entdeckungen: Eines der Gene kodiert für eine (*Z,Z*)-Farnesyldiphosphatsynthase, das andere Gen für eine Santalene- und Bergamotensynthase. Transgene *Nicotiana glauca*-Pflanzen, die beide Gene tragen, produzieren das komplette Spektrum der in *S. habrochaites* mit dem *Sst2*-Lokus verknüpften Sesquiterpene. Die gefundene *Z*-Isoprenyldiphosphatsynthase katalysiert *in vitro* den Aufbau von (*Z,Z*)-FPP sowohl aus IPP und DMAPP als auch aus Neryldiphosphat (NPP) und DMAPP. Die identifizierte *S. habrochaites* Terpensynthase ist insofern besonders, als dass sie eine im Plastiden lokalisierte Sesquiterpensynthase ist und (*Z,Z*)-FPP umsetzt. Bisher bekannte Sesquiterpensynthasen finden sich im Cytosol und katalysieren Ionisierung und Zyklisierung von (*E,E*)-FPP. Sequenzanalysen ordnen die *S. habrochaites* Santalene- und Bergamotensynthase (ShSBS) als KSL-Enzym ein.

KSL-Enzyme aus *Santalum* spp. katalysieren ebenfalls die Bildung von Santalenen und Bergamotenen.²⁵⁴ Santalene sind metabolische Vorgänger der Komponenten des Sandelholz-Duftes und werden bei der Parfümherstellung verwendet. Gleiches gilt für *cis*-Abienol, das auch von KSL-Enzymen gebildet wird.²⁵⁵ Im Arbeitskreis von Prof. Alain Tissier am Leibniz-Institut für Pflanzenbiochemie in Halle (Saale) wird sehr aktiv an der Biosynthese dieser Terpenoide in glandulären Trichomen von Tabak- und Tomatenpflanzen geforscht.

1.2 Molekulares Docking

Um enzymatische Vorgänge auf molekularer Ebene zu verstehen, ist Wissen über die Strukturen der beteiligten Moleküle unabdingbar. Verschiedene Methoden der Molekülmodellierung am Computer kommen zum Einsatz, wenn experimentelle Verfahren qualitativ

unzureichende Ergebnisse liefern, zu aufwändig oder schlicht nicht anwendbar sind. Einen umfangreichen und aktuellen Überblick der Methoden im Bereich der Molekülmodellierung bietet die Spezialausgabe zum 25-jährigen Jubiläum der Zeitschrift *Journal of Computer-Aided Molecular Design* („*Special Issue: The next 25 years: Commemorating the 25th anniversary of the Journal of Computer-Aided Molecular Design*“ – Januar 2012, zum Beispiel ^{256–268}). Sie zeigt auf, dass die fortlaufende Entwicklung der Rechenmaschinen mit der Zeit die Simulation größerer molekularer Systeme oder die Anwendung aufwändigerer Verfahren erlaubt. *Molecular Modelling* (engl. Molekülmodellierung) und Computerchemie bringen oft direkt neues Verständnis oder leiten zu neuen empirischen Experimenten. Die Modellierung molekularer Prozesse ist somit ein wichtiges Methodenfeld für Veranschaulichung und Erkenntnisgewinn im gesamten Spektrum der Naturwissenschaften, sowohl im akademischen als auch im industriellen Bereich. Im Folgenden soll molekulares *Docking* ohne den Anspruch auf einen vollständigen Überblick erläutert werden.

1.2.1 Definition

Molekulares *Docking* ist eine Computermethode zur Vorhersage von Bindemodi eines Ligandmoleküls in oder an einem Rezeptormolekül.

Im Rahmen dieser sehr allgemeinen Definition sind verschiedene Blickpunkte zu adressieren, welche in diversen Computerprogrammen unterschiedlich umgesetzt sind:

- Ligand- und Rezeptormolekül: Konzeptionell wird mit dem Ligandenbegriff ein kleines Molekül verbunden, während ein Rezeptormolekül meist groß genug ist, um den Liganden teilweise oder vollständig zu umfassen. Für die Modellierung ist die Rolle beider Moleküle als Rezeptor bzw. Ligand nicht erheblich, da es in jedem Fall darum geht, die intermolekularen Wechselwirkungen beider zu beschreiben. Insbesondere beim *Docking* zweier Proteine aneinander wird nicht zwischen Ligand und Rezeptor unterschieden. Der Begriff des Liganden wird im Bereich des molekularen *Docking* verwendet, ohne dass die Bindeeigenschaft des entsprechenden Moleküls am Rezeptor gegeben ist.
- Bindemodus: Geometrische Anordnungen von Ligand und Rezeptor mit summarisch attraktiven intermolekularen Wechselwirkungen werden allgemein als Bindemodi bezeichnet. Im engeren Sinne sind Bindemodi die geometrischen Anordnungen, welche für die Interaktion zwischen Rezeptor und Ligand energetisch bevorzugt sind und sich trotzdem durch Unterschiede in den Wechselwirkungen voneinander abgrenzen. Es werden experimentell nachgewiesene und modellierte Bindemodi differenziert. Im Rahmen des molekularen *Docking* werden vorgeschlagene, also modellierte, Bindemodi behandelt.
- *Docking*-Pose: Kontrastierend zum Bindemodus ist der Begriff der *Docking*-Pose (im Folgenden auch einfach „Pose“) zu erwähnen, welcher eine konkrete, vom *Docking*-Algorithmus erzeugte Rezeptor-Ligand-Geometrie bezeichnet.

Molekulares *Docking* wird in der Regel auf Proteine als Rezeptoren mit kleinen organischen Molekülen als Liganden angewandt. Im Bereich der supramolekularen Chemie ist molekulares *Docking* zwar möglich, wird aber wenig benutzt. Die Ursache hierfür ist in der meistens geringen Größe und oft relativ einfach experimentell aufklärbaren Struktur supramolekularer Systeme zu sehen. Einen aktuellen Überblick über Molekulares *Docking* bieten Meng *et al.* (²⁶⁹, siehe auch ^{270,271}).

1.2.2 Das *Docking*-Problem

Als *Docking*-Problem wird die Herausforderung bezeichnet, zwischen Ligand- und Rezeptormolekül die Geometrie mit maximaler Attraktion beider Moleküle zu finden. *Docking*-Programme modellieren die Wechselwirkungen zwischen Ligand und Rezeptor. Es gibt aber auch Programme, die zusätzlich versuchen, Solvatationseffekte und Entropiebeiträge einzubeziehen und Rezeptor-Ligand-Affinitäten zu berechnen.^{272–278}

Mathematisch kann das *Docking*-Problem als ein multidimensionales Optimierungsproblem einer Bewertungsfunktion über dem Raum der Atomkoordinaten im Rahmen der kovalenten Bindungen der Moleküle gesehen werden. Die Dimension des Optimierungsproblems ist abhängig von den Freiheitsgraden der betrachteten Moleküle. Darunter befinden sich bei konstanter absoluter Position einer der beiden Strukturen mindestens die sechs Freiheitsgrade, die andere Struktur zu rotieren oder zu verschieben, jeweils in x-, y- und z-Richtung. Hinzu kommen je nach Modellierungsmethode die intrinsischen Freiheitsgrade des Liganden und des Rezeptors.²⁶⁹ Das *Docking*-Problem kann als formale Grundlage für alle Lösungsalgorithmen in *Docking*-Programmen verstanden werden.

1.2.2.1 Komplexität des *Docking*-Problems

Die Komplexität des *Docking*-Problems hängt in der Praxis nicht nur von den interagierenden Molekülstrukturen, sondern auch von der verwendeten Modellierung ab. Zeitgemäße *Docking*-Programme modellieren den Liganden flexibel, d.h. zum Einpassen in die Rezeptorstruktur mögliche Ligandkonformationen werden unter Einbezug der Rezeptorstruktur erstellt. Zunehmend betrachten *Docking*-Programme auch die Flexibilität des Rezeptors, wobei im Fall von Proteinen zwischen Algorithmen unterschieden werden kann, in welchen nur einzelne Aminosäureseitenketten flexibel sind (siehe beispielsweise²⁷⁹) und solchen, wo auch die Beweglichkeit des Peptid-Rückgrates einbezogen wird.²⁸⁰

Der Begriff des Freiheitsgrades ist in vielen *Docking*-Programmen gleichbedeutend mit einer zusätzlichen Dimension des Suchraums zu den bereits erwähnten Translations- und Rotationsfreiheitsgraden der beiden Molekülstrukturen zueinander. Durch die kovalente Verknüpfung der Ligand- und Rezeptoratome sind der Positionierung der Atome untereinander physikalische Grenzen gesetzt. Um eine hohe Zahl von Rezeptor-Ligand-Anordnungen rechnerisch sondieren zu können, werden vereinfachende Modelle benutzt, welche die Freiheitsgrade der relativen Atompositionierungen einschränken. So werden zum Beispiel in vielen *Docking*-Programmen Bindungslängen und -winkel als konstant betrachtet.

Zahlreiche *Docking*-Programme betrachten frei rotierbare Bindungen als Freiheitsgrad, wie Abbildung 14 am Beispiel verdeutlicht. Hinzu können – je nach Modellierung – Freiheitsgrade aus der Betrachtung des Protonierungsgrades von Rezeptor- und Ligandstruktur kommen, welche aber als zusätzliche Rezeptoren bzw. Liganden modellierbar sind. Somit sind diese Freiheitsgrade auf analoge Probleme zurückführbar. Konjugierte π -Bindungen und somit eingeschränkte Drehbarkeit, auch in Peptidbindungen, werden unterschiedlich modelliert, oft in Abhängigkeit verwendeter molekülmechanischer Kraftfelder (einige Beispiele: ^{281–284}, Übersichtsartikel: Kitchen *et al.*,²⁸⁵, Meng *et al.*,²⁶⁹).

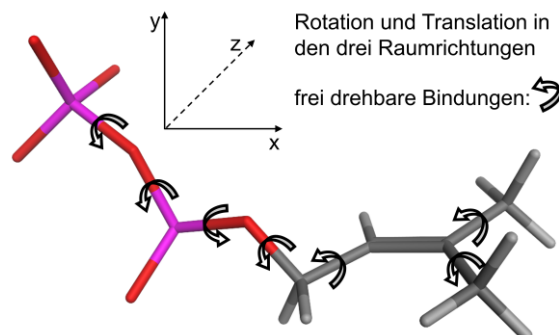


Abbildung 14: DMAPP als Beispiel für Freiheitsgrade eines Liganden bei molekularem Docking. Wasserstoff- und Rezeptoratome werden in vielen Docking-Programmen zusätzlich betrachtet. Die Abbildung wurde unter Benutzung von MOE erstellt.²⁸⁶

Sadjad und Zsoldos haben grundlegende Überlegungen zur Komplexität des Einpassens starrer Liganden in eine Rezeptorstruktur unternommen und zeigen, dass bereits dieses Problem NP-schwer ist.²⁸⁷ Praktische Ansätze zur Lösung des Docking-Problems sind in aller Regel zweigeteilt in die Erzeugung von Posen und deren anschließende Bewertung.

1.2.2.2 Posengenerierung – Sampling

Der zu durchsuchende Raum von Konformationen der beiden Moleküle und deren möglichen relativen räumlichen Lagen zueinander ist in den meisten Fällen hochdimensional.²⁸⁸ Der Sampling-Prozess eines Docking-Programms ist dafür verantwortlich Posen aus diesem Raum auszuwählen, zum Beispiel durch:

- Systematische Suche
- Finden eines lokalen Optimums einer Bewertungsfunktion (siehe Abschnitt 1.2.2.3, Posenbewertung – Scoring)
- Diversität gegenüber früher erzeugten Posen
- Kombinationen aus obigen Kriterien

Für die Generierung von Rezeptor-Ligand-Posen haben sich verschiedene Verfahren etabliert, die oft stochastische Ansätze beinhalten, um zu versuchen, ein repräsentatives Abbild des gesamten Posenraums zu erhalten, beispielsweise:

- Evolutionsinspiriert²⁸⁸
- Genetischer Algorithmus²⁸⁹
- Iterative stochastische Eliminierung²⁹⁰
- Monte Carlo-Verfahren²⁹¹
- Populationsbasiert²⁹²

Zsoldos *et al.* bieten eine Klassifikationsübersicht für Docking-Programme (²⁸¹, siehe auch Huang und Zou,²⁷⁰). Die Erzeugung einzelner Posen ist ein vergleichsweise schneller Schritt in Docking-Programmen, allerdings ist ein vollständiges Sampling unmöglich und auch nicht sinnvoll ohne eine Möglichkeit, Posen nach objektiven Kriterien aus der Lösungsmenge auszuschließen.

1.2.2.3 Posenbewertung – Scoring

Durch das Sampling generierte Docking-Posen werden durch eine Scoring-Funktion bewertet, um sie qualitativ miteinander vergleichen zu können. Da molekulares Docking meist auf Rezeptor-Ligand-Systeme mit zahlreichen (über 100) Atomen angewendet wird und auch für große Mengen an Molekülstrukturen, ist die Zeitdauer, welche für die

Auswertung einer Pose mit der *Scoring*-Funktion benötigt wird, ein maßgebliches Kriterium für deren praktische Anwendbarkeit. Es haben sich verschiedene Kategorien von *Scoring*-Funktionen entwickelt:

- Formorientiert ^{293–302}: ein wesentliches Element der *Scoring*-Funktion ist die räumliche Passform des Liganden zum Rezeptor.
- Gitterbasiert ^{303–305}: die *Scoring*-Funktion wird an vordefinierten Gitterpunkten ausgewertet, womit der Lösungsraum diskretisiert wird.
- Empirische Kraftfelder ^{279,306–311}: Wechselwirkungen werden durch Kraftfelder beschrieben, die aus quantenchemischen Berechnungen oder experimentellen Strukturen abgeleitet wurden.
- Basierend auf quantenmechanischen Berechnungen beteiligter Wechselwirkungen ^{312–319}

Um die Beschreibungsmacht der an sich limitierten Ansätze zu erhöhen, sind zahlreiche kombinierte Verfahren entwickelt worden.^{281,297,301,320–328} Huang *et al.* berichten über den aktuellen Stand der Wissenschaft zu *Scoring*-Funktionen im Protein-Ligand-*Docking*, welche sie in die Klassen empirisch, kraftfeld- und wissensbasiert einteilen.³²⁹

Ein Durchlauf eines *Docking*-Experiments besteht aus der Generierung einer Pose und der anschließenden Bewertung mittels der benutzten *Scoring*-Funktion. Die meisten *Docking*-Programme suchen in zahlreichen Iterationen weitere *Docking*-Posen und bieten dem Benutzer schließlich eine entsprechend der *Scoring*-Funktion sortierte Reihe an *Docking*-Posen. Als Besonderheit können fragmentbasierte *Docking*-Algorithmen gelten, die den Liganden zerlegen, für die Einzelteile gute *Docking*-Posen suchen und anschließend versuchen den Liganden unter Benutzung dieser Posen in der Bindetasche zu rekonstruieren.³³⁰

1.2.3 Historische Entwicklung

Aus historischer Sicht hat sich die Bedeutung des molekularen *Docking* im Laufe der Zeit gewandelt. Einerseits haben sich die Rechenmaschinen weiterentwickelt und andererseits, auch als Folge des technischen Fortschritts, entstanden neue Ansprüche an das molekulare *Docking*. Eine der ersten theoretischen Untersuchungen von Ligand-Rezeptor-Wechselwirkungen im Bereich der Biochemie beinhaltete im Jahr 1975 die Notwendigkeit umfangreicher visueller Inspektionen und manueller Interventionen.³³¹ Mit der Etablierung von Algorithmen, welche verschiedene Ligandenkonformationen docken und bewerten, wurde in den 1980er Jahren ein automatisches Sortieren der erzeugten *Docking*-Posen nach objektiven Kriterien eingeführt.^{296,302,332,333} Diese Kriterien umfassten zunehmend physikalisch-chemische Eigenschaften der beteiligten funktionellen Gruppen bzw. Atome, um mit *Scoring*-Funktionen die Interaktionen zwischen Atomen besser widerspiegeln zu können.^{297,305,320,324,334,335} Mit wachsender Rechnerleistung und -zugänglichkeit rückten weitere Aspekte in den Blickpunkt der Forschung (ausgewählte Meilensteine und Beispiele, grob chronologisch geordnet von Mitte der 1990er Jahre bis 2012):

- Genetische Algorithmen zur Konformationsauswahl für Ligandstrukturen ^{288,289}
- Vorhersage von Protein-Protein-Wechselwirkungen (Protein-Protein-*Docking*) ^{289,336}
- Benutzung von Kraftfeldoptimierungen für erzeugte *Docking*-Posen ³²¹
- Monte-Carlo-*Sampling* zur Konformationsauswahl für Ligandstrukturen ²⁹¹
- Modellierung von Solvatation für Ligand und Rezeptorprotein ^{337–339}

- Ableitung von Wechselwirkungspotenzialen aus experimentell bestimmten Rezeptor-Ligand-Komplexen^{309,340}
- Adressierung von Rezeptorflexibilität^{280,341}
- *Docking* in Protein-Homologiemodelle³⁴²
- „blindes“ *Docking*: *Docking* ohne Vorgabe der Bindestelle am Rezeptor^{343–346}
- Betrachtungen zu enthalpischen und entropischen Beiträgen zur freien Bindungsenergie eines Liganden am Rezeptor^{275–277,347–350}
- „Hierarchisches *Screening*“ (engl., durchkämmen – hier zum Beispiel von Molekülbibliotheken) zur Konformationsauswahl für Ligandstrukturen³⁵¹
- *Scoring*-Funktion für Bindeaffinität des Liganden am Rezeptor^{352,353}
- „Demokratisches“ *Docking* mit verschiedenen Programmen^{352,354}
- Kraftfeldgestützte Modellierung des kompletten Ligandenbindeprozesses am Rezeptor²⁷²

Diese zahlreichen Erweiterungen haben zum wissenschaftlichen Fortschritt im molekularen *Docking* beigetragen, allerdings ist es schwierig, *Docking*-Programme qualitativ zu vergleichen (Cole *et al.*,³⁵⁵), auch wenn regelmäßig versucht wird, die bessere Eignung eines Programms einem anderen gegenüber zu zeigen.^{326,356–361} Allein die Vielzahl von *Docking*-Programmen kann als Indikator für reges Forschungsinteresse einerseits, aber auch große wissenschaftlich-methodische Herausforderungen andererseits dienen.

In diesem kurzen Abriss über molekulares *Docking* zeigen sich zum Teil unterschiedliche Zielstellungen bzw. divergente Entwicklungen der *Docking*-Programme – so unterscheiden Huang *et al.* folgende Ziele bzw. Anwendungsgebiete³²⁹:

- Beste Bewertung der Pose, die dem experimentellen Bindemodus am meisten ähnelt
- Möglichst genaue Vorhersage der Bindeaffinität eines Liganden zum Rezeptor
- Virtuelles *Screening* zum Finden potenzieller Liganden aus einer Molekülbibliothek

Diese unterschiedlichen Auffassungen dessen, was molekulares *Docking* leisten kann oder können soll führen trotzdem dazu, dass sich nur wenige *Docking*-Programme etablieren, weil sich in vielen Fällen herausstellt, dass die Betonung eines bestimmten Aspektes der Modellierung wenig Allgemeingültigkeit besitzt.³⁶²

2 MOTIVATION UND ZIELSTELLUNG

2.1 Das wissenschaftliche Interesse an prenylierenden Enzymen

Ergebnisse aus der Grundlagenforschung zur Substrat- und Produktspezifität von Enzymen, die oft auch durch computerchemische Methoden erzeugt oder unterstützt werden, sind die Basis für Anwendungen in biotechnologischen Prozessen. Enzymatische Prenylierung hat potenzielle Anwendungen in der Wirkstoffproduktion, der Geruchs- und Geschmacksstoffindustrie oder im Bereich Kosmetik.^{123,138,162,363} Des Weiteren werden Ansätze verfolgt biotechnologisch mit prenylierenden Enzymen Kraftstoffe und Kautschuk zu erzeugen.^{185,364,365}

Auf der einen Seite ist die Substratpromiskuität einiger Terpenzyklen von großem Interesse für bioenzymatische Synthesen. Durch die Nutzung solcher Enzyme können unterschiedliche Ausgangsstoffe zu komplexen Kohlenstoffgerüsten aufgebaut werden. Auf der anderen Seite sind auch Enzyme mit breitem Produktspektrum für die bioenzymatische Synthese interessant. Derartige Enzyme sind vielversprechende Ausgangspunkte für zielgerichtete Mutationen, um ein bereits charakterisiertes Produktspektrum zu modulieren und für gewünschte Katalysen zu optimieren.

Auch die ökologische Wirkung insbesondere pflanzlicher Terpene ist ein weitreichendes Forschungsfeld, weil viele Pflanzenarten evolutiv spezifische Anpassungen ihres Terpenoidprofils an Umgebungsfaktoren entwickelt haben.¹ Ein tieferes Verständnis für die biochemischen Grundlagen der Produktion wichtiger Phytohormone wie Gibberelline und Cytokinine begründet das Interesse an DMAPP:Adenosinnukleosid-Transferasen und Kaurensynthase-ähnlichen Enzymen. Ein wesentlicher Blickpunkt im Bereich der Terpensynthesen ist die Modellierung der enzymatischen Produktbildung, in der vielfältige Katalyseprodukte entstehen. Einige KSL-Enzyme sind insofern besonders, als dass sie *Z*-Isomere von Prenyldiphosphaten als Substrate benutzen, die erst durch spezialisierte Enzyme bereitgestellt werden müssen.³⁶⁶ Über Stoffwechselwege, in denen *Z*-Prenyldiphosphate umgesetzt werden, ist weitaus weniger bekannt als über die herkömmliche Terpenbiosynthese.

2.2 Molekulares *Docking*

Die Vielfalt der existierenden *Docking*-Programme kann nicht darüber hinwegtäuschen, dass das hohe Ziel der schnellen Vorhersage von Bindeaffinitäten an Rezeptoren konzeptionelle und technische Weiterentwicklungen bedingt.^{329,362} Zusätzlich zur Frage, wie präzise *Docking* mit der Erzeugung diskreter Rezeptor-Ligand-Komplexe experimentelle Geometrien und mit *Scoring*-Funktionen experimentell ermittelte Affinitäten widerspiegeln kann, bestehen Ansatzmöglichkeiten um die Modellierung molekularer Interaktionen robuster und realistischer zu gestalten. Hierbei gibt es verschiedene Optionen, welche unterschiedliche Anwendungen des *Docking* unterstützen (nach Huang *et al.*,³²⁹):

- Schaffung eines großen Trainingsdatensatzes aus experimentellen Rezeptor-Ligand-Komplexen mit Affinitätsdaten
- Akkurater Einbezug von Solvatations- und Entropiebeiträgen in *Scoring*-Funktionen, um Affinitäten berechnen zu können

- Einführung von Mehrkörperpotenzialen zur besseren Beschreibung von Wechselwirkungsgeometrien oder detailliertere Atomtypisierung und entsprechende Parametrisierung
- Präzisere Potenziale zur Beschreibung von unzureichend modellierten Wechselwirkungen wie Metallionenkomplexierung

Diese Herausforderungen gehen Hand in Hand mit den Ansprüchen, die an molekulares *Docking* und seine Ergebnisse gestellt werden können.

Molekülmodellierungen und quantenchemische Berechnungen zeigen, dass in prenylierenden Enzymen die Wechselwirkungen zwischen intermediären Karbokationen-spezies und Aminosäureseitenketten maßgeblich die enzymatische Produktbildung beeinflussen.^{64,367} Insbesondere gibt es Hinweise darauf, dass die starken Wechselwirkungen zwischen Karbokationen und Aromaten den Reaktionsverlauf modulieren. Bisher unveröffentlichte Ergebnisse von Wolfgang Brandt, IPB Halle, zeigen ähnliche Umstände für die Wechselwirkung zwischen Prenylkationen und den freien Elektronenpaaren der Schwefelatome in Methioninseitenketten (Brandt *et al.*, in Vorbereitung). Diese Wechselwirkungen spielen potenziell eine wichtige Rolle in den Katalysen von Prenyltransferasen und Terpensynthesen. Sie werden durch Kraftfeldmethoden, die auch im molekularen *Docking* Anwendung finden, nicht modelliert.

2.3 Zielstellungen dieser Arbeit

Mittels virtuellem *Screening* sollen Vorhersagen für Inhibitoren der *Arabidopsis thaliana* DMAPP:ATP/ADP-Transferasen unterbreitet werden. Hierzu ist die Erstellung von Homologiemodellen für diese Enzyme auf Basis ihrer Aminosäuresequenzen notwendig. Anschließend besteht die Perspektive der chemischen Synthese vorgeschlagener potenzieller Inhibitoren durch Dimitar Vasilev am Leibniz-Institut für Pflanzenbiochemie in Halle (Saale). Bioaktivitätsuntersuchungen durch die Kooperationspartner Lukáš Spíchal, Václav Mik und Markéta Gemrotová (Palacký Universität, Olomouc, Tschechische Republik) können danach Aufschluss über die Aktivität der vorgeschlagenen Substanzen geben, um Struktur-Wirkungs-Beziehungen aufzustellen.

Ein weiteres Ziel dieser Arbeit ist die Homologiemodellierung folgender vier ausgewählter Kaurensynthase-ähnlicher Enzyme (KSL-Enzyme, für die Aminosäuresequenzen siehe A 6, Seite 151):

- *ent*-Kaurensynthase aus *Arabidopsis thaliana* (AtKS)
- *cis*-Abienolsynthase aus *Nicotiana tabacum* (NtABS)
- Santalen- und Bergamotensynthase aus *Solanum habrochaites* (ShSBS)
- β -Phellandrensynthase aus *Solanum lycopersicum* (SIPHS)

Diese Enzyme sind teilweise biochemisch charakterisiert und katalysieren die in Abbildung 15 dargestellten Umsetzungen.

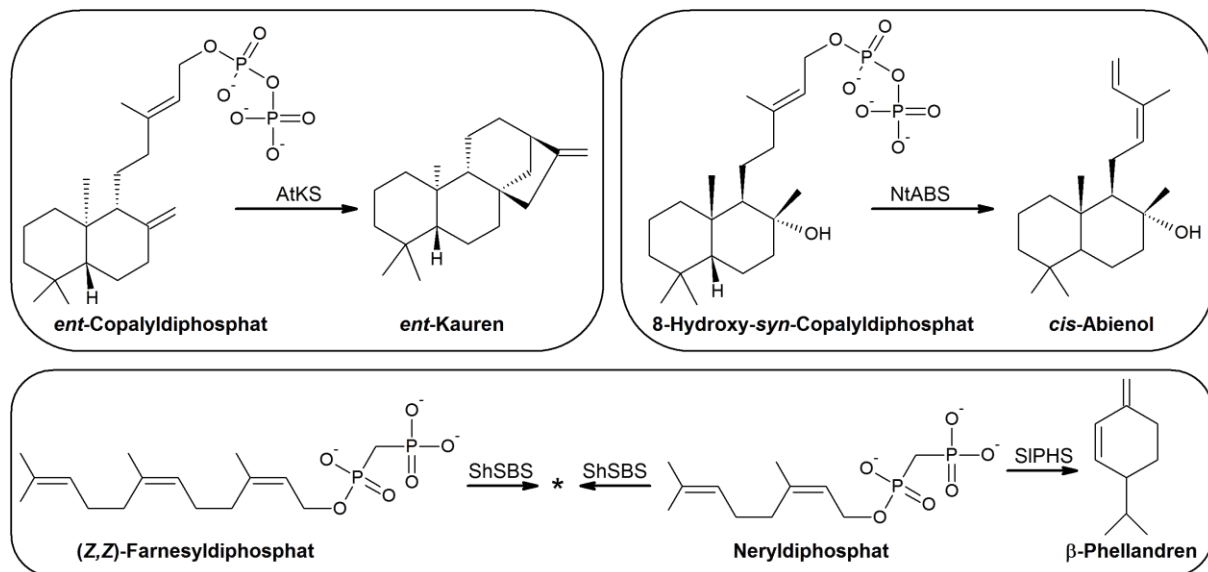


Abbildung 15: Substrate und Hauptprodukte der vier betrachteten KSL-Enzyme. Die Enzymkatalyse läuft jeweils unter Abspaltung von Diphosphat ab. Für die ShSBS-Produkte (*) aus (Z,Z)-FPP siehe Tabelle 20, Seite 89, sowie Abbildung 49, Seite 86, und Abbildung 50, Seite 87. Für ShSBS-Produkte aus Neryldiphosphat siehe Tabelle 18, Seite 82.

Die erstellten Modelle sollen Ausgangspunkt für weitere Untersuchungen bezüglich der Produktbildung sein. Es besteht die Absicht, zu diesem Zweck molekulares *Docking* von Intermediaten der enzymkatalysierten Reaktionen zu benutzen. Dabei stellen sich folgende Fragen:

- Inwiefern kann molekulares *Docking* benutzt werden, um Intermediate einer enzymkatalysierten Reaktion sinnvoll in die Proteinbindetasche einzupassen?
- Können aus *Docking*-Experimenten Vorschläge zu Übergangszuständen und deren Stabilisierung während der enzymatisch katalysierten Reaktion abgeleitet werden?
- Ist das *Docking* von Intermediatstrukturen ein geeignetes Verfahren, um Inhibitoren zu finden?
- Wie wird die Produktbildung in den ausgewählten KSL-Enzymen moduliert? Was sind die Ursachen für unterschiedliche Produkte in Enzymen, die sich in Primär- und Tertiärstruktur ähneln?

Die Modellierung von Wechselwirkungen zwischen Karbokationen und Aromaten bzw. Schwefelatomen von Methioninseitenketten im Enzym ist ein Kernpunkt für die Adressierung der obigen Fragen. Eine entsprechende Modifikation eines *Docking*-Programms wird angestrebt, um die Aussagekraft von *Docking*-Posen intermediärer Karbokationspezies in Terpensynthasen zu untersuchen und ihre Aussagekraft zu bewerten.

NtABS und ShSBS werden im Labor von Romy Töpfer am Leibniz-Institut für Pflanzenbiochemie Halle charakterisiert. Auf der Grundlage der Homologiemodelle sollen Vorschläge für zielgerichtete Mutagenesen an diesen Enzymen erarbeitet werden. Von der entsprechenden Rückkopplung aus den biochemischen Charakterisierungen werden weitere Hinweise auf die Mechanismen der spezifischen Produktbildung erwartet, die zu neuen Modellhypothesen führen können.

3 METHODEN

Dieses Kapitel gibt die bei der Erstellung dieser Arbeit verwendeten Methoden wider. Sofern nicht anders angemerkt, wurden sämtliche Berechnungen mit den Standardeinstellungen der jeweiligen Software bzw. Eingabemasken durchgeführt.

3.1 Homologiemodellierung

Dieser Abschnitt beschreibt das Verfahren zur Erstellung diverser Homologiemodelle und verwendete Methoden zur Beurteilung und Verfeinerung derselben.

3.1.1 Homologiemodellierung mit YASARA

Das Programm YASARA³⁶⁸ beinhaltet mehrere Protokolle zur Homologiemodellierung von Proteinen. Das Vorgehen im Standardverfahren umfasst dabei folgende Schritte:

- Falls keine *Template*-Struktur vorgegeben wurde:
 - PSI-BLAST³⁶⁹ der Zielsequenz in der Uniprot-Datenbank²⁴⁵ zur Erstellung einer PSSM (engl. *position-specific scoring matrix*, positionsspezifische Bewertungsmatrix) mit deren Hilfe aus der PDB (Datenbank mit über 70.000 experimentell aufgeklärten Proteinstrukturen)³⁹ potenzielle *Template*-Strukturen ausgewählt werden
 - Sortierung der *Template*-Strukturen nach *Alignment*- und Strukturqualität gemäß WHAT_CHECK³⁷⁰ aus der PDBfinder2-Datenbank³⁷¹
 - Homologiemodelle werden nur für die besten *Template*-Strukturen erstellt.
- Zusatzinformationen werden ggf. für die Erstellung von *Alignments* zwischen allen *Templates* und dem Zielprotein einbezogen, wie zum Beispiel vorhergesagte Sekundärstrukturen des Zielproteins oder Strukturelemente des *Template*, welche Einfluss auf Lückenbildung im *Alignment* haben könnten.³⁷² Die *Alignments* werden in Anlehnung an Bewertungsmatrizen des SSALN-Algorithmus erstellt.³⁷¹
- *Alignments* mit schlechter Abdeckung bestimmter Bereiche können zur Ablehnung einer *Template*-Struktur führen, wenn auch alternative, mit einer stochastischen Methode (King und Sternberg,³⁷²) erzeugte *Alignments* keine Verbesserung erbringen.
- Bei oligomeren *Template*-Strukturen werden ggf. Oligomermodelle erstellt.
- Bei Insertionen bzw. Deletionen im *Alignment* wird eine spezielle Untermenge der PDB für die Auswahl geeigneter *Loop*-Strukturen durchsucht.
- Falls im *Template* Liganden vorhanden sind, werden diese automatisch parametrisiert und inklusive ihrer intermolekularen Wechselwirkungen bei der Rückgratmodellierung gemäß des verwendeten YASARA2-Krauffelds mit einbezogen.³⁰⁶
- Initiale Modellierung der Aminosäureseitenketten entsprechend einer repulsiven Energiefunktion³⁷³
- Optimierung der *Loop*-Regionen durch Modellierung zahlreicher Konformationen bei jeweiliger Optimierung der Seitenkettenkonformationen
- Verfeinerung der Seitenkettenrotamere unter Berücksichtigung elektrostatischer Interaktionen und empirischer Packungsterme³⁰⁶
- Vollständige Protonierung und anschließende Optimierung des Wasserstoffbrückennetzwerks

- Jeweilige Energieoptimierung der vorläufigen Modelle mit expliziter Lösungsmittelmodellierung (Wasser) im YASARA2-Krauffeld
- Modellvalidierung durch eine Reihe von Tests des Gesamtmodells auf Abweichungen von empirischen Standardwerten, ausgedrückt durch z-Scores
- Aminosäureweise Berechnung von z-Scores
- Erstellung eines Hybridmodells durch sukzessiven Austausch von schlecht bewerteten Regionen im bestbewerteten Modell durch die entsprechenden Modellregionen aus anderen Homologiemodellen

Die Anführung eines z-Score z gibt an, wie viele Standardabweichungen σ ein Wert x von einem (in der Regel) empirisch ermittelten Normwert μ abweicht:

$$z = \frac{x - \mu}{\sigma}$$

Gleichung 1: Berechnung eines z-Score

Folgende Qualitätskriterien werden entsprechend des verwendeten Krauffeldes mit z-Scores ausgewertet:

- Positionierung von Wassermolekülen
- Bindungslängen, Bindungswinkel und Torsionswinkel
- Coulomb- und Van-der-Waals-Wechselwirkungen

Nur für das YASARA2-Krauffeld kommen folgende Kriterien hinzu:

- *Packing1D*: distanzabhängige Packungswechselwirkungen
- *Packing3D*: richtungsabhängige Packungswechselwirkungen, nur für gängige Atomtypen in Proteinen und DNA/RNA

Die Modellqualität insgesamt wird durch eine gewichtete Summe über die Kriterien *Torsionswinkel*, *Packing1D* und *Packing3D* bewertet:

$$\text{Insgesamt} = 0,145 \cdot \text{Torsionswinkel} + 0,390 \cdot \text{Packing1D} + 0,465 \cdot \text{Packing3D}$$

Gleichung 2: Berechnung von z-Score „Insgesamt“ in YASARA zur Bewertung von Proteinmodellen

Die energetische Bewertung ist bei der Berechnung von z-Scores so normiert, dass negative Werte zu Stande kommen, wenn die entsprechende Kriteriumsabweichung ungünstiger ist als in dem Datensatz, auf dessen Basis diese Normen gebildet werden.

3.1.2 Isopentenyltransferasen aus *Arabidopsis thaliana*

Den Ausgangspunkt für die Homologiemodellierung bilden neun als Isopentenyltransferasen annotierte Aminosäuresequenzen (bereitgestellt von Dr. Lukáš Spíchal, Palacký Universität, Olomouc, siehe A 1, Seite 144) mit den folgenden NCBI-Identifikatoren:

Tabelle 4: Genlokus-Identifikatoren der AtIPT für die NCBI-Datenbank

	AtIPT1	AtIPT2	AtIPT3	AtIPT4	AtIPT5	AtIPT6	AtIPT7	AtIPT8	AtIPT9
Lokus	BAB 59040	BAB 59042	BAB 59043	BAB 59044	BAB 59041	BAB 59045	BAB 59046	BAB 59047	BAB 59048

Wie in Abschnitt 1.1.7.2 bereits erwähnt, sind AtIPT2 und AtIPT9 als DMAPP:tRNA-Transferasen annotiert und liegen somit außerhalb des Interessenbereichs dieser Arbeit.

3.1.2.1 Voruntersuchungen

Eine standardmäßige Homologiemodellierung mit MOE ⁽²⁸⁶⁾ zur Sequenz von AtIPT1 basierend auf einer lokalen Kopie der Proteindatenbank PDB (Stand 2007) lieferte *Template*-Vorschläge. Mit MOE wurde die Sequenzidentität zwischen den ermittelten *Template*-Kandidaten und den AtIPT-Sequenzen untersucht. Zur Suche nach weiteren homologen Proteinen wurde eine blastp-Suche ^(369,374,375) mit der Sequenz von AtIPT1 durchgeführt. Entsprechende *Alignments* und Distanzberechnungen erfolgten mit ClustalW ^(376,377) und eine Baumdarstellung der Sequenzähnlichkeit mit dem Werkzeug „Interactive Tree Of Life“ (iTOL). ^{378,379}

3.1.2.2 Vorbereitungen für die Homologiemodellierungen

Für die Erstellung von Homologiemodellen für AtIPT1 bzw. AtIPT3 bis AtIPT8 wurde jeweils ein *Template* manuell erstellt, um die Positionierung der Liganden ATP und DMAPP in die Modellierung einzubeziehen: Durch strukturelles *Alignment* der beiden PDB-Röntgenkristallstrukturen 3A8T (*Humulus lupulus* Isopentenyl:Adenylat-Transferase – HIIPT – mit ATP und einem Phosphation in der Bindetasche) und 2ZE7 (*Agrobacterium tumefaciens* Isopentenyltransferase mit AMP, Dimethylallyl-S-thiolodiphosphat (DMASPP) und Zn⁺⁺-Ion in der Bindetasche) mit MOE konnte entsprechend eine Positionierung für die Prenyldonorstruktur in der Bindetasche der pflanzlichen Isopentenyl:Adenylat-Transferase vorgeschlagen werden. Für einen Vergleich der beiden hier benutzten Strukturen und eine Analyse der Adenylatbindung siehe Chu *et al.*⁵⁵. Diese Publikation beinhaltet bereits Vorschläge zur Positionierung von DMAPP in der Bindetasche der HIIPT, die ebenfalls von der Struktur der Isopentenyltransferase aus *A. tumefaciens* (2ZE7) abgeleitet wurden – analog dem folgenden Verfahren.

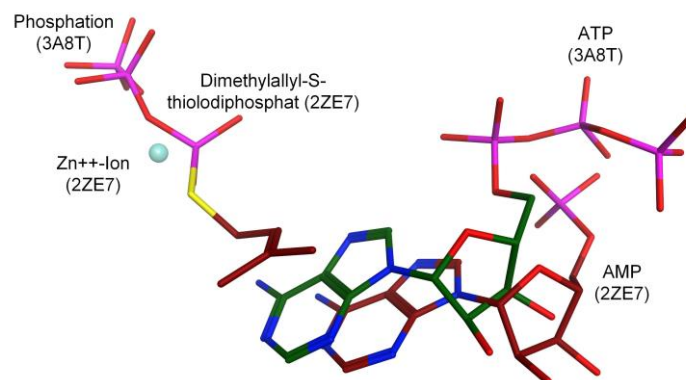


Abbildung 16: Relative Lage der Liganden aus HIIPT und *Agrobacterium tumefaciens* IPT (HIIPT: 3A8T, grüne Kohlenstoffatome; *A. tumefaciens* IPT: 2ZE7, braune Kohlenstoffatome) bei strukturellem *Alignment* der gesamten Röntgenkristallstrukturmodelle mit MOE

Die abgeleitete Pose des Prenyldonormoleküls DMASPP (inkl. Zink-Gegenion) in der Bindetasche des pflanzlichen Enzymmodells zeigte sterische Kollisionen mit in der Röntgenkristallstruktur aufgelösten Wassermolekülen. Diese Wassermoleküle und das in der Bindetasche positionierte Phosphation wurden entsprechend gelöscht und die Positionierung des DMASPP in die Bindetasche der Enzymstruktur aus Hopfen übernommen. Mittels des LigX-Moduls von MOE erfolgte eine Einpassung des neuen Liganden durch lokale Energieoptimierungen (MMFF94x-Kraftfeld) in die Bindetasche des *Template*-Modells. Außerdem wurde mit dem MOE-Modul Protonate3D der Protonierungszustand in der Bindetasche angepasst. Anschließend erfolgte eine manuelle Korrektur der Prenyldonorposition mit

Hinblick auf beschriebene Wechselwirkungen wie das konservierte *p-Loop*-Motiv zur Diphosphatkomplexierung.^{55,66,380} Das so erstellte Proteinmodell mit den Liganden DMASPP mit Zink-Ion und AMP wurde abschließend in YASARA mit dem YASARA2-Kraftfeld in einer mit Wassermolekülen gefüllten Lösungsmittelbox optimiert.

Die erzeugte Struktur wurde exemplarisch als *Template* für eine Homologiemodellierung mit YASARA für die AtIPT1-Sequenz verwendet. Danach wurde im resultierenden Strukturmodell das Schwefelatom des DMASPP durch ein Sauerstoffatom ersetzt. Eine folgende Energieoptimierung mit dem YASARA2-Kraftfeld in YASARA unter Benutzung des PM3-Ladungsmodells resultierte in dem für diese Arbeit finalen Modell für AtIPT1.

Für die Modellierung von AtIPT3 bis AtIPT8 wurde ebenfalls durch strukturelle Überlagerung des Hopfenenzym und des bakteriellen Enzyms ein Hybrid-*Template* erzeugt. In diesem Modell wurde die Positionierung des DMASPP in die Proteinstruktur der HIIPT übernommen und das Schwefelatom des DMASPP vor Beginn der Homologiemodellierung durch ein Sauerstoffatom ersetzt, um die nachträgliche Änderung zu ersparen. Aus demselben Grund wurden die kristallisierten Zinkionen durch Magnesiumionen ersetzt. Die resultierende Struktur wurde jeweils als *Template* für eine YASARA-Homologiemodellierung verwendet.

3.1.2.3 Bewertung der erstellten Proteinmodelle

Die Qualität der einzelnen Homologiemodelle wird von YASARA mit *z-Scores* und zusätzlich als wörtliche Interpretation ausgegeben. Die sieben erstellten Modelle wurden außerdem mit ProSA-web auf ihre qualitative Ähnlichkeit zu Röntgenkristall- bzw. NMR-Strukturmodellen untersucht.^{381,382} Mit MOE wurde eine Analyse der Proteingeometrien mit Auftragungen der ϕ - ψ -Winkelkombinationen der einzelnen Aminosäuren durchgeführt.

3.1.2.4 Vergleich der Homologiemodelle

Ein strukturelles *Alignment* der sieben erstellten AtIPT-Modelle und der beiden Röntgenkristallstrukturmodelle HIIPT und *A. tumefaciens*-IPT mit MOE sollte weiterhin Aufschluss über wesentliche Unterschiede in den Modellen geben. Um die Aminosäuren, welche in den Modellen an der Ligandenbindung direkt beteiligt sind, zu identifizieren, wurden in dem erzeugten strukturellen *Alignment* alle AMP- und DMAPP-Strukturen in den AtIPT-Modellen markiert, mit der Option „Extend – Near Residues“ selektiert und im *Alignment* rot gefärbt. Anschließend wurde eine grafische Zusammenstellung der *Alignment*-Segmente mit eingefärbten Aminosäuren erzeugt. Zur Menge der modellierten AtIPT-Sequenzen wurden die Sequenzen der beiden Röntgenkristallstrukturen der *A. tumefaciens* IPT und der HIIPT hinzugefügt. Für diese neun Sequenzen wurde ein multiples Sequenz-*Alignment* mit ClustalW berechnet.

Das paarweise *root-mean-square deviation* (RMSD, siehe Gleichung 3) wurde auf dem strukturellen *Alignment* mit MOE bezüglich der Proteinrückgratotope berechnet, um Tertiärstrukturunterschiede quantifizieren zu können.

$$RMSD(A, B) = \sqrt{\frac{1}{N} \sum_{i=1}^N d(A_i, B_i)^2}, \text{ mit } A_i \in A \text{ und } B_i \in B \text{ für } i \in \mathbb{N}, 1 \leq i \leq N$$

Gleichung 3: Berechnung des RMSD für zwei Atommengen *A* und *B* mit gleicher Kardinalität *N*. Als Distanzfunktion *d* fungiert der euklidische Abstand.

3.1.3 Kaurensynthese-ähnliche Enzyme

Die Aminosäuresequenzen der vier KSL-Enzyme AtKS, NtABS, ShSBS und SIPHS wurden von Prof. Alain Tissier, Leibniz-Institut für Pflanzenbiochemie, Halle (Saale), zur Verfügung gestellt. Sie dienen als Ausgangspunkt für die Untersuchung von Sequenzhomologien und die Erstellung von Homologiemodellen.

3.1.3.1 Voruntersuchungen

Eine Eingabe der AtKS-Sequenz in blastp (Version 2.2.26, ^{369,374,375}) liefert zahlreiche signifikant ähnliche Aminosäuresequenzen (e-Wert kleiner als 10^{-100}) und ordnet die Sequenz in den Bereich pflanzlicher Terpenzyklen ein. Fünf Merkmale werden innerhalb der Sequenz annotiert:

- Substratbindetasche
- „Deckel“-Motiv für die Bindetasche
- Metallion-bindendes Motiv
- Aspartat-reiche Region 1
- Aspartat-reiche Region 2

Unter den annotierten Sequenzen finden sich auch die vier zu modellierenden KSL-Enzyme. Ein standardmäßiger Start von Homologiemodellierungen mit YASARA wurde für jedes der vier Zielproteine durchgeführt. Die Sequenzähnlichkeit der verwendeten *Template*-Strukturen zu den Aminosäuresequenzen der Zielproteine wurde mittels ClustalW untersucht und mit Hilfe von iTOL visualisiert. Für die vier Zielproteine wurde jeweils eine Standard-YASARA-Homologiemodellierung mit der *Template*-Struktur der *Taxus brevifolia* Taxadiensynthase (TbTS, PDB-Identifikator 3P5R, Köksal *et al.*, ⁴⁷) durchgeführt.

3.1.3.2 Bewertung der erstellten Proteinmodelle

Die resultierenden Homologiemodelle wurden mit YASARA und ProSA-web auf ihre Qualität untersucht. Um die Unterschiede einzelner Aminosäuren zwischen den modellierten Proteinen und dem *Template* zu untersuchen, wurden mit MOE ein Sequenz- und ein Struktur-*Alignment* erstellt.

3.1.3.3 Docking von Substratstrukturen und strukturelle Verfeinerung

Aus den erstellten Proteinmodellen wurden manuell mit MOE die Ligandstrukturen der Homologiemodellierung gelöscht und mit GOLD^{289,326} entsprechende Substratstrukturen in die Bindetaschen gedockt. Die verwendeten Eingaben sind in Tabelle 5 erfasst.

Tabelle 5: Verwendete Einstellungen für das Docking mit GOLD. Die Mittelpunkte sind als (x, y, z)-Tripel notiert. Als Bindetaschenradius wurden jeweils 10 Å eingegeben.

	AtKS	NtABS	ShSBS	SIPHS
Ligand	jeweils <i>ent</i> -Copalyldiphosphat		(Z,Z)-FPP	Neryldiphosphat
Mittelpunkt	140; -5; 85	140; -6; 85	137; -5; 88	140; -5; 85

Die Komplexe mit der besten *Score*-Bewertung nach GOLD wurden anschließend in YASARA in einer mit Wasser als Lösungsmittel gefüllten Standardsimulationsbox mit dem YASARA2-Kraftfeld geometrieoptimiert. Alle Aminosäuren der Modelle, die ein Atom mit einem Abstand von höchstens 4,5 Å von einem Ligandatom besitzen, wurden mit MOE ausgewählt und im Sequenz-*Alignment* farblich markiert. Für diese Aminosäuren erfolgt eine detaillierte Diskussion möglicher Funktionen.

3.1.3.4 Mutationsvorschläge und Aktivitätsuntersuchungen

Basierend auf den erstellten Homologiemodellen wurden die Modelle von NtABS und ShSBS verglichen und Vorschläge für zielgerichtete Mutationen abgeleitet, die zur wechselseitigen Umfunktionalisierung führen sollen. Vorgeschlagene Mutationen und anschließende Enzymaktivitätstests mit verschiedenen Substraten wurden von Romy Töpfer am Leibniz-Institut für Pflanzenbiochemie durchgeführt. Für ShSBS und NtABS wurde dazu jeweils rekombinantes Protein aus *Escherichia coli* aufgereinigt und für die Aktivitätsuntersuchungen verwendet. Für ShSBS kam zusätzlich ein transienter Test zum Einsatz. Dafür wurde ein Vektor mit den Genen für (*Z,Z*)-Farnesyldiphosphatsynthase aus *Solanum habrochaites* und ShSBS in *Agrobacterium tumefaciens* geklont. Die transgenen Bakterien wurden zur Transfektion von *Nicotiana benthamiana*-Pflanzen verwendet. Die Extraktion von Sesquiterpenoiden erfolgte mittels SPME (*solid phase micro extraction*, engl. eine Extraktion durch Adsorption an eine feste Phase) oder *head space extraction* (engl. Dampfphaseextraktion: thermisch beschleunigte Extraktion in die Gasphase). Diterpene wurden in einer Hexanphase gelöst und aufdestilliert. Die Produktidentifizierung erfolgte per GC-MS (Gaschromatographie gekoppelt mit Massenspektrometrie).

3.2 Suche nach Inhibitoren für *Arabidopsis thaliana* Isopentenyltransferasen

Die in diesem Abschnitt dargelegten Methoden wurden auf Basis der erstellten Homologiemodelle angewendet. Sie werden hier mit ableitbaren Erkenntnissen aus den Röntgenkristallstrukturmodellen der *A. tumefaciens* IPT mit aufgelösten Liganden (siehe Tabelle 12, Seite 57) sowie mit Homologieschlüssen anhand der HIPT kombiniert. Für die Inhibitorsuche wurde zugrunde gelegt, dass eine möglichst spezifische Wechselwirkung mit der DMAPP-Bindetasche stattfinden soll. Hintergrund für diese Anforderung ist, dass *in vivo* etliche Enzyme Adenosinmono-, -di- und -triphosphate umsetzen und eine spezifische Inhibition auf Basis der Nukleosidphosphatbindetasche unrealistisch ist.^{383,384}

3.2.1 Pharmakophorsuche

Für die Suche nach Liganden der DMAPP:ATP/ADP-Transferasen aus *A. thaliana* wurde ein virtuelles *Screening* mittels Pharmakophorsuche in Konformationsdatenbanken durchgeführt. Die durchsuchten Konformationsdatenbanken umfassten für die Pharmakophorsuche vorbereitete Datenbanken in der Abteilung Natur- und Wirkstoffchemie am Leibniz-Institut für Pflanzenbiochemie Halle (Saale) und sind in Tabelle 6 mit wesentlichen Merkmalen aufgelistet.

Tabelle 6: Kurzcharakterisierung der Strukturdatenbanken für die Pharmakophorsuche

	Anzahl Verbindungen	Anzahl Konformationen	Stand
Beilstein	162.258	ca. 470.000	27.05.2010
MOE	2.824.292	nicht ermittelt	November 2009
Makrolide	1.241	11.608	27.12.2006
hauseigen	11.502	12.924	11.06.2008
Spresi	37.795	67.936	20.12.2006

Die Pharmakophordefinition basiert auf einer DMASPP-Konformation, die aus einem kraftfeldoptimierten Homologiemodell stammt. Sie ist in Abbildung 17 veranschaulicht. Es wurden drei Merkmale entsprechend der Diphosphatgruppe als anionische Merkmale oder Wasserstoffbrückenakzeptoratome definiert. Zusätzlich wurde in Analogie zur Komplexierungsposition des Metallions ein kationisches oder Wasserstoffbrückendonator-Merkmal festgelegt. Das fünfte Merkmal ist ein Wasserstoffbrückenakzeptor-Merkmal am Brückenatom zwischen Diphosphat und organischem Rest. Um eine Größenbeschränkung der Treffermoleküle zu erreichen, wurden manuell 27 Ausschlussvolumina definiert, so dass nur in Richtung des Dimethylallylrestes weitere Anknüpfungen möglich sind.

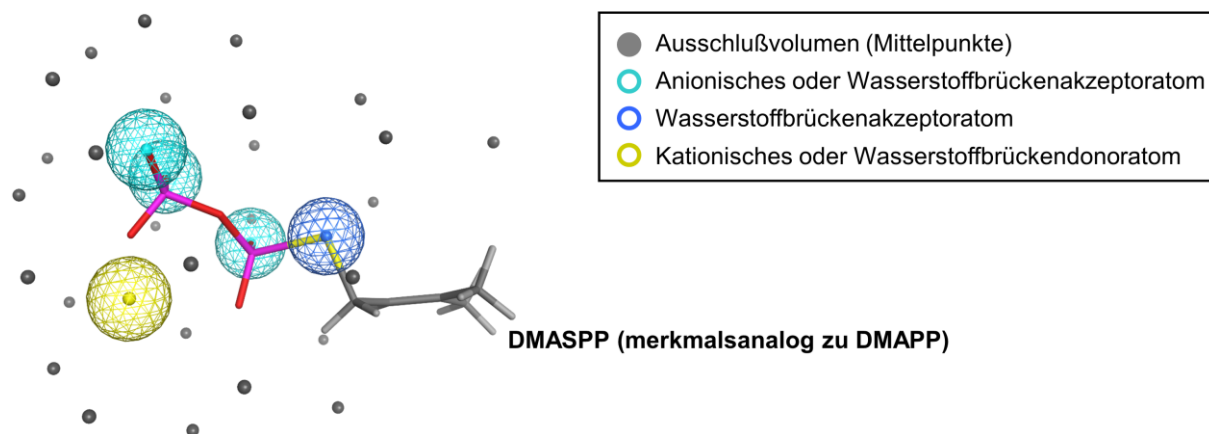


Abbildung 17: Grafische Darstellung (MOE) der verwendeten Pharmakophordefinition. Die 27 Ausschlussvolumina besitzen jeweils einen Radius von 2,4 Å (hier sind nur die Mittelpunkte dargestellt), während die fünf echten Pharmakophormerkmale jeweils mit einem Radius von 1 Å definiert sind. Bis auf das Kationen-/Wasserstoffbrückendonator-Merkmal basieren die Pharmakophormerkmale auf Atompositionen der zu Grunde gelegten DMASPP-Konformation. Für eine textuelle Darstellung des Pharmakophors siehe A 8, Seite 154.

Die Anzahl der Merkmale, welche bei der Datenbanksuche erreicht werden muss, um einen Datenbankeintrag als Treffer zu werten, wurde empirisch so festgelegt, dass die Anzahl der Ergebnisse im Bereich weniger hundert bis einiger tausend Konformationen liegt. Durch diese Anforderung ergab sich, dass mindestens vier der fünf Merkmale erfüllt sein müssen.

3.2.2 Vorschlag und Verifikation potenzieller AtIPT-Liganden

Die Ergebnisse des virtuellen *Screening* inspirierten einige potenzielle AtIPT-Liganden für anschließende Herstellung und Testung. Hierbei wurde Wert darauf gelegt, dass die Strukturen ein relativ explizites Analogon zum Prenylrest enthalten. Eine gewisse Diversität wurde angestrebt, um auf der Basis experimenteller Ergebnisse Struktur-Wirkungsbeziehungen aufstellen zu können.

Dimitar Vasilev synthetisierte ausgewählte Inhibitorvorschläge. Anschließend wurden diese von Markéta Gemrotová an der Palacký Universität, Olomouc, Tschechische Republik, mit ELISA-Tests an AtIPT1 auf ihre inhibitorischen Eigenschaften getestet und analysiert (ELISA: *Enzyme-linked immunosorbent assay*, engl., ein enzymgekoppeltes Immunadsorptionsverfahren). Als Resultat der Aktivitätstests wurden für die 17 getesteten Substanzen IC_{50} -Werte berechnet (*inibitory concentration*, engl. Inhibitionskonzentration; hier: Konzentration des Liganden, bei der 50% des Enzyms inhibiert sind). Die Enzymaktivität wurde durch Beobachtung der Produktbildung im ELISA mit HPLC-MS/MS (*high performance liquid chromatography*, engl. Hochleistungsflüssigkeitschromatographie, hier

gekoppelt mit Tandem-Massenspektrometrie) ermittelt. Die höchste getestete Konzentration der potenziellen Inhibitoren ist 300 μM und die errechneten IC_{50} -Werte wurden entsprechend extrapoliert. Ferner überprüfte Markéta Gemrotová die Fähigkeit von AtIPT1, AtIPT3, AtIPT4, AtIPT5, und AtIPT8 als alternative Substrate *cis*- bzw. *trans*-HMBPP (in *cis*- bzw. *trans*-Position hydroxyliertes DMAPP) umzusetzen.

3.3 Molekulares *Docking* von Reaktionsintermediaten in *Mentha spicata* Limonensynthese und *Solanum habrochaites* Santalen- und Bergamotensynthese

Um das Potenzial von molekularem *Docking* für die Auf- und Erklärung der Bindemodi von Intermediaten von Terpensynthesereaktionen zu untersuchen, wurden umfangreiche *in silico*-Experimente durchgeführt. Für die Umsetzung dieses Ansatzes kamen selbst erstellte *Shell*-Skripte, für solche Skripte interpretierbare Anweisungsdateien und Skript-basierte Auswertungen bzw. grafische Darstellungen mit der Software R³⁸⁵ zum Einsatz. Die Untersuchungen zur Positionierung von kationischen Reaktionsintermediaten in den Bindetaschen zweier Terpensynthesen wurden mit dem *Docking*-Programm PLANTS (engl. **Protein Ligand ANT System**, Protein-Ligand-Ameisen-System, Korb *et al.*,³⁸⁶) durchgeführt. Der Quellcode des Programms wurde freundlicherweise von Dr. Oliver Korb (Cambridge Crystallographic Data Centre) und Prof. Dr. Thomas Exner (Universität Konstanz) zur Verfügung gestellt. Das ermöglichte die Anpassung der CHEMPLP-*Scoring*-Funktion in PLANTS für bestimmte Wechselwirkungen kationischer Reaktionsintermediate der enzymkatalysierten Reaktionen mit Rezeptoratomen sowie die Untersuchung weiterer Parameter auf die Modellierung der Rezeptor-Ligand-Wechselwirkungen bei Terpensynthesen mit PLANTS.

3.3.1 Das *Docking*-Programm PLANTS

Dieser Abschnitt beschreibt die wesentlichen Charakteristika des *Docking*-Programms PLANTS analog der Originalpublikation von Korb und Kollegen.³⁸⁶ Sie veröffentlichten 2006 die erste Anwendung von ACO (engl. **Ant Colony Optimization**, Ameisenkolonie-Optimierung) auf die Optimierung von Rezeptor-Ligand-Wechselwirkungen. Die grundsätzlichen Komponenten dieses *Docking*-Programms sind Atomtypisierung, verschiedene Kraftfeldterme zur Beschreibung inter- und intramolekularer Wechselwirkungen sowie die stochastische Optimierung einer *Scoring*-Funktion mittels ACO im Definitionsbereich folgender Freiheitsgrade (entsprechend Korb *et al.* 2007,³⁸⁷):

- Drei Freiheitsgrade für die Rotation des Liganden
- Drei Freiheitsgrade für die Translation des Liganden
- Ein Freiheitsgrad für jeden drehbaren Torsionswinkel des Liganden
- Ein Freiheitsgrad für jeden drehbaren Torsionswinkel von entsprechend zu berücksichtigenden Aminosäureseitenketten des Rezeptorproteins
- Ein Freiheitsgrad für jedes potenzielle Wasserstoffbrückendonatoratom (Hydroxyl- und Aminogruppenwasserstoffatome)

Die Angabe eines Punktes und eines Radius durch den Benutzer ergibt einen kugelförmigen Definitionsbereich für die Translationsfreiheitsgrade und soll grob die Bindetasche des Rezeptors orten. Die Translationsfreiheit wird zu Intervallen von jeweils 0,1 Å diskretisiert. Für die Rotationsfreiheitsgrade des Liganden und auch die relevanten Torsionswinkel (im Fall potenzieller Wasserstoffbrückendonatoratome Bindungswinkel) von Ligand und Rezeptor

kommt eine Diskretisierung zu Schritten von je 1° zur Anwendung. Es resultiert Gleichung 4 als formale Beschreibung des Optimierungsproblems.

$$\min_{\vec{x} \in \mathbb{R}^n} f(\vec{x}) : \mathbb{R}^n \mapsto \mathbb{R}$$

wobei $\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, $n \in \mathbb{N}$ und $x_i \in \mathbb{N}$: $0 \leq x_i \leq 360$, für $i \in \mathbb{N}$: $1 \leq i \leq n$, falls x_i relevanter Torsions- oder Rotationswinkel;

$$\begin{pmatrix} x_i \\ x_{i+1} \\ x_{i+2} \end{pmatrix} \in \mathbb{R}^3 \text{ mit } \begin{pmatrix} x_i \\ x_{i+1} \\ x_{i+2} \end{pmatrix} = \vec{P} + r\vec{m}, i \in \mathbb{N}: 1 \leq i \leq n - 2 \text{ für } \vec{P} \in \mathbb{R}^3, r \in \mathbb{R}: r > 0, \text{ mit}$$

$$\vec{m} = 0,1 \cdot \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix} \text{ für } m_x, m_y, m_z \in \mathbb{N}: 0 \leq m_x, m_y, m_z \leq 10r, \text{ falls } \begin{pmatrix} x_i \\ x_{i+1} \\ x_{i+2} \end{pmatrix} \text{ Translationsvektor}$$

Dabei sind

$f(\vec{x})$	Scoring-Funktion
\vec{x}	Vektor der Freiheitsgrade
n	Anzahl der Freiheitsgrade
x_j	Einzelner Freiheitsgrad ($j \in \mathbb{N}: 1 \leq j \leq n$)
\vec{P}	Gegebener Mittelpunkt (in den Koordinaten des Rezeptors)
r	Gegebener Radius der Bindetasche (in Å).

Gleichung 4: Optimierungsproblem der Scoring-Funktion

Die Anzahl der Freiheitsgrade resultiert somit aus der Größe der Bindetasche (gegebener Radius), der Flexibilität der Bindetasche (als flexibel definierte Aminosäureseitenketten) und der Flexibilität des Liganden (jeweils Anzahl der rotierbaren Torsionswinkel unter Vernachlässigung von Wasserstoffatomen, die weder Wasserstoffbrückendonoren noch -akzeptoren sind).

3.3.1.1 Erzeugung von Docking-Posen

Durch die beschriebene Diskretisierung der Zielfunktion ist die Anwendung des *Max-Min-Ant System*-Algorithmus von Stützle und Hoos möglich.³⁸⁸ Eine formale Beschreibung analog der Originalpublikation findet sich im Anhang ab Seite 155 (A 9). Die Verwendung eines Ameisenalgorithmus zur Erstellung von Molekülkonformationen auf einer *Docking-Scoring*-Funktion basiert auf einem stochastischen Rückkopplungsmechanismus zur iterativen Lösung des Optimierungsproblems durch Pheromonspuren, die als Pheromon-Vektoren gespeichert werden. Diese sorgen in Folgeiterationen des Algorithmus mit größerer Wahrscheinlichkeit für Werterealisationen der Freiheitsgrade, die sich in vorangegangenen Iterationen als günstig entsprechend der *Scoring*-Funktion erwiesen, indem sie nach jeder Iteration in Form einer Rückkopplung durch die *Scoring*-Funktion verändert werden (siehe Gleichung 7, Seite 156).

In jeder Iteration des PLANTS-Algorithmus wird eine bestimmte Anzahl von *Docking*-Lösungen erzeugt, welche die Autoren des Programms mit der Anzahl simulierter Ameisen gleichsetzen, die entsprechenden Pheromonspuren folgen. Diese Ameisenkoloniegröße kann vom Benutzer gewählt werden. Die Iterationszahl des Algorithmus hängt von der Anzahl frei drehbarer Bindungen und der Anzahl von Schweratomen des Liganden ab (siehe Gleichung 8, 156). Ein Skalierungsfaktor erlaubt die Steuerung des Kompromisses zwischen Ergebnishüte und Rechenaufwand. Die Ergebnishüte behandelten Korb und Kollegen als

den Anteil aller *Docking*-Lösungen auf einem spezifischen Rezeptor-Ligand-Datensatz, in dem die Posen einen geringeren RMSD-Wert als 2 Å von der experimentell aufgeklärten Wechselwirkungsgeometrie besitzen.³⁸⁶

PLANTS verwendet einen Simplex-Algorithmus für die lokale Optimierung von *Docking*-Posen. Unter Verwendung dieses von Nelder und Mead (³⁸⁹) beschriebenen Optimierungsalgorithmus wird für alle Posen versucht, die lokale Lösung zu verbessern. Anschließend wird eine verfeinerte Optimierung mit stärkerer Konvergenzbedingung durchgeführt.

3.3.1.2 Diversifikationsheuristik

Um trotz des Rückkopplungsmechanismus der Pheromonaufdatierung neue Werte für Freiheitsgrade zu realisieren, bedient sich PLANTS mehrerer heuristischer Ansätze:

- Bei der Neuberechnung der Pheromonspuren werden nicht absolut lokale, sondern leicht regionale Rückkopplungen verwendet, um die Wahrscheinlichkeit zu erhöhen, Werte in einem Bereich um ein gefundenes lokales Optimum erneut zu realisieren (siehe Gleichung 7, Seite 156).
- Damit trotz positiver Rückkopplung weiterhin verschiedene Konformationen erzeugt werden, wird der Effekt der Pheromonverdunstung modelliert, so dass hohe Pheromonwerte im Laufe mehrerer Iterationen abgetragen werden und stochastisch neue Realisierungen zu Stande kommen (siehe Verdunstungsrate ρ in Gleichung 7, Seite 156).
- Pheromonwerte sind nach oben und unten fest begrenzt, damit Freiheitsgrade grundsätzlich alle Werte annehmen können (siehe Gleichung 9, Seite 156).
- PLANTS führt explizit Diversifikation durch: Wenn mehr als zehn der (*score*-wertig) besten *Docking*-Lösungen einer Iteration sich untereinander im *Score* um weniger als 2% des *Score* der (*score*-wertig) besten Lösung seit der letzten Diversifikation unterscheiden, wird erneut eine Diversifikation durchgeführt. Hierzu werden die Pheromonspuren durch eine Aufwertung von geringen Pheromonwerten geglättet (siehe Gleichung 12 in ³⁸⁸, für PLANTS wird ein Glättungsfaktor von 0,5 verwendet). Im Falle dreier aufeinanderfolgender Pheromonspurglättungen werden alle Pheromonwerte reinitialisiert.

3.3.1.3 Clustering der Docking-Lösungen

Nach Ablauf aller Iterationen liefert der PLANTS-Algorithmus die am besten bewertete *Docking*-Lösung und alle *Docking*-Lösungen, die durch die lokale und die verfeinerte lokale Optimierung erzeugt wurden. Nach der Sortierung der *Docking*-Posen entsprechend der *Scores* werden aus den am besten bewerteten Ligandenkonformationen standardmäßig zehn Konformationen ausgewählt, so dass das paarweise RMSD mindestens 2 Å beträgt. Sowohl die Anzahl von Konformationen als auch der RMSD-Grenzwert können vom Benutzer verändert werden.

3.3.1.4 Scoring in PLANTS

Für PLANTS wurden von Korb *et al.* die empirischen *Scoring*-Funktionen PLP und CHEMPLP beschrieben und implementiert (in ²⁷⁹ und ³⁸⁷ bezeichnet als $\text{PLANTS}_{\text{PLP}}$ bzw. $\text{PLANTS}_{\text{CHEMPLP}}$). PLP steht für *piecewise linear potential* (engl. stückweise lineares Potenzial, siehe Abbildung 18) und bezeichnet eine einfache Bewertungsfunktion für interatomare Distanzen, die intervallweise linear definiert ist. Das PLP wurde bereits früher für die Beschreibung molekularer Wechselwirkungen eingesetzt.^{288,390–392}

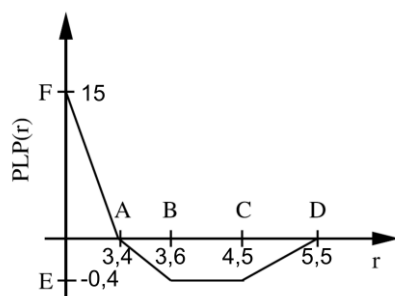


Abbildung 18: PLP-Scoring-Funktion für die abstandsabhängige Bewertung atomarer Interaktionen, nach Korb *et al.*,³⁸⁷. Der interatomare Abstand r wird in Å angegeben. Der angegebene Wert $E=-0,4$ gilt für „sterische“ Wechselwirkungen (entsprechend³⁸⁷). PLP-Score-Werte sind dimensionslos.

Die Linearität in den Intervallen erlaubt schnelle Interpolation. Die konstante Bewertung des Abstandsoptimums mit dem Wert E erlaubt sogar ein einfaches Auslesen des entsprechenden *Score*-Wertes. Das konsequente Ignorieren von Wechselwirkungen mit einem Abstand größer als D begrenzt die Zahl der zu betrachtenden Wechselwirkungen.

Beide *Scoring*-Funktionen, wahlweise PLP oder CHEMPLP, können in PLANTS zur Bewertung der Komplementarität von Rezeptor und Ligand zueinander benutzt werden (zusätzlich besteht die Option, die *Scoring*-Funktion PLP95 zu verwenden, auf die hier nicht eingegangen wird, siehe dazu^{288,386}). PLP und CHEMPLP besitzen ähnliche Charakteristika, liefern aber unterschiedliche Ergebnisse – für einen Vergleich beider im Rahmen von PLANTS sei hier auf Korb *et al.*,²⁷⁹ verwiesen. Die Werte der *Scoring*-Funktionen kombinieren verschiedene Energiebeiträge zu den Wechselwirkungen zwischen Rezeptor und Ligand – positive Werte implizieren daher negative Bindungsaffinität, also Abstoßung zwischen Ligand und Rezeptor. Die verwendeten Funktionen zur Bewertung der verschiedenen Energiebeiträge basieren auf (zum Teil paarweise) atomtypspezifischen Parameterwerten. Das PLANTS-Programm akzeptiert Rezeptor- und Ligandeingaben im TRIPOS-mol2-Format, so dass eine Abbildung von den Atomtypen des TRIPOS-Formats auf die PLANTS-spezifischen Atomtypen notwendig ist (siehe „*Table 1*“ in²⁷⁹).

In Gleichung 5 ist als Beispiel die termweise Aufschlüsselung der CHEMPLP-*Scoring*-Funktion angegeben.

$$f_{PLANTS_{CHEMPLP}} = f_{plp} + f_{hb} + f_{hb-ch} + f_{hb-CHO} + f_{met} + f_{met-coord} + f_{met-ch} + f_{met-coord-ch} + f_{clash} + f_{tors} + c_{site}$$

wobei die Summanden folgende Energiebeiträge bedeuten:

- f_{plp} Summe des PLP für Rezeptor-Ligand-Atompaare mit Fallunterscheidung zwischen
- Wasserstoffbrückenbindung (PLPpartbond)
 - Metallionchelatierung (PLPpartmetal)
 - unpolar-unpolare Rezeptor-Ligand-Kontakte (PLPpartsteric)
 - polar-unpolare Rezeptor-Ligand-Kontakte (PLPpartburpolar)
- und eines repulsiven stückweise linearen Potentials (PLPpartrepulsive) für
- Donor-Donor-Kontakte
 - Akzeptor-Akzeptor-Kontakte
 - Donor-Metallion-Kontakte
- f_{hb} , f_{hb-ch} , f_{hb-CHO} distanz- und winkelabhängige Wasserstoffbrückenbindungsterme (basierend auf ChemScore-Potenzialen^{326,393,394}), dabei:
- f_{hb} für Donor-Akzeptor-Atompaare, höchstens ein ist Atom formal geladen
- f_{hb-ch} für Donor-Akzeptor-Atompaare, beide Atome sind formal geladen
- f_{hb-CHO} für Donor-Akzeptor-Atompaare, in denen ein Sauerstoffatom als Akzeptor fungiert und ein Wasserstoffatom Donor ist, das an ein Kohlenstoffatom gebunden ist, das wiederum an ein aromatisches Stickstoffatom, das als Akzeptor klassifiziert wurde, gebunden ist
- f_{met} distanz- und winkelabhängiges Potenzial für Kalzium- und Magnesiumionen
- $f_{met-coord}$ Potenzial basierend auf der Einpassung idealer Koordinationspolyeder für andere Metallionen
- f_{met-ch} bei geladenen Metallionakzeptoratomen: wie f_{met} , nur zusätzlich skaliert
- $f_{met-coord-ch}$ bei geladenen Metallionakzeptoratomen: wie $f_{met-coord}$, nur zusätzlich skaliert
- f_{clash} empirisches Potenzial zwischen Schweratomen eines Moleküls, die mindestens drei Bindungen voneinander entfernt sind
- f_{tors} Torsionswinkelpotenzial für alle rotierbaren Bindungen außer Wasserstoffbrückendonorgruppen (aus dem TRIPOS Kraftfeld³⁹⁵, TRIPOS_TORS (Ligand), TRIPOS_TORS_PROT (Protein))
- c_{site} Strafterm für Lage von Atomen außerhalb der vordefinierten Bindetaschensphäre (ATOMS_OUTSIDE_BINDINGSITE, Wert: +50)

Gleichung 5: Aufschlüsselung der CHEMPLP-Scoring-Funktion aus PLANTS. Eine detaillierte formale Beschreibung und Auflistung der Parameterwerte ist zu finden in Korb et al. 2009,²⁷⁹. Die Bezeichnungen „Donor“ und „Akzeptor“ beziehen sich auf Wasserstoffbrückenbindungen. „Atompaare“ umfassen hier jeweils ein Ligand- und ein Rezeptoratom. In Klammern sind programminterne Bezeichner für die entsprechenden Terme angeführt.

3.3.1.5 Parametrisierung von PLANTS

Sowohl die Wahl der Parameter für den ACO-Algorithmus als auch die Parametrisierung der *Scoring*-Funktionen PLP und CHEMPLP wurden von Korb *et al.* detailliert beschrieben.^{279,386,387} Mittels eines hierarchischen Ansatzes wurden die Gewichtungsfaktoren für die einzelnen *Scoring*-Funktions-Beiträge variiert und anschließend evaluiert. Für die Parameterermittlung wurde eine speziell ausgewählte Untermenge des *Astex Diverse Set* (experimentell ermittelte Rezeptor-Ligand-Wechselwirkungsstrukturen, Nissink *et al.*,³⁹⁶) als Trainingsdatensatz verwendet. Nach *Docking*-Versuchen mit verschiedenen Parameterkombinationen wurden die Parameter entsprechend der Auswahl festgelegt, welche zur geringsten strukturellen Abweichung zwischen einem Testdatensatz (Teilmenge des *Astex Diverse Set*) und den durch PLANTS vorhergesagten Wechselwirkungsstrukturen führte. Maß für die Genauigkeit der *Docking*-Ergebnisse ist in dieser Vorgehensweise die (relative) Anzahl vorhergesagter Ligand-Rezeptor-Interaktionsposen, deren RMSD-Wert zur jeweiligen experimentellen Geometrie weniger als 2 Å beträgt.

3.3.2 Versuch und Definitionen zur *Cluster*-Option

Bezüglich der Option zum *Clustering* von Ergebnissen sind zwei Parameter von Bedeutung:

<code>cluster_structures</code>	Anzahl der Ligandkonformationen, die vom <i>Cluster</i> -Algorithmus zurückgegeben werden (Standardwert: 10)
<code>cluster_rmsd</code>	RMSD-Ähnlichkeitsgrenzwert für den <i>Cluster</i> -Algorithmus (Standardwert: 2.0 Å)

Im weiteren Vorgehen dieser Arbeit werden zwei verschiedene Grundsätze im Umgang mit diesen Parametern gepflegt, die hier definiert werden sollen. Es wird unterschieden:

- **Unabhängiges *Docking*:** Mehrfacher Start des PLANTS-Programms, wobei die Ergebnisposen komplett unabhängig voneinander erzeugt werden und die Werte der Parameter `cluster_rmsd` und `cluster_structures` keine Rolle spielen.
- **Diversitätsorientiertes *Docking*:** Erzeugung mehrerer Posen in Abhängigkeit voneinander (`cluster_structures` > 1) entsprechend des *Cluster*-Algorithmus, wobei zwischen allen Ergebnisposen paarweise ein struktureller Unterschied mindestens des (positiven) Wertes des `cluster_rmsd`-Parameters gefordert ist.

Ein Sonderfall des diversitätsorientierten *Docking* soll **erschöpfendes *Docking*** sein. Dies beschreibt die Anforderung einer extrem hohen Anzahl von Posen, so dass der Algorithmus vor Erreichen der geforderten Posenzahl terminiert. Kriterium hierfür ist ein unterschrittener Grenzwert für die Verbesserung des *Score* während der Simplex-Optimierung.²⁷⁹

Um zu untersuchen, ob die *Cluster*-Einstellungen von PLANTS signifikanten Einfluss auf die *Docking*-Ergebnisse haben können, wurde in einem Versuch veranschaulicht, wie stark sich die Ergebnisposenmengen zwischen den Einstellungen `cluster_structures` = 1 mit 30-fachem *Docking* und `cluster_structures` = 30 unterscheiden. Dazu wurde das Terpinylkation (siehe A 26, Seite 225) mit PLANTS in die Substratbindetasche der Röntgenkristallstruktur der Limonensynthase aus *M. spicata* gedockt.

3.3.3 Modifikation der CHEMPLP-*Scoring*-Funktion

In der Berechnung des *Score*-Wertes für die CHEMPLP-*Scoring*-Funktion (siehe Gleichung 5, Seite 42) findet eine Fallunterscheidung statt, welche die verschiedenen Möglichkeiten

von interatomaren Rezeptor-Ligand-Wechselwirkungen berücksichtigt. Entsprechend der Paarung von Atomtypen und der Distanz zwischen beiden Atomen wird der CHEMPLP-Score zugewiesen. Diese Fallunterscheidung wurde um zwei Fälle erweitert, um Wechselwirkungen zwischen karbokationischen Atomen des Liganden und

- aromatischen Atomen einer Aminosäureseitenkette im Rezeptor mit E_{aro} bzw.
- Schwefelatomen von Methioninseitenketten im Rezeptor mit E_{met}

speziell zu bewerten. Dazu wurde von Michael Dressel in der Arbeitsgruppe Computerchemie am Leibniz-Institut für Pflanzenbiochemie in Halle (Saale) eine Routine im PLANTS-Quellcode programmiert, die alle relevanten Atome der Bindetasche und des Liganden auf die jeweilige Eigenschaft prüft und ggf. eine entsprechende Markierung zuweist. Diese Atomtypmarkierung wird zur Fallunterscheidung in der Potenzialberechnung verwendet, in der statt des jeweils originalen Parameterwertes $E = 0,4$ für diese Wechselwirkungen die neuen Werte E_{aro} und E_{met} eingeführt werden. Für die effiziente Ausführung dieser Änderung verfasste der Autor dieser Arbeit das Skript `transformer` (siehe A 11, Seite 158), welches die entsprechenden Stellen im Quellcode mit jeweils einem neuen Wert ersetzt, anschließend den Quellcode kompiliert und somit eine ausführbare Version von PLANTS erzeugt, welche die veränderten Parameterwerte im Namen trägt:

PLANTS_ E_{aro} - E_{met}

Das Skript `transformer` dient als Hauptinstanz zum Aufruf anderer Routinen, welche das *Docking* mit den neuen CHEMPLP-Parametern durchführt (A 12, Seite 162) und dabei auch Parameter entsprechend an die untergeordneten Instanzen übergibt. Zusätzlich zu den Größen in Tabelle 7 muss der Benutzer die absoluten Pfade zu den .mol2-Dateien von Ligand und Rezeptor angeben.

Tabelle 7: Eingabevariablen für das Hauptskript `transformer`, welche in den *Docking*-Untersuchungen verwendet wurden und durch den Benutzer übergeben werden

Bezeichner	Bedeutung
E_{aro}	Konkreter Wert für E im CHEMPLP für aromatisch-kationische Wechselwirkungen zwischen Kohlenstoffatomen
E_{met}	Konkreter Wert für E im CHEMPLP für Wechselwirkungen zwischen Schwefelatomen von Methioninseitenketten und Karbokation-Atomen
A	Menge der unterschiedlichen Werte für E im CHEMPLP für aromatisch-kationische Wechselwirkungen zwischen Kohlenstoffatomen
M	Menge der unterschiedlichen Werte für E im CHEMPLP für Wechselwirkungen zwischen Schwefelatomen von Methioninseitenketten und Karbokation-Atomen
N	Anzahl der auszugebenden <i>Docking</i> -Posen
\vec{P}	Mittelpunkt der Bindetasche, als Tripel (x, y, z) in Rezeptorkoordinaten
r	Radius der Bindetasche in Å
A_{flex}	Menge der Aminosäuren, die beim <i>Docking</i> flexibel sein sollen, als Aufzählung von Aminosäure und Sequenzposition, zum Beispiel: ALA315, GLU501
d_{clust}	RMSD-Wert für den <i>Cluster</i> -Algorithmus (in Å), Angabe ist fakultativ: wird dieser Parameter gesetzt, so wird diversitätsorientiert gedockt; entspricht der PLANTS-Eingabe <code>cluster_rmsd</code>

3.3.4 Informationsfluss

Die folgende Tabelle 8 listet die verwendeten Skripte und die von ihnen erzeugten Ausgabe-dateien auf, wobei kursiv gedruckte Ausdrücke Bezeichner sind, die in den unterschiedlichen Instanzen variieren (siehe Tabelle 9, Seite 50).

Tabelle 8: Verwendete Bash-Shell-Skripte und die jeweils erzeugten Dateien. In Spalte „#“ ist die Anzahl erzeugter Dateien eingetragen. In der Regel wurden diese jeweils im png- und im pdf-Format erstellt. Ihre Anzahl multipliziert sich entsprechend, falls mehrere Analysen durchgeführt wurden mit Ausnahme der Dateien, die aus `PPQdepict` resultieren. Für den Schlüssel zu den kursiv gedruckten Bezeichnern siehe Tabelle 7 und Tabelle 9, Seite 50.

Skript, Verweis	Namen erzeugter Dateien	#
Beschreibung des Dateiinhalts		
metaPLANTS A 13, Seite 165	<code>scores_sum_top N_{top}</code>	1
	<i>Boxplot</i> der <i>Score</i> -Verteilungen für alle benutzten Potenziale nebeneinander, mit blauer Linie bei jeweiligem Mittelwert	
	<code>$W_{lig-rec_sum_top} N_{top}$</code>	$ W $
	<i>Boxplot</i> der Distanzverteilungen für alle benutzten Potenziale nebeneinander, mit blauer Linie bei jeweiligem Mittelwert	
	<code>Rstats_top N_{top}.out</code>	1
	Potenzialweise: Aufstellung der statistischen Beschreiber aus <i>Boxplots</i> inkl. Mittelwert, 95%-Konfidenzintervall und Ausreißeranzahl – zuerst für <i>Scores</i> , anschließend für jede gemessene Distanz	
	<code>GRAPH/scores_ $E_{spec_top} N_{top}$</code>	$ A + M $
	Für einen Wert eines der beiden Parameter nebeneinander <i>Boxplots</i> der <i>Scores</i> für alle Werte des anderen Parameters, mit blauer Linie bei jeweiligem Mittelwert	
	<code>GRAPH/dist_ $W_{lig-rec_} E_{spec_} top N_{top}$</code>	$ A + M $
Für einen Wert eines der beiden Parameter nebeneinander <i>Boxplots</i> der gemessenen Distanzen für alle Werte des anderen Parameters, mit blauer Linie bei jeweiligem Mittelwert		
corDistScores A 15, Seite 176	<code>GRAPH/cor_ $W_{lig-rec_} E_{spec_} top N_{top}$</code>	$ W \cdot (A + M)$
	Auftragung von <i>Scores</i> gegen gemessene Abstände einer Wechselwirkung für alle Varianten eines Parameters bei konstantem anderen Parameter, inkl. Regressionsgeraden und Pearson-Korrelationskoeffizienten	
	<code>cor_ $W_{lig-rec_} all_ top N_{top}$</code>	$ W $

Skript, Verweis	Namen erzeugter Dateien	#
	Beschreibung des Dateiinhalts	
	Farbliche Kodierung der Pearson-Korrelationskoeffizienten zwischen Scores und gemessenen Distanzen für alle Kombinationen aus den Schritten beider Parameter, Eintragung des Zahlenwertes für jedes verwendete Potenzial	
violinPLANTS A 16, Seite 180	scores_violin_top N_{top}	1
	$ A \times M $ Matrix mit Violinen-Darstellung der Scores für alle verwendeten Potenziale, mit <i>Boxplot</i> und blauer Linie bei jeweiligem Mittelwert	
	GRAPH/scores_violin_only E_{aro} E_{met} top N_{top}	0 bzw. 1
	Nur bei genau einem untersuchten Potenzial: Violinen-Darstellung der Scores, mit <i>Boxplot</i> und blauer Linie bei Mittelwert	
	GRAPH/scores_violin_ E_{spec} top N_{top}	$ A + M $
	Nebeneinander: Violinen-Darstellung der Scores für alle Werte eines Parameters bei konstantem anderen Parameter	
	$W_{lig-rec}$ _violin_sum_top N_{top}	$ W $
	$ A \times M $ Matrix mit Violinen-Darstellung für alle gemessenen Distanzen, jeweils mit <i>Boxplot</i> und blauer Linie bei jeweiligem Mittelwert	
	GRAPH/dist $W_{lig-rec}$ _violin_only E_{aro} E_{met} _top N_{top}	0 bzw. $ W $
	Nur bei genau einem untersuchten Potenzial: Violinen-Darstellung einer gemessenen Distanz, mit <i>Boxplot</i> und blauer Linie bei Mittelwert	
GRAPH/dist $W_{lig-rec}$ _violin_ E_{spec} top N_{top}	$ A + M $	
Nebeneinander: Violinen-Darstellung einer gemessenen Distanz für alle Werte eines Parameters bei konstantem anderen Parameter, jeweils mit <i>Boxplot</i> und blauer Linie bei jeweiligem Mittelwert		
violinOverlay A 17, Seite 185	scores_viOverlay_ E_{spec} top N_{top}	$ A + M $
	Ein Diagramm: Violinen-Darstellung der Scores von grün nach rot für alle Werte eines Parameters bei konstantem anderen Parameter, mit Linie bei jeweiligem Median	
	dist_ $W_{lig-rec}$ _viOverlay_ E_{spec} top N_{top}	$ W \cdot (A + M)$
Ein Diagramm: Violinen-Darstellung einer gemessenen Distanz von grün nach rot für alle Werte eines Parameters bei konstantem anderen Parameter, mit Linie bei jeweiligem Median		

Skript, Verweis	Namen erzeugter Dateien	#
	Beschreibung des Dateiinhalts	
	$GRAPH/comp_W_{lie-rec}_W'_{lie-rec}_vio_E_{spec_top}$ N_{top}	$(A + M) \sum_{i=1}^{W-1} i$
	Ein Diagramm: Violinen-Darstellung zweier gemessener Distanzen in Grün- bzw. Rottönen für alle Werte eines Parameters bei konstantem anderen Parameter, mit Linie bei jeweiligem Median	
violinTops A 18, Seite 190	$GRAPH/scores_tops_E_{aro}_E_{met}$	$ A \cdot M $
	Ein Diagramm: Violinen-Darstellung der Scores von rot nach grün für zehn aufeinanderfolgende Intervalle von jeweils zusätzlich 10% der Top-Posen für ein Potenzial, mit Linie bei jeweiligem Median	
	$GRAPH/tops_W_{lig-rec}_E_{aro}_E_{met}$	$ W \cdot A \cdot M $
	Ein Diagramm: Violinen-Darstellung einer gemessenen Distanz von rot nach grün für aufeinanderfolgende Intervalle von jeweils zusätzlich 10% der Top-Posen für ein Potenzial, mit Linie bei jeweiligem Median	
drawPLPs A 19, Seite 192	usedPLPs	1
	Ein Diagramm: Auftragung der benutzten Potenziale, mit Originalpotenzial in schwarz, alle anderen Potenziale farbig	
partPLANTS A 20, Seite 195	$DOCKED/scoreParts_top N_{top}$	$ A \cdot M $
	Ungewichtete Auftragung aller Score-Anteile mit positiver Standardabweichung über alle betrachteten Docking-Posen; in der Legende Korrelationskoeffizienten zwischen Gesamt-Score und dem jeweiligen Score-Anteil	
	$partCor_W_{lig-rec}_top N_{top}$	$ W $
	<i>Heatmap</i> -Darstellung pro betrachteter Wechselwirkung: Mit Rot-Grün-Verlauf kodierte Korrelationskoeffizienten zwischen Score-Anteilen und gemessenen Distanzen	
	$partCor_E_{spec}_E'_{spec_top} N_{top}$	$ A \cdot M $
	<i>Heatmap</i> -Darstellung pro verwendetem Potenzial: Mit Rot-grün-Verlauf kodierte und zusätzlich beschriftete Korrelationskoeffizienten zwischen Score-Anteilen und gemessenen Distanzen	
calcDistances A 22, Seite 200	$distances/E_{aro}_E_{met_dist_C_C'}.txt$	$ A \cdot M \cdot N$
	Für das jeweilige Potenzial: eine Spalte mit Distanzen entsprechend der beiden geometrischen Beschreiber; Reihenfolge entsprechend der PLANTS-Ausgabenummerierung	
	$distances/super_dist_C_C' .txt$	N

Skript, Verweis	Namen erzeugter Dateien	#
Beschreibung des Dateiinhalts		
	In Spalten für alle benutzten Potenziale: mit Tabulatorzeichen getrennte Distanzen entsprechend der beiden geometrischen Beschreiber; Reihenfolge entsprechend der PLANTS-Ausgabenummerierung; mit Potenzialen in Kopfzeile	
evaTransform_c A 23, Seite 204	graphics/perc_3.4_5.5_E_aro_E_met_top N_top	$ A \cdot M $
Balkendiagramm zum Prozentsatz der <i>Docking</i> -Posen, für den die jeweils gemessenen Distanzen im <i>Score</i> -relevanten Abstandsbereich zwischen 3,4 Å und 5,5 Å liegen (Anmerkung: evaTransform_c ist eine Vorversion von evaTransform_e und erzeugt weitere Dateien, welche den Ausgaben von evaTransform_e entsprechen)		
evaTransform_e A 24, Seite 208	graphics/e_dist_C__C'_E_spec_top N_top	$(A + M) \cdot N$
Für die jeweilige Potenzialreihe und Top-Posenzahl: dreiteilige Darstellung mit Dichte-Diagramm und <i>Boxplot</i> der Distanzen sowie <i>Heatmap</i> -Darstellung der Kolmogorov-Smirnov-p-Werte zur paarweisen Wahrscheinlichkeit zweier empirischen Verteilungen		
graphics/e_scores_E_spec_top N_top		$ A + M $
Für die jeweilige Potenzialreihe und Top-Posenzahl: dreiteilige Darstellung mit Dichte-Diagramm und <i>Boxplot</i> der <i>Scores</i> sowie <i>Heatmap</i> -Darstellung der Kolmogorov-Smirnov-p-Werte zur paarweisen Wahrscheinlichkeit zweier empirischer Verteilungen		
graphics/e_rank_overview_N_top		1
Spearman-Rangkorrelation zwischen gemessenen Abständen (bzw. dem <i>Score</i> -Anteil plpPartSteric) und den jeweiligen Rängen im <i>Scoring</i> des <i>Docking</i> , in Form einer <i>Heatmap</i> , die zeilenweise die gemessenen Distanzen bzw. <i>Scores</i> und spaltenweise die verwendeten Potenziale listet		
graphics/e_relevance_N_top		1
Wie in vorheriger Zelle, aber mit numerischer Beschriftung der <i>Heatmap</i> -Zellen		
graphics/e_rank_values_N_top		1
<i>Heatmap</i> -Darstellung der Prozentsätze an <i>Docking</i> -Posen, bei der die jeweils betrachtete Wechselwirkung im <i>Score</i> -relevanten Abstandsbereich zwischen 3,4 Å und 5,5 Å liegt (zeilenweise die gemessenen Distanzen und spaltenweise die verwendeten Potenziale)		
graphics/e_PLPpartsteric_E_spec_N_top		$ A + M $

Skript, Verweis	Namen erzeugter Dateien	#
Beschreibung des Dateiinhalts		
	Für die jeweilige Potenzialreihe und Top-Posenzahl: dreiteilige Darstellung mit Dichte-Diagramm und <i>Boxplot</i> des jeweiligen <i>Score</i> -Anteils <code>plpPartSteric</code> sowie <i>Heatmap</i> -Darstellung der Kolmogorov-Smirnov-p-Werte zur paarweisen Wahrscheinlichkeit zweier empirischer Verteilungen	
	graphics/e_cor_cor_N_top	1
	<i>Heatmap</i> -Darstellung des Pearson-Korrelationskoeffizienten zwischen der monotonen Reihe der für E_{aro} bzw. E_{met} verwendeten Parameter und den gemessenen Distanzen (bzw. dem <i>Score</i> -Anteil <code>plpPartSteric</code>), zeilenweise die gemessenen Distanzen und spaltenweise die verwendeten Potenzialgruppen	
	graphics/e_cor_cor_values_N_top	1
	Wie in vorheriger Zelle, aber mit numerischer Beschriftung der <i>Heatmap</i> -Zellen	
	graphics/e_cor_C_C'_E_aro-E_met-N_top	W · A · M
	Auftragung der gemessenen Distanzen und des <i>Score</i> -Anteils <code>plpPartSteric</code> über die Ränge der <i>Docking</i> -Posen; Angabe des Spearman-Korrelationskoeffizienten	
PPQdepict A 25, Seite 219	graphics/ppq_E_aro-E_met_d.ph4	$\leq 15 \cdot A \cdot M $
	MOE-lesbare Pharmakophordarstellung eines hierarchischen <i>Clustering</i> der Positionen der Karbokation-Atome und flexiblen Seitenketten aller <i>Docking</i> -Posen des jeweiligen Experimentes mit einer Klassendistanz von d (in Å, siehe Tabelle 9); mit farblicher Markierung der einzelnen Klassen; Ergebnisdateien werden nur erzeugt, wenn weniger als sieben Klassen resultieren; flexible Seitenketten werden entweder durch ein Bezugsatom (Schwefelatom bei Methionin) oder den Mittelpunkt des aromatischen Systems (aromatische Aminosäuren) repräsentiert	
	graphics/ppq_E_aro-E_met_d.txt	$\leq 15 \cdot A \cdot M $
	Textuelle Beschreibung der Darstellung der Datei in der vorherigen Zelle: Zuordnung der verwendeten Farben zu den Klassen und Angabe der Klassenmittelpunkte für jede Repräsentanz	
	graphics/ppq_E_aro-E_met_full.txt	A · M
	MOE-lesbare Pharmakophordarstellung aller Repräsentanzpunkte (Karbokationen, Schwefelatom bei Methionin, Mittelpunkt des aromatischen Systems bei aromatischen Aminosäuren) der <i>Docking</i> -Posen des jeweiligen Experimentes; farbliche Kodierung der Repräsentanz	

Skript, Verweis	Namen erzeugter Dateien	#
	Beschreibung des Dateiinhalts	
	graphics/ppq_ E_{aro} _ E_{met} _full.txt	$ A \cdot M $
	textuelle Beschreibung der Darstellung der Datei wie in der vorherigen Zelle: Zuordnung der verwendeten Farben zu den Repräsentanzen	

Vor dem Start von `partPLANTS` wurde für jedes Experiment mit unabhängigem *Docking* das Skript `collectFeats` (A 21, Seite 198) gestartet, um entsprechende Dateien `features.csv` zusammenzustellen, welche zu jedem *Docking*-Ergebnis die Zusammensetzung des jeweiligen Gesamt-Score beinhalten. Für diversitätsorientiertes *Docking* liefert `PLANTS` die jeweilige Datei automatisch. Einige Skripte erstellen ferner Textdateien mit Zwischenergebnissen, die in Tabelle 8 nicht aufgeführt sind, siehe dazu die Bash-Shell-Skripte im Anhang. In Tabelle 8 aufgeführte Skripte, die nicht direkt oder indirekt von `transformer` aufgerufen werden, wurden nachträglich einzeln gestartet und fordern zum Teil weitere Eingabeparameter für die Auswertung der *Docking*-Experimente. Diese Parameter umfassen die Größen:

- N_{top} Anzahl der in der (jeweiligen) Auswertung zu betrachtenden Top-Posen
- W Menge der zu analysierenden Distanzen

Zur Analyse der umfangreichen Daten wurde ein Dateiformat definiert, welches für einen Aufruf von `transformer` die zu analysierenden Abstände mit regulären Ausdrücken spezifiziert. Die entsprechenden Analysespezifikationen werden von dem Skript `evaPLANTS`, verwendet, um die zu messenden Größen mit Hilfsskripten zu ermitteln und auszuschreiben (siehe A 28, Seite 227). Die konkret verwendeten Analysespezifikationen sind in A 28, Seite 227, angeführt. Für die Messung von Abständen wurden zwei Varianten gewählt. Einerseits werden einfache euklidische Abstände zwischen Atompaaaren berechnet. Andererseits wird für die Wechselwirkungen eines karbokationischen Atoms eines Liganden zu aromatischen Aminosäureseitenketten des Rezeptors der Abstand zwischen kationischem Atom und Aromatenzentrum kalkuliert. Als Zentrum eines Aromaten dient hierbei der geometrische Mittelpunkt der aromatischen Atome – diese müssen vom Benutzer in der Analysedatei angegeben werden.

Tabelle 9: Bezeichner für die Bildung von in Tabelle 8 angegebenen Dateinamen

Platzhalter	Bedeutung
$ A $, $ M $ bzw. $ W $	Kardinalität der Menge A , M bzw. W
$W_{lig-rec}$	Wechselwirkungskürzel der Form "lig_LIGATOM_rec_RECATOM" bzw. "lig_LIGATOM_center_ANAFILLINE", wobei: LIGATOM die Kennzahl des entsprechenden Ligandenatoms ist, RECATOM die Kennzahl des entsprechenden Rezeptoratoms ist und ANAFILLINE die Zeilennummer der entsprechenden Analysedatei ist, in welcher die Rezeptoratomkennzahlen gelistet sind, zu deren Zentrum der Abstand gemessen wurde (Aromatenzentrum-Definition)
E_{spec}	"aro_ E_{aro} " bzw. "met_ E_{met} "
GRAPH	„pdf“ bzw. „png“ – Ordner, in denen die entsprechenden Grafiken

Platzhalter	Bedeutung
	abgelegt werden, relativ zum Hauptverzeichnis des jeweiligen <i>Docking</i> -Experiments
<i>C</i> bzw. <i>C'</i>	Jeweils textuelle Beschreibung für bestimmte Atome oder das arithmetische Mittel zwischen bezeichneten Atomen, siehe hierzu auch die verwendeten Analysedateien (A 28, Seite 227)
<i>DOCKED</i>	Verzeichniskürzel der Form „docked_ <i>E</i> _{aro} _ <i>E</i> _{met} “: Verzeichnisse mit den <i>Docking</i> -Ergebnissen der einzelnen Potenziale
<i>d</i>	Distanz (in Å), die für hierarchisches <i>Clustering</i> in der Auswertung verwendet wird, um Klassen voneinander zu trennen. Für <i>d</i> werden natürliche Zahlen zwischen inklusive 1 und 15 gewählt. Zur Wahrung der Übersichtlichkeit werden Ergebnisdateien nur erzeugt, wenn die resultierende Klassenzahl kleiner als sieben ist.

3.3.4.1 Informationsverwaltung

Die Zuordnung von Daten zu Experimenten folgt einer hierarchischen Ordnerstruktur. Die einzelnen Experimente besitzen Kennzahlen im Datums-/Uhrzeitformat, die an zusätzliche Auswerteskripte übergeben werden können und die zu analysierenden *Docking*-Ergebnisse eindeutig identifizieren. Unterordner entsprechend Tabelle 9 wurden benutzt, um Experimente zu trennen und eine hierarchische Struktur der mit den *in silico*-Experimenten verknüpften Daten zu erzeugen.

3.3.4.2 Verwendete Parameterkombinationen

Es wurden Parameterkombinationen entsprechend Tabelle 10 verwendet.

Tabelle 10: Parameterkombinationen für die *Docking*-Versuche in die Modelle von *Mentha spicata* Limonensynthese („L“ in Spalte R, R für Rezeptor) und *Solanum habrochaites* Santalen- und Bergamotensynthese („S“ in Spalte R). Spalte K beinhaltet die Übersetzung der Datums-Identifikatoren in Buchstaben, welche im Rahmen dieser Arbeit zur einfacheren Identifikation der Experimente verwendet werden (K für Kodierung).

Identifikator (ID)	K	R	<i>N</i>	<i>d</i> _{clust}	Menge A	Menge M
20110804_15_17	A	L	100	2	-0,4; -2; -4; -6; -8; -10	-0,4; -2; -4; -6; -8; -10
20110805_13_25	B	L	100	2	-20; -30	-0,4; -20; -30
20110809_11_43	C	S	1.000	2	0; -0,4; -0,6; -0,8; -1; -5	0; -0,4; -0,6; -0,8; -1; -5
20110812_15_46	D	S	100.000	1	-0,4; -0,5; -0,6	-0,4; -0,5; -0,6
20110817_09_20	E	L	10.000	1	-0,4; -0,6; -0,8; -1	-0,4; -0,6; -0,8; -1
20110818_08_28	F	S	100	-	-0,4, -0,6, -0,8, -1	-0,4
20110818_12_38	H	S	10.000	1	-0,4; -0,45; -0,5; -0,55;	-0,4; -0,45; -0,5; -0,55;
20110905_11_56	J				-0,6	-0,6
20110810_08_19	K					
20110819_09_53	L	S	10	-	-0,4	-0,4
20110819_10_34	M					
20110831_09_32	N	S	1.000	1	-0,4	-0,4
20110902_09_59	O					
20110902_09_03	P	S	1.000	1	-0,399; -0,4; -0,401	-0,399; -0,4; -0,401
20110902_14_38	Q					
20110929_15_37	R	S	100	-	-0,4; -0,8; -1,2; -1,6; -2	-0,4; -0,8; -1,2; -1,6; -2
20110929_15_38	S					

Die Mengen A und M der Parameterwerte beinhalten hierbei in der Regel den Wert -0,4, welcher im unmodifizierten PLANTS-CHEMPLP für die untersuchten Wechselwirkungen zwischen Karbokationen und Aromaten bzw. Methionin-Schwefelatomen verwendet wird. Für das *Docking* in ShSBS wurden die Aminosäuren W746, M763, Y495 und Y519 flexibel gehalten und für die *Mentha spicata* Limonensynthase die Aminosäuren W324 und M458. Die Wahl der Parameterkombinationen erfolgte sukzessiv aufbauend auf jeweils früheren Ergebnissen (über vergebene Identifikatoren). Während zunächst nach signifikanten Unterschieden zwischen verwendeten Parameterwerten für E_{aro} und E_{met} gesucht wurde, wurde später zusätzlich auch der Aspekt „diversitätsorientiert vs. unabhängig“ adressiert. Die Untersuchung D zeigte, dass für $d_{clust} = 1 \text{ \AA}$ für das *Docking* in ShSBS nicht mehr als 10.000 Posen zu erwarten sind, bevor der PLANTS-Algorithmus vorzeitig terminiert. Die Posenzahl 10.000 zu fordern bedeutet demnach bei diversitätsorientiertem *Docking* einen Abbruch des PLANTS-Algorithmus mit einer Posenanzahl, welche bezüglich der Diversitätsforderung – bestimmt durch den Parameter `cluster_rmsd` – maximal ist.

3.3.4.3 Auswertung der *Docking*-Untersuchungen

Die Auswertung erfolgte hierarchisch auf Basis der erstellten graphischen Auftragungen bzw. Berechnungen von Korrelationen wie in Tabelle 8, Seite 45, aufgelistet. Die Pharmakophor-Darstellungen der großen Zahl von *Docking*-Ergebnissen und die entsprechenden *Clusterings* wurden zum Teil visuell hinsichtlich auffälliger Besonderheiten inspiziert.

3.3.5 Vorbereitung von Ligand- und Rezeptorstrukturen

Die jeweils zu dockenden Ligandstrukturen wurden manuell aufbereitet, um die Karbokation-Beschreibung widerzuspiegeln und eine entsprechende Erkennung durch den modifizierten PLANTS-Algorithmus zu gewährleisten (siehe A 26, Seite 225 sowie A 27, Seite 226). Dazu wurden die entsprechenden Atome in den Liganddateien als C.cat entsprechend der TRIPOS-Atomtypisierung markiert. Anschließend wurde mit der PM3-Methode in MOE jeweils eine Geometrieoptimierung unter Angabe der kationischen Ladung durchgeführt (jeweils `CHARGE=1` als Option). Abbildung 19 ist eine Veranschaulichung der verwendeten Liganddefinitionen.

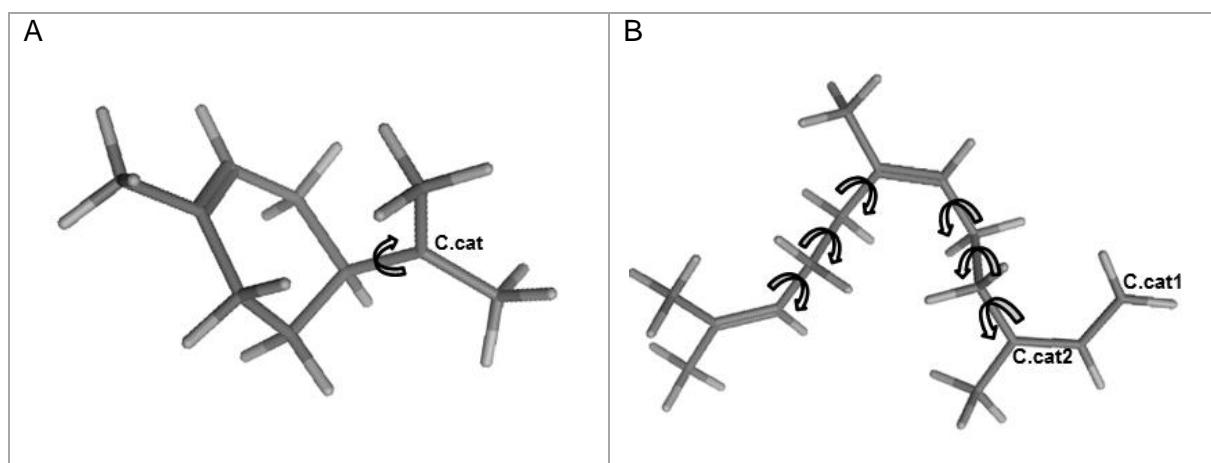


Abbildung 19: Für die *Docking*-Untersuchungen verwendete Kationen: A) α -Terpinylkation und B) (Z,Z)-Farnesylylkation. Die als formal geladen und C.cat definierten Atome sind gekennzeichnet und die Pfeile geben die frei drehbaren Bindungen an, die keine Methylgruppen binden.

Die für das *Docking* verwendeten Enzymstrukturen beinhalten in der Substratbindetasche jeweils den vollständig deprotonierten Diphosphatrest mit den drei komplexierenden Metallionen (Limonensynthase: Mangan, ShSBS: Magnesium). Für die Limonensynthase wurde hierbei das Röntgenkristallstrukturmodell mit dem PDB-Identifikator 2ONH (Hyatt *et al.*,¹²) verwendet, wobei der 2-Fluorolinalylrest in MOE manuell gelöscht wurde (siehe Abbildung 20). Für die ShSBS wurde auf Grundlage des erstellten Enzym-Substratkomplexes (siehe 3.1.3.3, Seite 35) analog verfahren (siehe Abbildung 21).

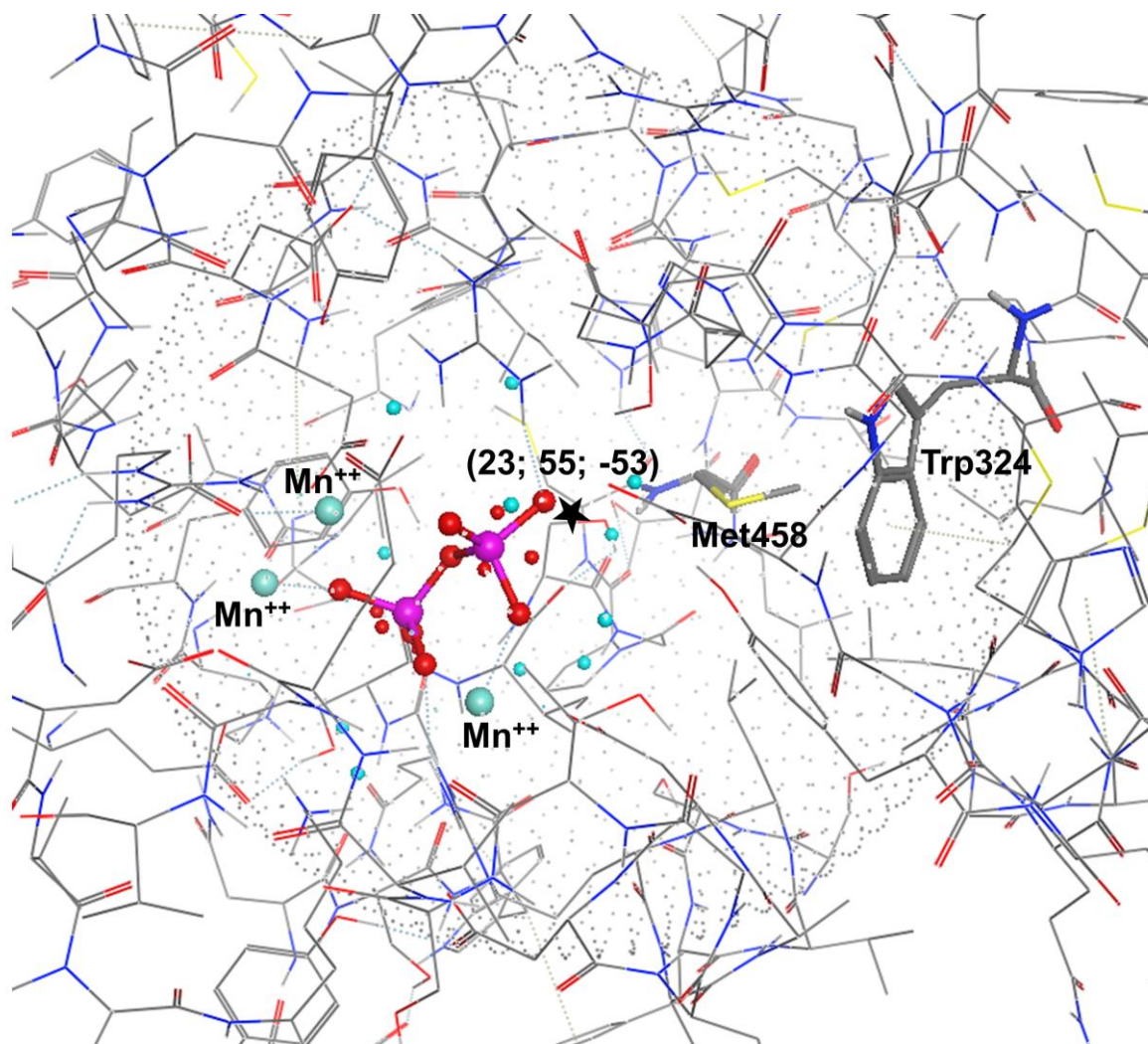


Abbildung 20: Bindetasche der Limonensynthase mit den im *Docking* flexiblen Aminosäuren Met458 und Trp324 und dem belassenen Diphosphatrest. Der schwarze Stern markiert den Mittelpunkt der Bindetaschensphäre (mit Koordinaten), die beim *Docking* verwendet wurde (Radius 10 Å, graue Punkte). Die isolierten cyanfarbenen und roten Kugeln sind Pharmakophorpunkte des Diphosphatrestes.

Abbildung 20 und Abbildung 21 vermitteln einen Eindruck von der Größe und räumlichen Struktur der Bindetasche von Terpensynthasen. Ferner verdeutlichen die Darstellungen, dass 10 Å ein hinreichend großer Wert für den Bindetaschenradius ist, der an das *Docking*-Programm übergeben wird.

Die α -Terpinylkationstruktur wurde bei den hier geschilderten Untersuchungen immer in die aufbereitete Struktur der Limonensynthase und das (*Z,Z*)-Farnesylkation in das entsprechende ShSBS-Modell gedockt.

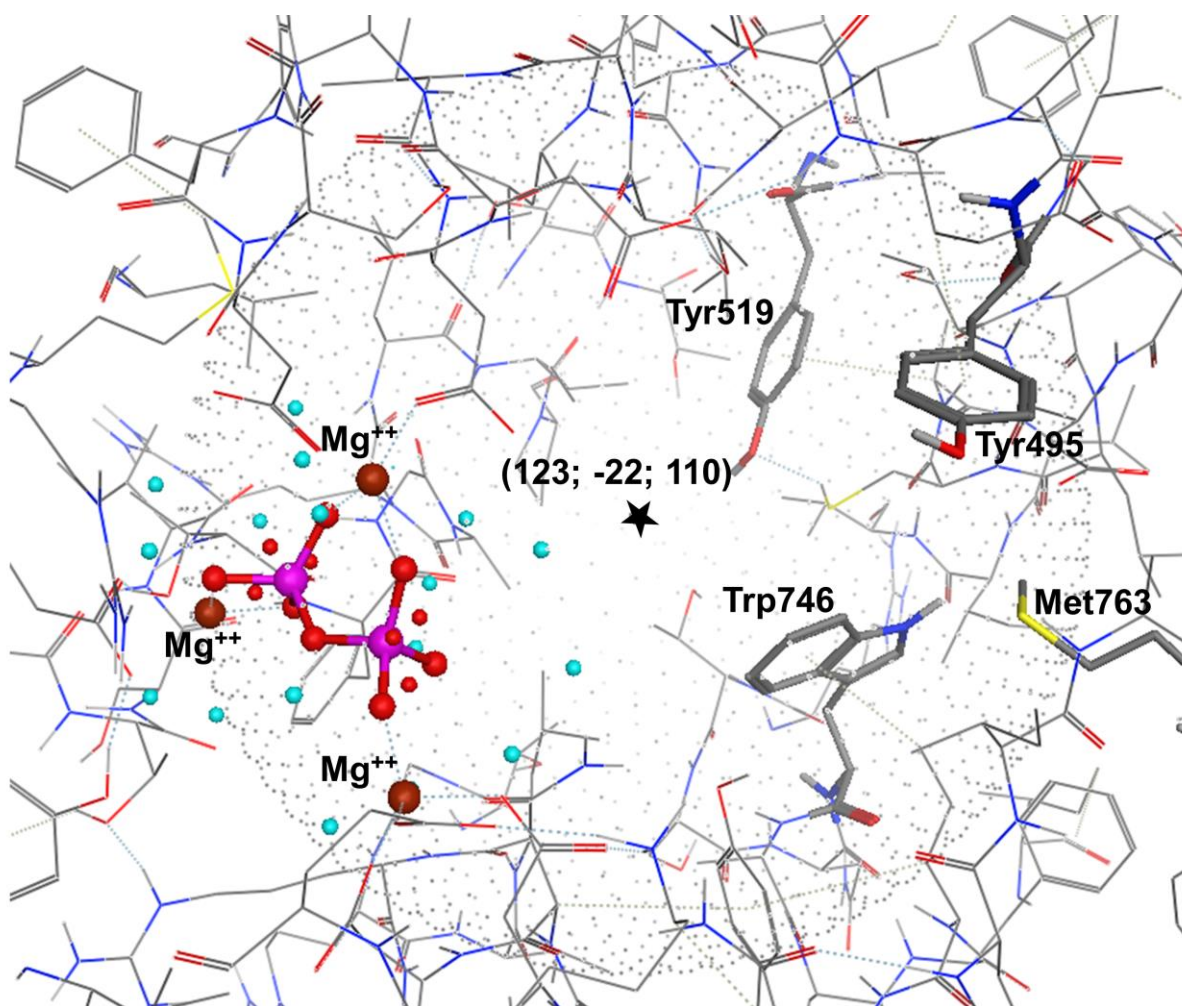


Abbildung 21: Bindetasche der ShSBS analog Abbildung 20 mit den im *Docking* flexiblen Aminosäuren Tyr495, Tyr519, Met763 und Trp746 und dem Diphosphatrest. Der schwarze Stern markiert den Mittelpunkt der Bindetaschensphäre (mit Koordinaten), die beim *Docking* verwendet wurde (Radius 10 Å, graue Punkte). Die isolierten cyanfarbenen und roten Kugeln sind Pharmakophorpunkte des Diphosphatrestes.

3.3.6 Das Geranylkation in verschiedenen Kraftfeldern

Zur Untersuchung der generellen Fähigkeit von Kraftfeldern ein allylisches Karbokation zu modellieren, wurde mit MOE²⁸⁶ das Geranylkation manuell konstruiert und anschließend mit den Kraftfeldern „Rule“ (regelbasiert), TAFF, PEF95SAC, Engh-Huber, OPLS-AA, CHARMM22, CHARMM27, Amber89/94/99 bzw. MMFF94(s/x) optimiert. Hierzu wurde jeweils die in MOE mit dem jeweiligen Kraftfeld verbundene Partialladungsberechnung verwendet.

3.4 Quantenchemische Berechnungen

3.4.1 Bildung von Santalen und Bergamoten

Auf der Basis veröffentlichter vorgeschlagener Reaktionsmechanismen (ohne Betrachtung der jeweiligen Wechselwirkungen mit dem Enzym) für die Bildung von Bergamotenen und Santalenen aus (*E,E*)-FPP³⁹⁷ und ¹⁰¹, siehe auch ^{6,207,254}) wurden Reaktionsschemata für die Bildung von Bergamoten- und Santalen-Isomeren aus (*Z,Z*)-FPP entworfen. Mit dem Programm Maestro³⁹⁸ wurden Jaguar-Rechnungen aufgesetzt, um freie Enthalpien zwischen Intermediaten der Reaktionen zu berechnen. Unter Verwendung der Dichtefunktionaltheorie

wurden flexible Koordinaten-Scans und Optimierungen durchgeführt. Dabei kam der Basissatz 6-31g+ in der B3LYP-Methode zum Einsatz. Die Koordinaten-Scans wurden auf Basis der erstellten Reaktionsschemata definiert und umfassen die Bedingung, dass die jeweils zu bildende Bindung durch schrittweise Annäherung der beiden beteiligten Atome zu Stande kommt. Um dies zu erreichen, wurde von der Ausgangsstruktur die Distanz der beiden Atome in wenigen Schritten (fünf bis elf) forciert auf 1,5 Å verringert. Auf diese Weise erzeugte intermediäre Strukturen wurden anschließend mit derselben Methode optimiert. Für die finalen Deprotonierungsreaktionen wurde ein Essigsäureanion manuell positioniert und anschließend eine Optimierung durchgeführt.

3.4.2 Wechselwirkung zwischen Karbokationen und Aromaten bzw. Ethylmethylsulfid

Zur Berechnung von Wechselwirkungsenergien zwischen Karbokationen und aromatischen Aminosäureseitenketten sowie Ethylmethylsulfid kamen weitere flexible Koordinaten-Scans zum Einsatz, die in Abbildung 22 schematisch dargestellt sind. Diese wurden mit der Hartree-Fock-Methode und dem Basissatz 6-31g in Jaguar (über Maestro) berechnet. Die verwendeten Ausgangskonformationen wurden mit Maestro manuell konstruiert.

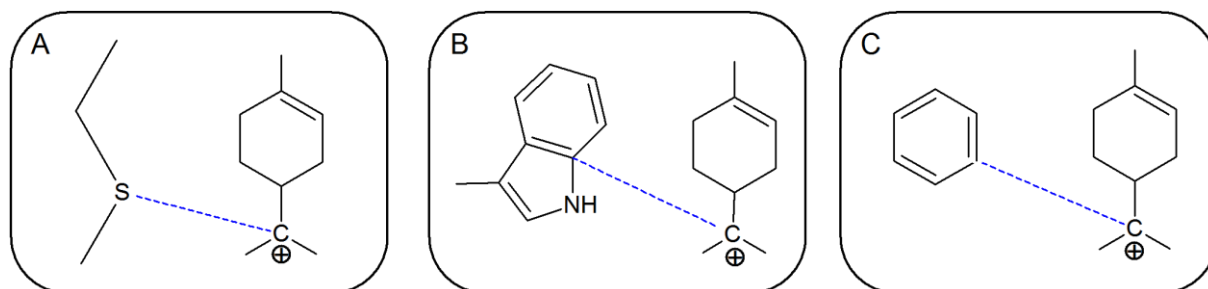


Abbildung 22: Flexible Koordinaten-Scans zwischen α -Terpinylkation und Modellen für Aminosäureseitenketten. Die gestrichelte blaue Linie bezeichnet die *Scan*-Koordinate, wobei jeweils die Distanz von 6 Å in 19 Schritten auf 1,5 Å verringert wurde. A) Ethylmethylsulfid als Modell für Methionin; B) 3-Methylindol als Modell für Tryptophan; C) Benzen als Modell für Phenylalanin

Wechselwirkungsenergien wurden näherungsweise als die Differenz zwischen den quantenmechanischen Energien bei großer Distanz und der entsprechenden Optimumdistanz berechnet. Um die Stabilisierung des α -Terpinylkations zwischen Tryptophan und Methionin zu untersuchen, wurde eine manuell erstellte Szenerie (siehe Abbildung 23) geometrieoptimiert. Dazu wurde ebenfalls Jaguar mit der Hartree-Fock-Methode und dem Basissatz 6-31g verwendet.

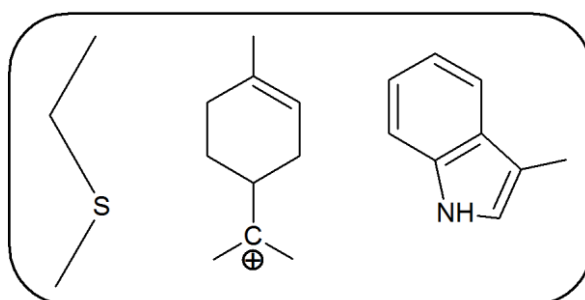


Abbildung 23: Schema einer Wechselwirkungsgeometrie zwischen α -Terpinylkation und zwei nukleophilen Gruppen

4 ERGEBNISSE UND DISKUSSION

Dieses Kapitel geht auf die Ergebnisse der durchgeführten Untersuchungen ein und setzt sie in Relation zu experimentellen Befunden, Fachliteratur und den formulierten Zielstellungen. Ausgewählte Aspekte werden dabei direkt am Ergebnis diskutiert. Eine abschließende Diskussion ist am Ende jedes Abschnitts zu finden.

4.1 Isopentenyltransferasen aus *Arabidopsis thaliana*

4.1.1 Homologiemodellierung

Die hohe Sequenzhomologie der modellierten Isoenzyme untereinander lässt in den meisten Fällen eine allgemeine Diskussion am Beispiel zu, so dass nicht jeweils einzelne Abbildungen gezeigt werden. Auffällige Unterschiede werden jedoch konkret behandelt und mit Modellabbildungen untermauert.

4.1.1.1 *Template*-Wahl aus Ergebnissen der Voruntersuchungen

Aus der durchgeführten Homologiemodellierung mit MOE resultiert Tabelle 11.

Tabelle 11: Homologiemodellierungs-*Templates* für AtIPT. Bei fehlenden Literaturangaben wurde jeweils nur die Kristallstruktur in der PDB veröffentlicht.

PDB-Code	Organismus/Enzymname, Literaturangabe
2QGN	<i>Bacillus halodurans</i> Isopentenyl:tRNA-transferase
1KNQ	<i>Escherichia coli</i> Glukonatkinase, ³⁹⁹
3D3Q	<i>Staphylococcus epidermidis</i> Isopentenyl:tRNA- $\delta(2)$ -Transferase
2YT5	<i>Mus musculus</i> PHD-Domäne des Metal-response element-binding transcription factor 2, ⁴⁰⁰
2JAQ	<i>Mycoplasma mycoides</i> subsp. <i>mycoides</i> SC Desoxyadenosinkinase, ⁴⁰¹
2ZEG	<i>Homo sapiens</i> Glutaminylzyklase (Mutante E201L), ⁴⁰²

Die Untersuchung der Sequenzidentität zwischen den in Tabelle 11 angeführten *Templates* und den AtIPT-Sequenzen zeigt nur geringe prozentuale Übereinstimmungen (siehe A 2, Seite 145). Die durchgeführte blastp-Suche mit der AtIPT1-Sequenz (siehe Seite 144) liefert unter den 100 höchstbewerteten Annotationen ausschließlich Treffer mit e-Werten unter 10^{-60} , eine Abdeckung der Anfragesequenz zwischen 47 % bis 100 % und genau einer annotierten Sequenz mit verfügbarer Struktur. Die entsprechende Röntgenkristallstruktur (PDB-Code 3A8T) ist annotiert als pflanzliche Isopentenyl:Adenylat-Transferase komplexiert mit ATP.^{55[125]} Mit der Veröffentlichung dieser Röntgenkristallstruktur des Enzyms aus *Humulus lupulus* (Echter Hopfen) ist 2010 eine neue potenzielle *Template*-Struktur verfügbar geworden. Weitere relevante Erkenntnisse für die strukturellen Gegebenheiten der Ligandenbindung bieten ferner die Röntgenkristallstrukturmodelle einer bakteriellen Isopentenyltransferase mit den PDB-Codes 2ZE5, 2ZE6, 2ZE7 und 2ZE8.⁶⁶ Die Strukturen beinhalten unterschiedliche Liganden, die Tabelle 12 anführt.

Tabelle 12: Übersicht über zu *Arabidopsis thaliana* DMAPP:ATP/ADP-Transferasen homologen Proteinstrukturen aus Röntgenkristallstrukturanalysen. Die Proteinstrukturen in den PDB-Einträgen 2ZE5, 2ZE6, 2ZE7 und 2ZE8 besitzen identische Aminosäuresequenzen.

PDB-Code	Organismus/Enzymbezeichnung	Ligand(en)
2ZE5	<i>Agrobacterium tumefaciens</i> Isopentenyltransferase	Adenosinmonophosphat (AMP)
2ZE6	<i>Agrobacterium tumefaciens</i> Isopentenyltransferase	AMP, Dimethylallyl-S-thiolodiphosphat
2ZE7	<i>Agrobacterium tumefaciens</i> Isopentenyltransferase	AMP, Dimethylallyl-S-thiolodiphosphat, Zink-Ion
2ZE8	<i>Agrobacterium tumefaciens</i> Isopentenyltransferase	Diphosphation
3A8T	<i>Humulus lupulus</i> Isopentenyl:Adenylate-Transferase	Adenosin-5-Triphosphat (ATP), Phosphation

Die Isopentenyl:Adenylat-Transferase aus *Humulus lupulus* (HIIPT) hat erheblich höhere Homologie zu den AtIPT, wie aus Abbildung 24 ersichtlich ist.

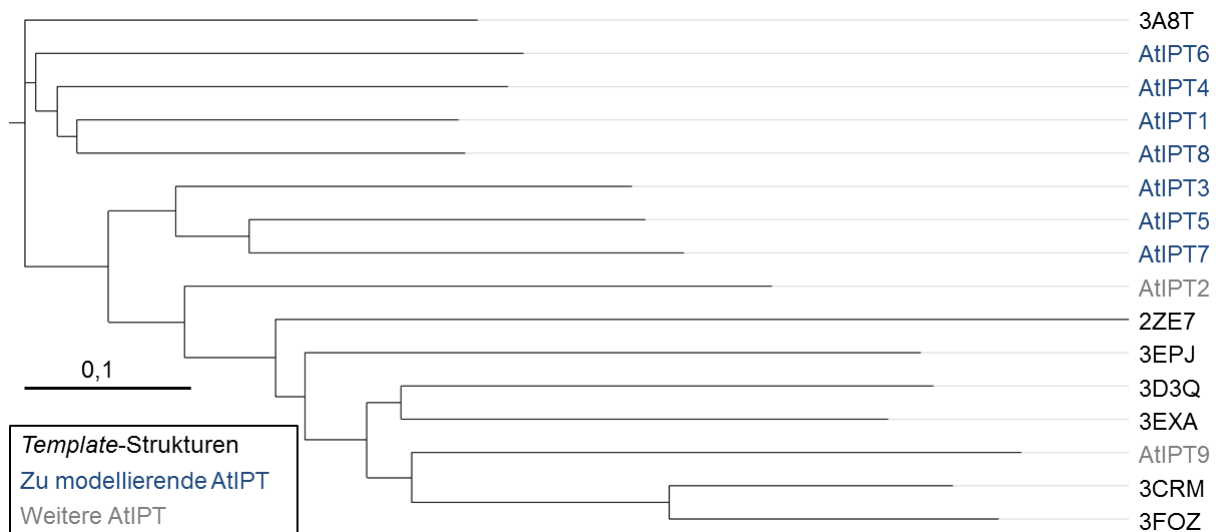


Abbildung 24: Sequenzähnlichkeitsbaum der AtIPT und potenzieller *Template*-Strukturen – vergleiche Tabelle 11 und Tabelle 12 zur Annotation der verwendeten PDB-Codes; *Alignment* und Distanzberechnungen erstellt mit ClustalW, Baumdarstellung mit iTOL.

Abbildung 24 verdeutlicht ferner folgende Aussagen:

- Die beiden Sequenzen von AtIPT2 und AtIPT9 besitzen relativ geringe Homologie zu den restlichen sieben AtIPT-Sequenzen und bilden eine eigene Klade mit allen gefundenen potenziellen Homologiemodellierungs-*Templates* bis auf die Isopentenyl:Adenylat-Transferase aus Hopfen (3A8T).
- AtIPT1 und AtIPT3 bis AtIPT8 sind untereinander sequenzähnlicher als zu anderen aufgeführten Proteinen.
- Die Aminosäuresequenzen zu den PDB-Codes 3A8T und (eingeschränkt) 2ZE7 sind den zu modellierenden Sequenzen von AtIPT1, AtIPT3 bis AtIPT8 am ähnlichsten.

4.1.1.2 Qualität der erstellten Modelle

Die Homologiemodelle der DMAPP:ATP/ADP-Transferasen sind entsprechend der Ausgabe von YASARA von guter Qualität. Dies untermauern die Qualitätsmerkmale in Tabelle 13.

Tabelle 13: YASARA-z-Scores (1.-4. Zeile) und ProSA-web-z-Scores (5. Zeile) für die Bewertung der finalen Homologiemodelle

	AtIPT1	AtIPT3	AtIPT4	AtIPT5	AtIPT6	AtIPT7	AtIPT8
Torsionswinkel	0,501	1,019	0,883	0,923	0,621	0,964	0,764
<i>Packing1D</i>	-0,268	0,224	-0,295	-0,105	-0,588	-0,022	-0,183
<i>Packing3D</i>	-0,387	-0,270	-0,298	-0,208	-0,519	-0,121	-0,367
<i>Insgesamt</i>	-0,212	0,110	-0,125	-0,004	-0,381	0,075	-0,131
z-Score	-8,25	-8,06	-8,37	-8,76	-8,05	-8,61	-8,88

Alle erstellten Modelle sind monomere Proteinstrukturen. Abweichungen in der Tertiärstruktur gibt es untereinander im Wesentlichen in *Loop*-Regionen und im Bereich beider Termini, wie Abbildung 25 veranschaulicht.

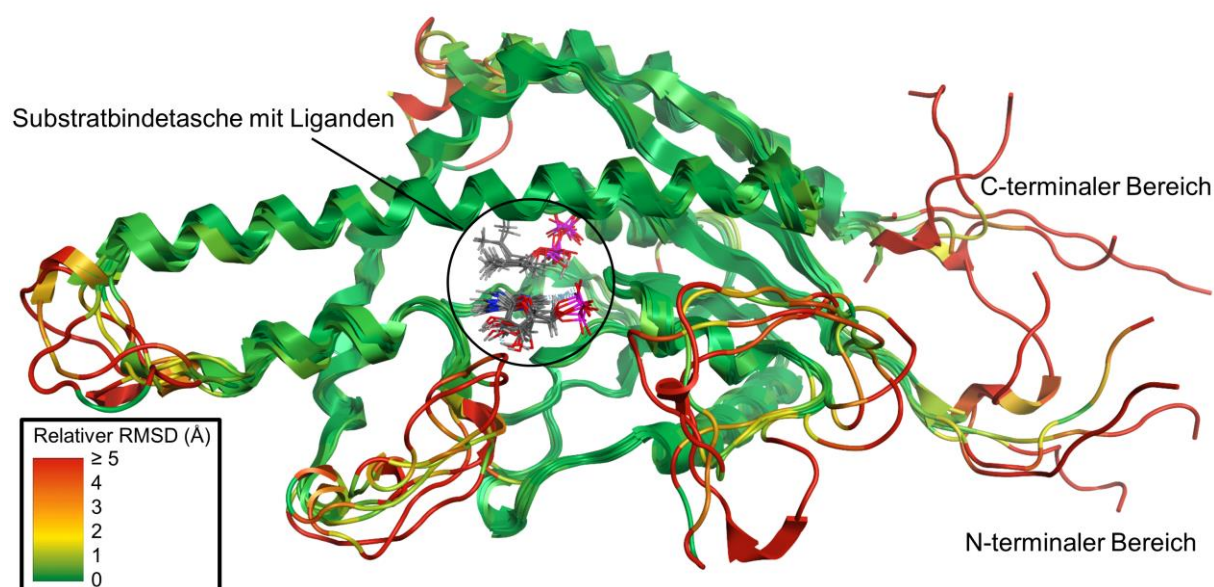


Abbildung 25: Tertiärstrukturdarstellung der strukturell superpositionierten AtIPT-Modelle (AtIPT1, AtIPT3 bis AtIPT8) mit einer Farbkodierung für RMSD-Unterschiede zwischen den Proteinrückgrat-Atomen, Abbildung erstellt unter Benutzung von MOE

Unter A 3 (Seite 145) sind die absoluten Lagen der z-Scores aus ProSA-web bezüglich der zugrunde liegenden Strukturdatenbanken aus Röntgenkristallographie und NMR sowie die jeweilige Auftragung des empirischen Potenzials aufgeführt. Der entsprechende z-Score der *Template*-Struktur der Modelle für AtIPT3 bis AtIPT8 beträgt -9.42 und ist identisch zu dem Wert für das AtIPT1-*Template*.

Aufgrund der identischen *Template*-Struktur für alle IPT außer IPT1 ähneln sich die finalen Homologiemodelle. Auch das Modell für IPT1 zeigt keine starken Abweichungen in der Tertiärstruktur von den restlichen AtIPT-Modellen, wie Tabelle 14 verdeutlicht.

Tabelle 14: RMSD in Å zwischen den AtIPT-Modellen bezüglich der Proteinerückgrat-atome. Angabe des Farbschlüssels in der Spalte ganz rechts; erstellt unter Benutzung von MOE.

	1	3	4	5	6	7	8	
1: IPT1	0.00	1.38	1.38	1.38	1.79	1.66	1.94	4.0
3: IPT3	1.38	0.00	1.53	1.26	1.75	1.28	1.65	3.5
4: IPT4	1.38	1.53	0.00	1.61	1.50	1.37	1.72	3.0
5: IPT5	1.38	1.26	1.61	0.00	1.94	1.54	1.75	2.5
6: IPT6	1.79	1.75	1.50	1.94	0.00	1.45	1.36	2.0
7: IPT7	1.66	1.28	1.37	1.54	1.45	0.00	1.54	1.5
8: IPT8	1.94	1.65	1.72	1.75	1.36	1.54	0.00	1.0
								0.5
								0.0

4.1.1.3 Analyse der Substratbindung

Die Erkennung des Adenylat-Liganden – insbesondere differenzielle Untersuchungen bezüglich Mono-, Di- und Triphosphat – ist für die Zielstellung dieser Arbeit von untergeordneter Bedeutung (eine entsprechende Modellierung wurde von Chu *et al.* an HIPT durchgeführt, siehe ⁵⁵). Hier soll der Schwerpunkt auf die Analyse der Bindung des Prenyl-donormoleküls und des Adenin-Teils von Prenylakzeptoren in den erstellten Homologiemodellen gelegt werden. Vergleichend werden die AtIPT-Homologiemodelle, die HIPT- und die *A. tumefaciens*-IPT-Röntgenkristallstrukturmodelle betrachtet.

Die entsprechend der beschriebenen Selektion (siehe Abschnitt 3.1.3.3, Seite 35) gebildeten *Alignment*-Segmente werden in diesem Abschnitt analysiert. Grundsätzlich zeigen sie für Aminosäuren der Substratbindetasche weitgehende Identität zwischen den Modellen der AtIPT. Die rot markierten *Alignment*-Positionen identifizieren entsprechend Aminosäuren der jeweiligen AtIPT mit ligandnahen Atomen. Diese Positionen sind auch im mit ClustalW berechneten Sequenz-*Alignment* rot hervorgehoben, das im Anhang unter A 5 (Seite 150) zu finden ist. Dieses *Alignment* unterscheidet sich leicht vom aus MOE resultierenden *Alignment* aufgrund von Unterschieden in den Algorithmen. Die Positionsangaben in den Kopfzeilen der folgenden Abbildungen und im Text beziehen sich auf das mit MOE erstellte multiple Sequenz-*Alignment*. Auch die Strukturabbildungen wurden mit MOE erzeugt und zeigen jeweils nur Ausschnitte, wobei unpolare Wasserstoffatome ausgeblendet wurden. Die schematisch dargestellten Interaktionen verfügen über eine Wechselwirkungsenergie von mindestens 8 kJ/mol. Schwächere Wechselwirkungen werden nicht dargestellt.

Die *Alignment*-Positionen 48-54 (siehe Abbildung 26) bilden ein p-Loop – ein über verschiedene Enzymklassen konserviertes Sequenzmotiv zur Erkennung von Phosphatresten.³⁸⁰ Hierbei sind in allen Homologiemodellen den Aminosäuren an den einzelnen *Alignment*-Positionen relativ klar Funktionen zuzuordnen, die mit den Vorschlägen bzw. Strukturmodellen aus der Literatur übereinstimmen.⁵⁵ Es gibt jedoch Unterschiede sowohl zwischen den Homologiemodellen untereinander als auch zur aufgeklärten Struktur des bakteriellen Enzyms.

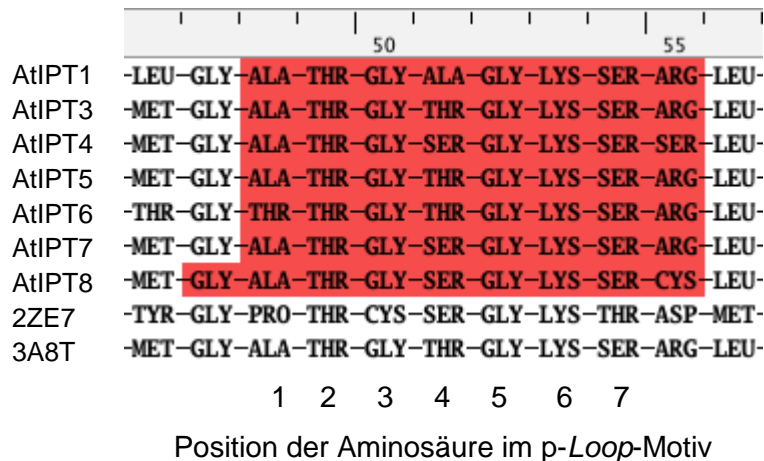


Abbildung 26: Segment A des Sequenz-Alignment der AtIPT und der beiden Template-Strukturen aus *A. tumefaciens* (2ZE7) und *H. lupulus* (3A8T)

Wesentlich fällt auf, dass in den Homologiemodellen für AtIPT3 bis AtIPT8 der α -Phosphatrest protoniert ist, während die Modellierung für AtIPT1 eine vollständig deprotonierte Diphosphatstruktur zeigt. In der veröffentlichten Struktur der *A. tumefaciens*-IPT sind keine Wasserstoffatome aufgelöst. Jedoch interagieren zwei Wassermoleküle mit dem DMASPP-Molekül (Abbildung 28, Seite 62). Es gibt mehrere Unterschiede zwischen den Prenyl-diphosphat-Bindemodi der AtIPT-Modelle, die in Abbildung 27 grafisch veranschaulicht werden. Aspekte, in denen sich die Modelle unterscheiden, sind:

- Ionische Wechselwirkungen von Lys47 mit dem Diphosphatrest: α - vs. β -Phosphat
- Wasserstoffbrücken zwischen Thr49 und dem Brückensauerstoffatom zwischen Prenyl- und Phosphatrest
- Wasserstoffbrücken zwischen den polarisierten Wasserstoffatomen an Rückgrat-Stickstoffatomen der Aminosäuren an den Positionen 50-54

Relativ konsistent ist in den Modellen die Wechselwirkung zwischen Ser54 und dem jeweiligen Metallkation zu beobachten (Zink in AtIPT1, Magnesium in AtIPT3 bis AtIPT8). Die Aminosäure an Position 55 zeigt keine direkten Interaktionen mit Substratmolekülen oder dem metallischen Kofaktor.

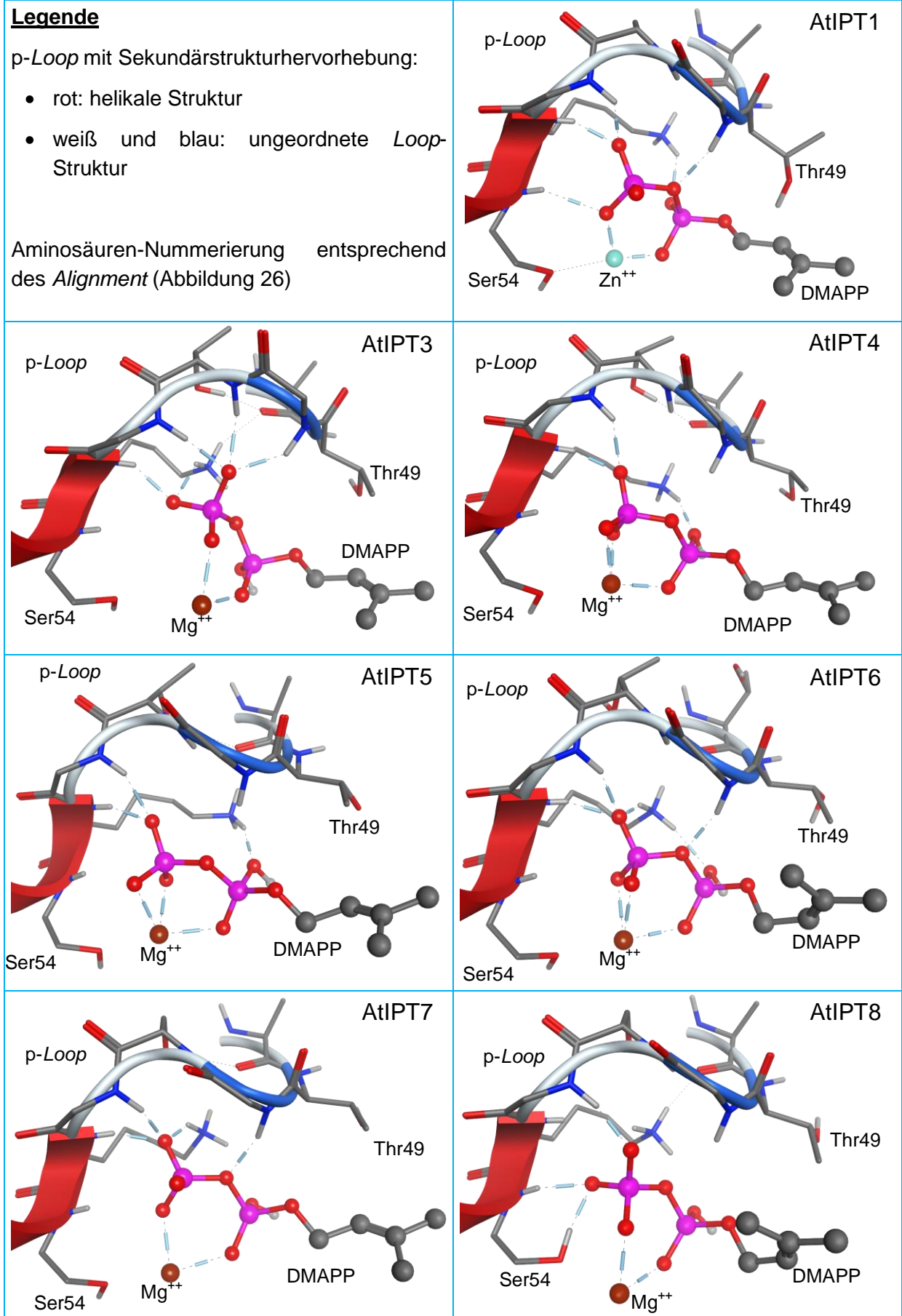


Abbildung 27: Wechselwirkungen des Diphosphatrestes mit dem p-Loop in den AtIPT-Modellen (die dargestellten Aminosäuren entsprechen den Positionen 1-7 des p-Loop)

Um diese Modelle in eine Relation zu Informationen aus der Literatur zu setzen, ist ein Vergleich mit der Bindung des DMASPP-Analogon in der Röntgenkristallstruktur der *A. tumefaciens* IPT (siehe Abbildung 28) sinnvoll.

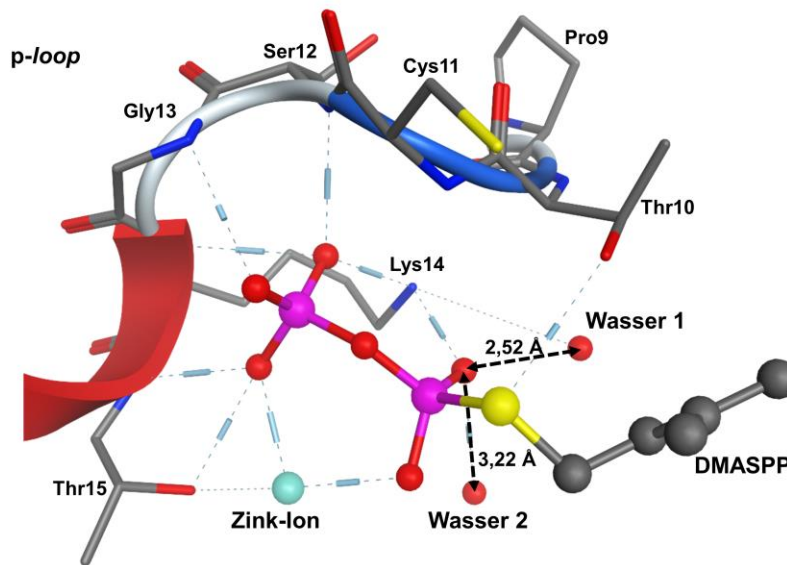


Abbildung 28: DMASPP-Komplexierung in der Kristallstruktur 2ZE7 der Isopentenyltransferase aus *A. tumefaciens* (absolute Nummerierung der Aminosäuren, Perspektive und Sekundärstruktur hervorhebung analog Abbildung 27, ohne Wasserstoffe)

Ein Unterschied zwischen den im Rahmen dieser Arbeit erstellten Modellen und der PDB-Struktur 2ZE7 besteht in der Sequenz des *p-Loop*. Auffällig ist insbesondere das Cystein an dritter Stelle des *p-Loop*-Motivs in 2ZE7, während für die AtIPT-Modelle diese Stelle konsequent durch ein Glycin besetzt ist (vgl. Abbildung 26). In der Struktur der *A. tumefaciens* Isopentenyltransferase (2ZE7) sind jedoch keine direkten Wechselwirkungen zwischen der Cysteinseitenkette und den Liganden in der Bindetasche zu beobachten. Trotz der verschiedenen Aminosäuren ist die wesentliche Ausrichtung des Rückgrats zur Diphosphatkomplexierung in den Modellen gegeben, auch wenn die konkreten Wechselwirkungen sich unterscheiden. Es ist davon auszugehen, dass in jedem Fall die hohe Elektronendichte am Diphosphat sowohl zum Metall-Gegenion als auch zu den entsprechenden polarisierten Wasserstoffatomen des *p-Loop*-Rückgrats starke elektrostatische Wechselwirkungen erzeugt.

Die Wassermoleküle 1 und 2 der Röntgenkristallstruktur stehen in Interaktion mit einem Sauerstoffatom der α -Phosphatgruppe. Inwiefern die drei involvierten Sauerstoffatome mit potenziellen Wasserstoffatomen an Wasserstoffbrückenbindungen untereinander beteiligt sind, geht aus dem Röntgenkristallstrukturmodell nicht hervor. In den erstellten Homologie-modellen sind keine Wassermoleküle enthalten. Falls die in der Röntgenkristallstruktur aufgelösten Wassermoleküle tatsächlich eine wichtige Funktion übernehmen, so spiegeln die Modelle dies nicht wider und andere Hypothesen können aufgestellt werden. Sugawara *et al.* diskutieren diese Wassermoleküle nicht.⁶⁶ Die Strukturoptimierung im Rahmen der hier durchgeführten Modellierung sorgt für plausible Positionierungen der Liganden und entsprechender Aminosäureseitenketten in dieser Region.

Das Serin an Position sieben des *p-Loop*-Motivs ist vollständig konserviert, was auch Abbildung 27 (Seite 61) wiedergibt. Dieser Serinrest ist in den AtIPT-Modellen so ausgerichtet, dass die freien Elektronen des Sauerstoffatoms der Hydroxylgruppe mit dem

entsprechenden Metallkation wechselwirken. Das ebenfalls konservierte Threonin an zweiter Stelle des p-Loop (Thr49 in Abbildung 27) spielt vermutlich eine Rolle bei der Aktivierung des Prenyldonormoleküls. Für das bakterielle Enzym mit einer Alaninmutante des entsprechenden Threonins wurde eine stark verminderte katalytische Aktivität berichtet.⁶⁶ Um die Prenyldonoraktivierung wahrnehmen zu können, müsste die Threoninseitenkette in den Modellen von AtIPT3 bis AtIPT8 jeweils um etwa 180° gedreht sein.

	 75		 85
AtIPT1	-SER-ASP-LYS-ILE-GLN-VAL-		-ILE-THR-THR-ASN-GLN-ILE-
AtIPT3	-SER-ASP-LYS-ILE-GLN-VAL-		-ILE-VAL-THR-ASN-LYS-ILE-
AtIPT4	-SER-ASP-LYS-MET-GLN-PHE-		-ILE-THR-THR-ASN-GLN-SER-
AtIPT5	-SER-ASP-LYS-ILE-GLN-VAL-		-ILE-VAL-THR-ASN-LYS-VAL-
AtIPT6	-SER-ASP-LYS-MET-GLN-ILE-		-ILE-VAL-THR-ASN-LEU-ILE-
AtIPT7	-SER-ASP-LYS-ILE-GLN-LEU-		-VAL-LEU-THR-ASN-LYS-VAL-
AtIPT8	-SER-ASP-LYS-ILE-GLN-PHE-		-VAL-THR-THR-ASN-GLN-MET-
2ZE7	-LEU-ASP-ARG-VAL-GLN-CYS-		-THR-GLY-SER-GLY-ARG-PRO-
3A8T	-SER-ASP-LYS-MET-GLN-VAL-		-ILE-THR-THR-ASN-LYS-ILE-

Abbildung 29: Segmente B und C des Sequenz-Alignment der AtIPT und der Template-Strukturen aus *A. tumefaciens* (2ZE7) und *H. lupulus* (3A8T)

Segment B in Abbildung 29 bildet eine weitere Gruppe benachbarter Aminosäuren, die mit dem Substratmolekül wechselwirken und sowohl in den Modellen als auch in den *Template*-Strukturen viele Aminosäureidentitäten zeigen. Das Aspartat an *Alignment*-Position 73 ist entsprechend der Homologie zur HIPT und 2ZE7 als die katalytisch aktive Aminosäure anzusehen, welche auch in den modellierten AtIPT im Verlauf der von Sugawara und Kollegen postulierten S_N2-Reaktion als Protonakzeptor fungiert.⁶⁶ Sugawara *et al.* schlagen weiter vor, dass unter anderem die in Abbildung 30 mit * markierten Einheiten an der Polarisierung der zu spaltenden Bindung beteiligt sind. Hierzu vermuten sie, dass Arg138 eine Konformationsänderung durchläuft und dem Sauerstoffatom an der zu brechenden Bindung Elektronendichte entzieht (siehe Abbildung 30).⁶⁶ Die diskutierte S_N2-Reaktion verläuft durch Protonübertragung von N⁶ auf die Carboxygruppe von Asp33 mit nukleophilem Angriff der freien Elektronen von N⁶ an C bei Abspaltung des Disphosphatrestes vom Prenylrest. Entsprechend dieses vorgeschlagenen Mechanismus ist es ebenfalls möglich, dass das Arginin an *Alignment*-Position 199 auch in den modellierten IPT die Funktion der Polarisierung der zu brechenden C-O-Bindung übernimmt, da diese Aminosäure über alle AtIPT, die HIPT und die *A. tumefaciens*-IPT konserviert ist.

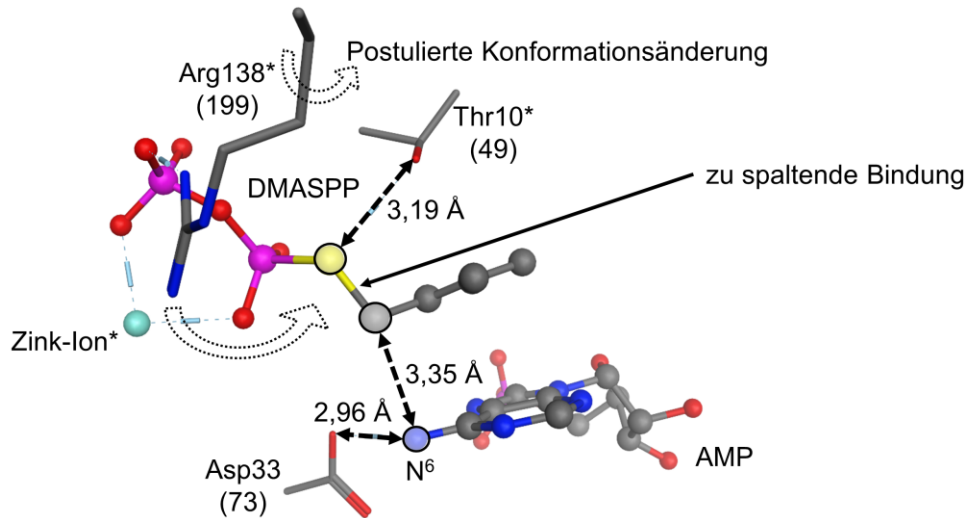


Abbildung 30: Ausschnitt aus der Bindetasche der Röntgenkristallstruktur der *A. tumefaciens* IPT (2ZE7). In Klammern ist jeweils die Position im MOE-Alignment mit den erstellten AtIPT-Modellen angegeben.

	140									
AtIPT1	-ILE-ALA-GLY-GLY-SER-ASN-SER-PHE-VAL-									
AtIPT3	-ILE-VAL-GLY-GLY-SER-ASN-SER-TYR-VAL-									
AtIPT4	-LEU-ALA-GLY-GLY-SER-ASN-SER-TYR-ILE-									
AtIPT5	-ILE-ALA-GLY-GLY-SER-ASN-SER-TYR-ILE-									
AtIPT6	-VAL-VAL-GLY-GLY-SER-ASN-SER-PHE-ASN-									
AtIPT7	-VAL-ALA-GLY-GLY-SER-ASN-SER-TYR-ILE-									
AtIPT8	-ILE-ALA-GLY-GLY-SER-ASN-SER-PHE-ILE-									
2ZE7	-LEU-GLU-GLY-GLY-SER-ILE-SER-LEU-LEU-									
3A8T	-LEU-VAL-GLY-GLY-SER-ASN-SER-PHE-ILE-									

Abbildung 31: Segment D des Sequenz-Alignment der AtIPT und der Template-Strukturen aus *A. tumefaciens* (2ZE7) und *H. lupulus* (3A8T)

Die ligandnahen Aminosäuren in Segment D des Alignment (Abbildung 31) bilden ein Sekundärstrukturelement welches parallel zum p-Loop positioniert liegt. Die Seitenkette des Serins an Position 140 des Alignment bildet in jedem Modell eine Wasserstoffbrücke zum Phosphatrest des Prenylakzeptormoleküls aus.

	195			200			
AtIPT1	-GLU-TYR-LEU-LEU-ARG-ARG-VAL-ASP-GLU-MET-MET-						
AtIPT3	-GLY-PHE-VAL-SER-GLU-ARG-VAL-ASP-LYS-MET-VAL-						
AtIPT4	-GLU-TYR-LEU-SER-LEU-ARG-LEU-ASP-LEU-MET-MET-						
AtIPT5	-SER-PHE-VAL-SER-GLU-ARG-VAL-ASP-LYS-MET-VAL-						
AtIPT6	-GLN-HIS-LEU-CYS-ASN-ARG-VAL-ASP-GLN-MET-ILE-						
AtIPT7	-SER-PHE-VAL-SER-LYS-ARG-VAL-ASP-ARG-MET-MET-						
AtIPT8	-GLU-TYR-LEU-SER-LYS-ARG-VAL-ASP-GLN-MET-MET-						
2ZE7	-THR-ARG-ALA-LYS-GLN-ARG-VAL-ALA-GLU-MET-PHE-						
3A8T	-ASP-TYR-LEU-ALA-LYS-ARG-VAL-ASP-ASP-MET-LEU-						

Abbildung 32: Segment E des Sequenz-Alignment der AtIPT und der Template-Strukturen aus *A. tumefaciens* (2ZE7) und *H. lupulus* (3A8T)

Die Bedeutung des Arginins an Position 199 in Segment E (Abbildung 32) wurde bereits diskutiert. Um jedoch rein sterisch die hypothetische Funktion der Polarisierung der zu brechenden Bindung wahrnehmen zu können, ist in den AtIPT-Modellen eine Konformationsänderung notwendig. Diese Konformationsänderung wird zum Teil durch die Seitenkette eines Asparagins (*Alignment*-Position 86, siehe Segment C in Abbildung 29) behindert, welches in den AtIPT konserviert ist, aber an dessen Position in der Agrobacterium-Struktur ein Glycin liegt. Die Situationsdarstellung in Abbildung 33 ist charakteristisch für die erstellten Modelle und veranschaulicht die relative Lage des Arginins und Asparagins zum Prenyldonormolekül. Das Prenyldonormolekül ist in Abbildung 33 ca. so orientiert wie in den vorangegangenen Strukturabbildungen in diesem Abschnitt.

Der Variation zwischen Leucin und Valin an Position 196 des *Alignment* (Abbildung 32) soll hier keine konkrete Bedeutung zugewiesen werden, weil die Aminosäuren sich chemisch sehr ähneln und den hydrophoben Teil der Dimethylallyl-Bindetasche begrenzen.

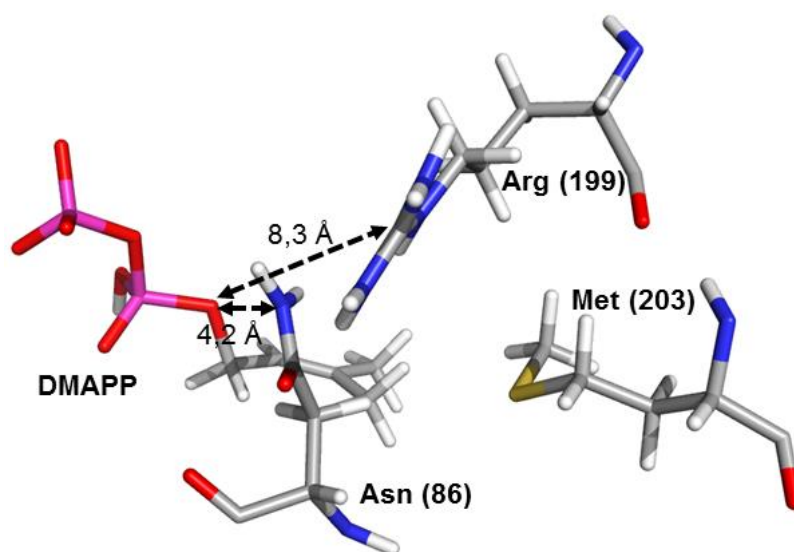


Abbildung 33: Ausschnitt aus der Bindetasche des Modells von AtIPT5. In Klammern ist jeweils die Position im *Alignment* mit den erstellten AtIPT-Modellen angegeben. Die Abbildung wurde unter Verwendung von PyMOL⁴⁰³ erstellt.

Auch das konservierte Isoleucin (*Alignment*-Position 233 in Segment F, siehe Abbildung 34) ist in allen AtIPT-Modellen Teil der hydrophoben Bindetasche und bildet jeweils mit der Aminogruppe Wasserstoffbrücken zum Adeninteil des Prenylakzeptormoleküls aus. Die unterschiedlichen Aminosäuren Alanin, Threonin bzw. Isoleucin an Position 232 von Segment F bilden einen Teil der Prenylakzeptorbindetasche und interagieren ebenfalls mit dem Adenylrest. Die positiv geladenen Aminosäuren an Position 231 ragen mit ihrer Seitenkette in Richtung des Lösungsmittels und bilden über ihre Sauerstoffatome Wasserstoffbrückenbindungen zum Zuckerrest des Prenylakzeptors. Interessant sind die Aminosäuren an den *Alignment*-Positionen 231 und 233 in der *A. tumefaciens* IPT (2ZE7). In der Röntgenkristallstruktur ist das Aspartat an Position 233 an der Komplexierung eines Metallions beteiligt, während das Aspartat an Position 231 ins Lösungsmittel ragt und das Isoleucin an Position 232 einen Teil der Bindetasche für den Adenylrest bildet.

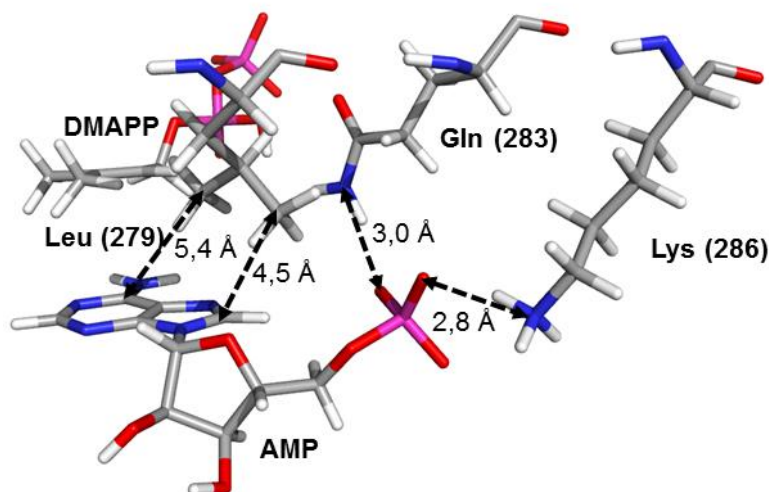


Abbildung 36: Ausschnitt aus der Bindetasche des Modells von AtIPT5. In Klammern sind die *Alignment*-Positionen angegeben. Die Abbildung unter Verwendung von PyMOL erstellt.

Für AtIPT3 identifizierte Galichet *et al.* nahe des C-Terminus eine Cysteinseitenkette, deren Farnesylierung sich auf die intrazelluläre Lokalisation auswirkt und die somit die Cytokininbiosynthese beeinflusst.²²⁸ Diese Region ist hier im entsprechenden Modell nicht vorhanden, weil eine Modellierung dieser terminalen Region zu große Unsicherheiten aufweist und somit zu sehr spekulativen Modellen führt bzw. allgemein aufgrund ihrer Lösungsmittel exponiertheit sehr flexibel sein könnte.

4.1.1.4 Diskussion der Proteinmodelle und der experimentellen Ergebnisse

Die Ähnlichkeit der AtIPT-Sequenzen zum *Template*, insbesondere bezüglich der an der Katalyse unmittelbar beteiligten Aminosäuren, lässt darauf schließen, dass die Modelle für die AtIPT im Wesentlichen korrekt sind und der katalytische Mechanismus auch in den modellierten AtIPT so funktioniert, wie von Sugawara *et al.* für HIPT beschrieben. Für AtIPT6 ist anzumerken, dass zwar ein plausibles Modell erstellt wurde, das zugehörige Gen aber *in vivo* nicht exprimiert wird.

Die Frage der Positionierung der diskutierten Argininseitenkette (siehe Abbildung 30, Seite 64 und Position 199 in *Alignment*-Segment E, Abbildung 32) kann mit theoretischen Methoden nur schwierig abschließend geklärt werden. Moleküldynamiksimulationen mit Kraftfeldern sind bezüglich der Modellierung von intermolekularen Wechselwirkungen relativ ungenau.^{268,404,405} Auch die notwendige Dauer einer möglichen Simulation, um statistisch signifikante Aussagen treffen zu können, ist diskutabel und eng mit der Wahl einer entsprechenden Methode verknüpft.²⁶³

Es stellt sich die Frage, ob das konservierte Threonin an zweiter Position des *p-Loop* in den Modellen von AtIPT3 bis AtIPT8 deshalb eine der Röntgenstruktur 2ZE7 konträre Rotation aufweist, weil das DMAPP-Molekül am α -Phosphat protoniert ist (vgl. Abbildung 27, Seite 61). Dadurch wären keine so starken elektrostatischen Wechselwirkungen möglich wie im Fall des vollständig deprotonierten α -Phosphats. Die entsprechende Protonierung entstammt der YASARA-Homologiemodellierung, welche auf der Basis lokaler Ladungsdichteverteilungen und tabellierter pK_s -Werte Protonierungsgrade zuweist.⁴⁰⁶ Da auch die Röntgenkristallographie in aller Regel nicht das Auflösungsvermögen bietet, um Protonierungsgrade zweifelsfrei zu bestimmen, kann hier über die Protonierung keine endgültige Aussage gemacht werden. Diana Schulze hat in ihrer Dissertation die Aktivierung

von Diphosphaten am Beispiel von DMAPP untersucht.²¹⁰ Sie beschreibt, dass ein geringerer Protonierungsgrad des Diphosphatrestes die Hydrolyse des organischen Restes vom Diphosphat bei Angriff eines Wassermoleküls erleichtert. Außerdem konnte sie durch quantenchemische Berechnungen an DMAPP zeigen, dass der C-C-C-O-Torsionswinkel in diesem Molekül bei 90° die Mesomeriestabilisierung des bei Diphosphatabspaltung entstehenden allylischen Karbokations fördert. Dies ging einher mit statistischen Untersuchungen an Röntgenkristallstrukturmodellen diphosphatbindender Proteine. Im Mechanismus, welcher in der vorliegenden Arbeit vorgeschlagen wird, kommt es initial nicht zu einer Protonierung des Diphosphatrestes. Stattdessen wird der Diphosphatrest durch Spaltung der C-O-Bindung ionisiert, so dass die Destabilisierung dieser Bindung und gleichzeitige Stabilisierung des Diphosphatrestes als treibende Kräfte gelten. Hinzu kommt als klassische Triebkraft der S_N2-Reaktion die HOMO-LUMO-Wechselwirkung zwischen dem nukleophilen Prenylakzeptor und dem im Reaktionsverlauf ionisierenden Dimethylallylrest (HOMO: engl. *highest occupied molecular orbital*, (energetisch) höchstes besetztes Molekülorbital; LUMO: engl. *lowest unoccupied molecular orbital*, (energetisch) niedrigstes unbesetztes Molekülorbital). Mit der Erkennung des Diphosphatrestes durch den *p-Loop* und der zusätzlichen Komplexierung eines divalenten Metallions ist eine starke Stabilisierung gegeben, die zwischen den Modellen leichte strukturelle Unterschiede aufweist.

Inwiefern die Bindetaschen der beiden Substrate für Solvensmoleküle zugänglich sind, ist mit theoretischen Methoden derzeit nicht zu beweisen. Einerseits müssten hierzu intermolekulare Wechselwirkungen akkurat berechnet werden. Andererseits sind extrem lange Simulationszeiten notwendig, um auszuschließen, dass entworfene Startstrukturen nur unbedeutende lokale Energieoptima oder kurzlebige Zustände darstellen.

Die Datenlage in der Literatur zu molekularen und enzymatischen Eigenschaften der hier modellierten AtIPT ist unvollständig und heterogen:

- AtIPT1 besitzt DMAPP:AMP-Transferaseaktivität.²¹⁴
- AtIPT3 wurde bisher nicht konkret charakterisiert.
- AtIPT4 akzeptiert ATP/ADP, aber nicht AMP.²¹⁵
- AtIPT5 wurde ebenfalls bisher nicht konkret charakterisiert.
- AtIPT6 wird *in vivo* nicht exprimiert.²¹⁹
- AtIPT7 ist bisher auch nicht konkret charakterisiert.
- AtIPT8 wurde nicht konkret charakterisiert, aber durch Extraktion aus *in vivo*-modifizierten Pflanzen untersucht.³⁸³

Lukáš Spíchal und Kollegen fanden Hinweise darauf, dass AtIPT in der Lage sind, differenziert DMAPP bzw. *cis*- bzw. *trans*-HMBPP mit AMP/ADP und ATP umzusetzen (unveröffentlichte Ergebnisse). Dazu wurden AtIPT1, 3, 4, 5 und 8 heterolog exprimiert und aufgereinigt und entsprechenden Enzymtests unterzogen. Allerdings konnte AtIPT7 nicht erfolgreich exprimiert werden und aufgereinigte AtIPT3 zeigte keine Aktivität. Die abgeleiteten Enzymaktivitäten der übrigen AtIPT zeigen ein sehr differenziertes Bild von Bevorzugungen unterschiedlicher Substratkombinationen, sowohl bezüglich des Prenyldonor- als auch des Prenylakzeptormoleküls. Die große Ähnlichkeit der erstellten Enzymmodelle lässt jedoch wenig Raum für eine Rationalisierung von Struktur-Wirkungs-Beziehungen mit theoretischen Methoden. Für eine solche Adressierung dieser Fragestellung sind sehr aufwändige Berechnungen notwendig, denn die unterschiedlichen Enzymaktivitäten können aus verschiedenen Ursachen resultieren, zum Beispiel:

- unterschiedliche Bindungsaffinitäten der Substrate zum jeweiligen Enzym
- entropische Energiedifferenzen zwischen verschiedenen Umsetzungsprozessen
- Flexibilitätsunterschiede in beteiligten Sekundärstrukturelementen der verschiedenen Isoenzyme

Weitere unveröffentlichte Ergebnisse aus der Arbeitsgruppe von Lukáš Spíchal zu AtIPT4 bestätigen eine wichtige Rolle der Aminosäure Asparagin an *Alignment*-Position 86 (siehe Abbildung 29, Seite 63 und Abbildung 33, Seite 65) in der Vermittlung der Substratspezifität. Durch zielgerichtete Mutagenesen an der *A. tumefaciens* IPT konnten ebenfalls Voraussetzungen für die Akzeptanz von HMBPP als Substrat identifiziert werden.⁶⁶ Diese können jedoch aufgrund der unvollständigen Homologie zwischen den modellierten AtIPT und der *A. tumefaciens* IPT im Bereich der Bindetasche nicht auf die hier erstellten Modelle übertragen werden.⁶⁶

4.1.2 Suche nach Inhibitoren

In diesem Abschnitt werden die Ergebnisse der Inhibitorsuche für die modellierten AtIPT präsentiert. Die Pharmakophorsuche nach DMASPP-Analoga lieferte eine Molekülstrukturdatenbank mit 3.540 Molekülen in 15.332 Konformationen. Die relative Molekülmasse der resultierenden Strukturen liegt zwischen 88 und 1908 wobei fast zwei Drittel der Strukturen eine relative Molekülmasse von unter 200 besitzen.

4.1.2.1 Auswertung und Diskussion der Pharmakophorsuche

Eine willkürliche Auswahl (siehe Abbildung 37) von Treffermolekülen zeigt diverse Probleme des virtuellen *Screening* auf.

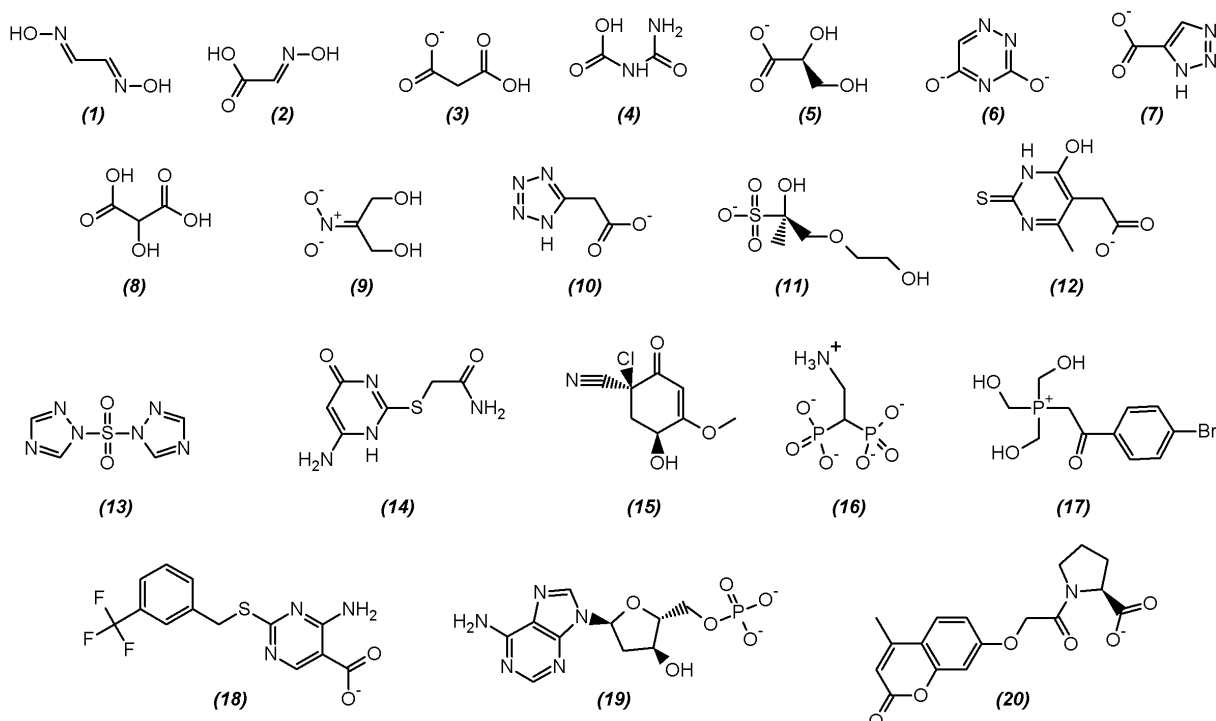


Abbildung 37: Zwanzig Beispiele für Strukturen von Treffern der Pharmakophorsuche. Die Darstellung von Protonierungsgraden und Delokalisationen stammt aus der Strukturdarstellung von MOE.

Einerseits kann ein großer Teil der Pharmakophorsuchergebnisse mit sehr geringem Molekulargewicht nicht genug Spezifität für eine Erkennung vom Enzym bieten, weil kleine Liganden meist kombinatorisch mehr Wechselwirkungsmöglichkeiten mit Rezeptoren haben. Andererseits sind höhermolekulare Strukturen zum Teil zu groß für die modellierten Binde-taschen. Die Ursache hierfür liegt in der Definition der kugelförmigen Ausschlussvolumina im Pharmakophor: In Richtung des hydrophoben Teils des Liganden wurden keine Ausschlussvolumina definiert, um evtl. passende Substituenten zu finden. Eine Herausforderung ist ferner die Zuweisung eines sinnvollen Protonierungsgrades, vor allem, wenn die Datenbank-einträge keine Protonierung beinhalten. Der konkrete Protonierungszustand ist allerdings wesentlich für das Infragekommen als Pharmakophortreffer. Durch die Lage des Diphosphatrestes im *p-Loop* und die daraus resultierende Positionierung der Pharmakophormerkmale resultiert in vielen Datenbanktreffern eine Carboxygruppe für die beiden anionischen Pharmakophormerkmale des β -Phosphates. Eine solche Carboxygruppe ist allein nicht in der Lage, das Metallion in der Bindetasche sinnvoll zu komplexieren. Dies liegt an der in der Regel tetrahedralen Komplexierung der relevanten Metallionen, welche die Präsenz weiterer Nukleophile bedingt. Ferner waren auch solche Strukturen Treffer der Pharmakophorsuche, die als reaktiv annotiert sind. Diese Substanzen kommen für die Inhibition von Enzymen in wässriger Lösung jedoch nicht in Frage.

Diese Unzulänglichkeiten des reinen Pharmakophorsuchprozesses können durch Filter über geeignete Beschreiber adressiert werden. Im Rahmen dieses Projektes war jedoch ein Ziel, neue Liganden für AtIPT zu synthetisieren, so dass die Ergebnisse des virtuellen *Screening* bis zu diesem Punkt als Inspiration für den modellbasierten Entwurf potenzieller Inhibitorstrukturen dienen. Da Enzyme Übergangszustände stabilisieren, sind Übergangszustandsanaloge als Inhibitoren wünschenswert, denn sie gehen erwartungsgemäß stärkere Wechselwirkungen mit dem Enzym ein als Substratanaloga.

4.1.2.2 Ergebnisse der Inhibitionstests im Labor

Unter Beachtung der Ergebnisse des virtuellen *Screening* und der strukturellen Gegebenheiten in den Prenyldonor-Bindetaschen der modellierten AtIPT wurden in Zusammenarbeit mit Dimitar Vasilev Strukturen zur chemischen Synthese ausgewählt. Dimitar Vasilev stellte anschließend einige vorgeschlagene Substanzen und eine Reihe von Derivaten synthetisch her. Es wurden 17 synthetisierte Substanzen zur Aktivitätsbestimmung an die Arbeitsgruppe von Dr. Lukáš Spíchal (Palacký Universität, Olomouc, Tschechische Republik) übersendet. Von diesen durch die Kooperationspartner getesteten potenziellen Inhibitoren weisen sechs Substanzen IC_{50} -Werte im Bereich zwischen 345 μ M und 847 μ M auf. Dies entspricht einer sehr geringen inhibitorischen Potenz. Für die restlichen 11 Substanzen ließ sich keine Inhibition an AtIPT1 nachweisen (unveröffentlichte Ergebnisse). In Anbetracht der differenziellen Umsetzungen von DMAPP und den hydroxylierten Varianten *cis*- und *trans*-HMBPP auch in Kombination mit den Nukleotidmono-, di- und triphosphaten sind Inhibitionstests auch an den anderen AtIPT interessant.

Die verwendete Methodik der Inhibitionsmessung mittels ELISA weist keine Bindung der Inhibitoren am Enzym nach, da lediglich die Produktbildung bei Zugabe der potenziellen Inhibitoren untersucht wurde. Dass sich auf Basis des virtuellen *Screenings* und der anschließenden Auswahl zur Synthese keine Substanzen mit stärkerer Inhibitionswirkung auf AtIPT1 finden ließen, ist den erwähnten Unzulänglichkeiten des virtuellen *Screenings* und den limitierten Möglichkeiten zur chemischen Synthese und der aufwändigen anschließenden Testung geschuldet.

Vorbehaltlich weiterer experimenteller Untersuchungen wie Derivatisierung und Optimierung der Enzymaktivitätstests werden in dieser Arbeit keine Strukturformeln der gefundenen inhibitorisch wirksamen Substanzen präsentiert.

4.1.2.3 Diskussion der Inhibitorsuche

Virtuelles *Screening* nach Liganden ist in der Regel ein grober, vorgeschalteter Prozess zum Ausschluss vieler Substanzen, die bereits theoretisch nicht binden können. Allerdings kann es auch immer falsch positive (vorhergesagte Liganden, die real nicht binden) und falsch negative (ausgeschlossene Strukturen bindender Substanzen) Treffer geben. Die Tatsache, dass Inhibitoren – in diesem Fall vermutliche Liganden – gefunden wurden, zeigt jedoch, dass virtuelles *Screening* zu Recht ein wichtiges Werkzeug der Wirkstoffforschung ist.

Die IC₅₀-Werte der sechs gefundenen Inhibitoren sind weder klein genug noch ausreichend gestreut, um Struktur-Wirkungs-Beziehungen aufstellen zu können. Eine Optimierung der Inhibitoren durch Derivatisierung und weitere Aktivitätstests, ggf. auch an AtIPT3, AtIPT4, AtIPT5, AtIPT7 und AtIPT8, kann neue Erkenntnisse liefern, welche in die Entwicklung spezifischer Inhibitoren der Cytokininbiosynthese durch IPT eingehen können.

4.2 Kauren-Synthase-ähnliche Enzyme

Das Vorgehen bei der Modellierung für die KSL-Enzyme unterscheidet sich maßgeblich von der Herangehensweise bei den AtIPT (Abschnitt 4.1). Grund hierfür ist die geringere Sequenzhomologie zwischen den KSL-Enzymen als zwischen den sieben Isoenzymen der AtIPT. Zur inhaltlichen Unterscheidbarkeit der beiden Abschnitte wird für die Diskussion einzelner Aminosäuren bei AtIPT der Dreibuchstabencode und bei KSL-Enzymen der Einbuchstabencode verwendet.

4.2.1 Voruntersuchungen

Die YASARA-Homologiemodellierung identifizierte die potenziellen *Template*-Strukturen

- *Salvia officinalis* (+)-Bornyldiphosphatsynthase (PDB-Identifikator 1N1B, ³¹)
- *Mentha spicata* (4S)-Limonensynthase (PDB-Identifikator 2ONH, ¹²)
- *Salvia fruticosa* 1-8-Cineolsynthase (PDB-Identifikator 2J5C, ⁴⁰)
- *Nicotiana tabacum* 5-*epi*-Aristolochensynthase (PDB-Identifikator 3M01, ⁴⁵)
- *Taxus brevifolia* Taxadiensynthase (TbTS, PDB-Identifikator 3P5R, ²⁷).

Per Sequenzvergleich gilt die TbTS als den Zielproteinen AtKS, NtABS, ShSBS und SIPHS am ähnlichsten, was Abbildung 38 veranschaulicht. Die Struktur der Taxadiensynthase aus *T. brevifolia* gilt damit als sinnvollstes *Template* für eine Homologiemodellierung der behandelten KSL-Enzyme.

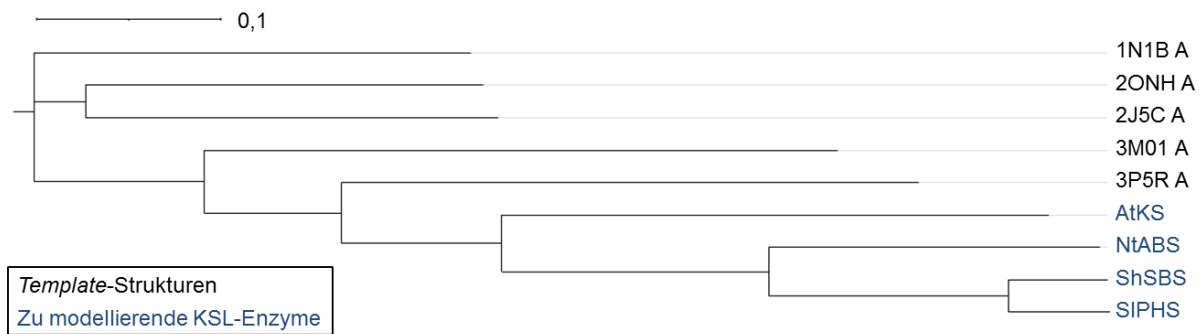


Abbildung 38: Baumdarstellung der Sequenzähnlichkeit zwischen den zu modellierenden KSL-Enzymen und potenziellen *Templates*. Die Abbildung wurde mit iTOL auf Basis eines ClustalW-*Alignments* erstellt, woher auch die Distanzen stammen. Die zusätzliche Bezeichnung mit „A“ steht für die beim *Alignment* verwendete Proteinkette.

Die Suche nach Enzymen, die in ihrer Sequenz der AtKS ähneln, liefert zahlreiche Treffer uncharakterisierter Enzyme, aber auch bereits beschriebener KSL-Enzyme (siehe Abbildung 39).

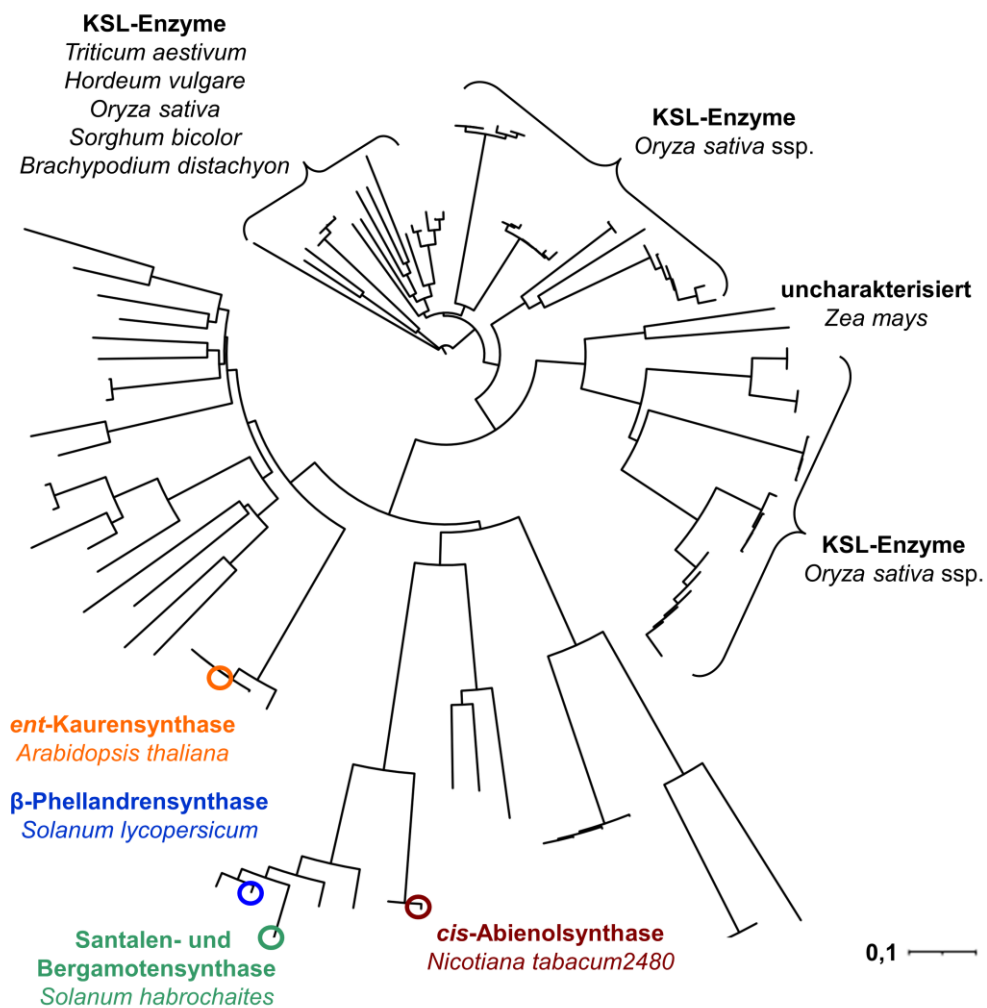


Abbildung 39: Baumdarstellung der Sequenzähnlichkeit von 101 Aminosäuresequenzen, die per blastp^{369,374,375} die größte Ähnlichkeit zur *A. thaliana* ent-Kaurensynthase-Sequenz haben. Die Abbildung wurde mit iTOL erstellt. Das Abstandsmaß stammt aus paarweisen *Alignments* entsprechend des *Blast Tree View*.

Wie Abbildung 39 veranschaulicht, weisen die KSL-Enzyme SIPHS, ShSBS und NtABS untereinander kleinere Sequenzdistanzen auf als zu AtKS. Da sie zur Familie der Solanaceae gehören, untermauert das die botanische Klassifizierung.

4.2.2 Homologiemodellierung

4.2.2.1 Qualität der erstellten Modelle

Die erstellten Modelle sind entsprechend des *Template* Dimere. Die Angaben in Tabelle 15 weisen auf eine gute Qualität der Modelle hin (siehe auch ProSA-web-Ausgaben: A 7, Seite 152).

Tabelle 15: YASARA-z-Scores (1.-4. Zeile) und ProSA-web-z-Scores (letzte Zeile) für die Bewertung der finalen Homologiemodelle (jeweils Kette A)

	AtKS	NtABS	ShSBS	SIPHS
Torsionswinkel	0,434	0,231	0,368	0,422
Packing1D	-1,261	-1,090	-1,406	-1,292
Packing3D	-0,707	-0,994	-1,063	-1,102
Insgesamt	-0,757	-0,854	-0,989	-0,955
z-Score	-13,02	-11,91	-11,98	-11,65

Als Resultat der Homologiemodellierung fehlen terminal jeweils einige Aminosäuren zur kompletten Aminosäuresequenz der Enzyme. Dies wirkt sich nicht auf die im Folgenden angeführten Betrachtungen aus.

4.2.2.2 Diskussion der modellierten Enzym-Ligand-Komplexe

Um Hypothesen für die Funktion einzelner Aminosäuren in der Nähe der Bindetasche der verschiedenen Enzymmodelle aufstellen zu können wurden anhand der 3D-Strukturen und des erstellten *Alignment* Unterschiede und Gemeinsamkeiten untersucht. Zur Veranschaulichung in den *Alignment*-Abschnitten wurden die putativen Funktionen einzelner Aminosäuren entsprechend der Farbkodierung in Abbildung 40 auf Basis der erstellten Homologiemodelle manuell annotiert. Die Abgrenzung der Funktionen voneinander erfolgt hierbei wie folgend. Als hydrophober Teil der Substratbindetasche wird der Bereich abseits der Metallbindungsmotive definiert. Aminosäuren sind nur dann als metallbindend markiert, wenn eine solche Wechselwirkung im Modell unmittelbar stattfindet. Weitere Aminosäuren der bekannten Metallbindemotive wurden nicht eingefärbt und verbleiben in schwarzer Farbe. Zur Markierung von Besonderheiten sind in den *Alignment*-Abschnitten einige Aminosäuren grün eingefärbt, welche dann explizit im laufenden Text diskutiert werden.

Für die im Folgenden anhand des Sequenz-*Alignment* diskutierten Aspekte dienen Abbildung 43 bis Abbildung 46 (Seite 76ff.) zur Veranschaulichung der erstellten Proteinmodelle.

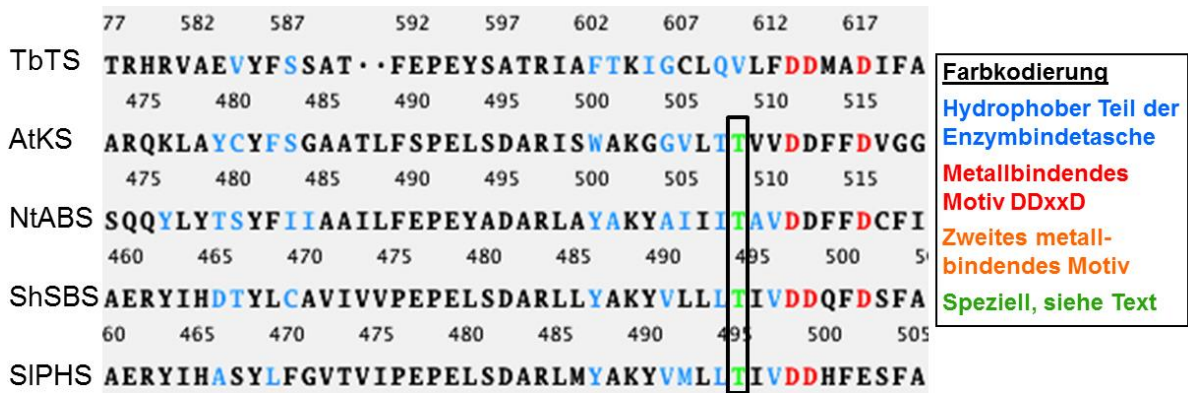


Abbildung 40: Sequenz-Alignment-Ausschnitt mit dem metallbindenden DDXD-Motiv. Die Positionsangaben beziehen sich jeweils auf die absolute Position der Aminosäure (unter der mittleren Ziffer der Zahl) in der jeweiligen Sequenz.

In allen vier Modellen bildet die Seitenkette des in Abbildung 40 grün markierten Threonins eine Wasserstoffbrücke zu einem Sauerstoffatom im Proteinrückgrat. Die Hydroxylgruppe ragt mit den freien Elektronenpaaren des Sauerstoffatoms in den hydrophoben Teil der Bindetasche und könnte mit Zwischenzuständen des Prenyldonor-Substrates interagieren. In TbTs könnte C719 eine analoge Funktion (Abbildung 41) übernehmen.

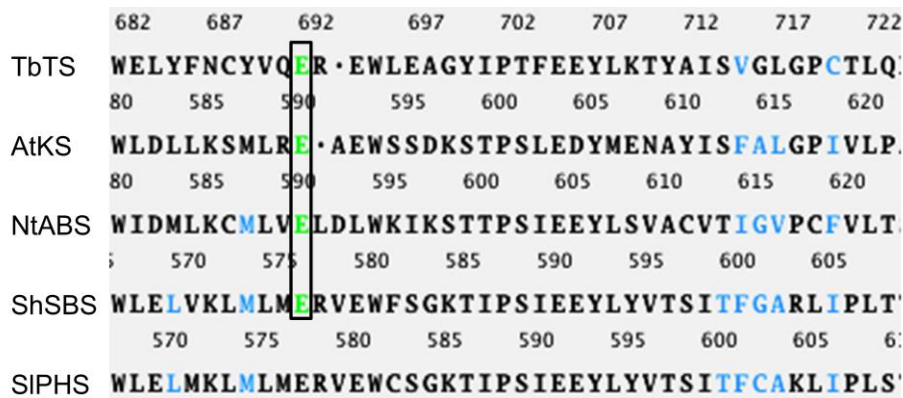


Abbildung 41: Alignment-Abschnitt mit einem konservierten Glutamin, welches laut der erstellten Homologiemodelle an der Metallionenbindung beteiligt ist (außer im Modell der SIPHS). Die Positionsangaben beziehen sich jeweils auf die absolute Position der Aminosäure (unter der mittleren Ziffer der Zahl) in der jeweiligen Sequenz. Die Farbkodierung entspricht der in Abbildung 40 verwendeten.

Abbildung 41 zeigt einen weiteren Abschnitt aus dem Alignment mit Aminosäuren der hydrophoben Teile der jeweiligen Bindetaschen und das erwähnte C719 der TbTS, welches in den modellierten Proteinen im Alignment hydrophoben Aminosäuren gegenübersteht. Dies entspricht einer Umkehrung der Verhältnisse aus Abbildung 40. Hier ist ein Funktionstausch zwischen V610 in TbTS und den markierten Threoninen (in Abbildung 40) gegenüber C719 in TbTS und den entsprechenden Isoleucinen bzw. Phenylalanin (in Abbildung 41) postulierbar: Die hydrophobe Aminosäure in TbTS korrespondiert im Sequenz-Alignment mit polaren Aminosäuren in den KSL-Enzymen und umgekehrt.

	747	752	757	762	767	772		827	832	837	842	847	852
TbTS	FELVSLSW	RLT	NDTK	TYQA	EKARG	QQ		IFNLR	LCVQ	IFY	KFID	GYGIAN	EEIKDY
	645	650	655	660	665	670		725	730	735	740	745	
AtKS	YKLVSTMGR	LLN	DIQGF	KRESA	EGKL			FLKMS	KVLN	LFY	RKDD	GFT	·S·NDLMSL
	45	650	655	660	665	670		725	730	735	740	745	750
NtABS	CNCTAAV	ARL	INDI	HSYKRE	QAESST			FWTTI	NAAY	SIH	THGD	GYRFP	·EEFKNH
	635	640	645	650	655			710	715	720	725	730	
ShSBS	CNCTGRV	LRI	LDLQ	DSKKE	QKEDSV			FWRTSN	WADF	IYL	LQTD	GYRIA	·EEMKNH
	635	640	645	650	655			710	715	720	725	730	735
SIPHS	WNCSGRV	MRI	LNDL	QDSKRE	QKEVSI			FWRTSK	WAHF	TYI	SQTD	GYRIA	·EEMKNH

Abbildung 42: Alignment-Abschnitte mit dem zweiten Metallbindemotiv und weiteren Aminosäuren des hydrophoben Teils der Bindetasche. Links ist ein konserviertes Arginin hervorgehoben, das in allen Modellen an der Komplexierung des β -Phosphates beteiligt ist. Rechts ist das hervorgehobene Tyrosin (bzw. Histidin bei NtABS) möglicherweise an der Protonabstraktion beteiligt. Die Positionsangaben beziehen sich jeweils auf die absolute Position der Aminosäure (unter der mittleren Ziffer der Zahl) in der jeweiligen Sequenz. Die Farbkodierung entspricht der in Abbildung 40 verwendeten.

In Abbildung 42 wird angedeutet, dass das Strukturmodell von NtABS eine andere Metallionenkomplexierung als die restlichen Modelle aufweist. Dies wird im Vergleich der folgenden Bindetaschenausschnitte (Abbildung 43 bis Abbildung 46) deutlich. Da jedoch das zweite Metallbindemotiv in der Literatur als konserviert beschrieben ist (siehe Köksal *et al.*,²⁷) und das *Alignment* dies auch widerspiegelt ist an dieser Stelle von einem Artefakt der Modellierung auszugehen. Eine Möglichkeit, wie ein solches Artefakt zu Stande kommen kann, besteht in einer groben Homologiemodellierung, wenn eine andere anionische Aminosäure als die hier üblichen einem Metallion nahekommt. Dies kann durch Geometrieoptimierung nicht behoben werden, wenn die Energiebarriere die Komplexierung eines Metallions von einer nukleophilen Aminosäure (zum Beispiel Aspartat) zu lösen nicht überwunden wird.

Das in Abbildung 42 markierte Arginin ist in den Modellen auf verschiedene Arten an der Diphosphaterkennung beteiligt. Während es in den Modellen von AtKS, NtABS, SIPHS und der *Template*-Struktur TbTS mit dem β -Phosphat wechselwirkt, interagiert es in ShSBS mit dem α -Phosphat des Liganden.

Im Röntgenkristallstrukturmodell von TbTS ist Y835 so positioniert, dass es eventuell mit der Hydroxylgruppe an der Protonabstraktion beteiligt sein könnte. Auffällig ist die Konservierung (siehe Abbildung 42) dieses Tyrosins im Vergleich zu benachbarten Aminosäuren. Die gleiche vorgeschlagene Funktion könnten aber auch Wassermoleküle nahe der halboffenen Substratbindetasche übernehmen und auch eine Deprotonierung durch das Diphosphation ist grundsätzlich möglich.

Im Allgemeinen konkurriert in Terpensynthesen die Deprotonierung mit der Hydrolyse von intermediären Kationen.³⁶⁷ Die Thermodynamik der Hydrolyse von GPP wurde von Brandt und Kollegen mit quantenchemischen Berechnungen untersucht.³⁶⁷ Sie zeigten, dass die Hydrolyse in basischer Umgebung thermodynamisch ungünstig ist, während sie unter sauren Bedingungen spontan ablaufen kann. Whittington *et al.* spekulieren am Beispiel der Bornyl-diphosphatsynthese aus *Salvia officinalis* über Protonübertragung auf Wasser.³¹ Dies kann durch die hier angeführten Modelle weder bestätigt noch widerlegt werden, weil das

Vorhandensein von Wassermolekülen in der Substratbindetasche bzw. die Zugänglichkeit intermediärer Kationen für Wasser mit theoretischen Methoden kaum belegbar sind. Eine weitere Hypothese zur Protonabstraktion in Terpensynthesen ist die Übertragung auf das Diphosphation. Diese Möglichkeit wurde hauptsächlich durch Betrachtungen der jeweiligen Bindetaschen abgeleitet, in welchen alternative Deprotonierungsvorgänge unwahrscheinlich scheinen.^{30,407,408} Auch diese Variante der Deprotonierung muss für die erstellten KSL-Enzymmodelle in Betracht gezogen werden. Es ist zu erwähnen, dass die spezifische Protonabstraktion in manchen Terpensynthesen die Produktbildung moduliert, wie Ergebnisse von Garms *et al.*⁴⁰⁹ und Jindal und Sunoj⁴¹⁰ zeigen.

Die folgenden Abbildungen der dreidimensionalen Strukturen der modellierten Enzyme zeigen, dass jeweils aromatische und hydrophobe Aminosäuren den Teil der Bindetasche dominieren, in welchem der Prenylrest positioniert ist (jeweils erstellt unter Benutzung von MOE; Wassermoleküle und apolare Wasserstoffatome sind nicht dargestellt und Kohlenstoffatome folgen dem Farbschema aus Abbildung 40, Seite 74).

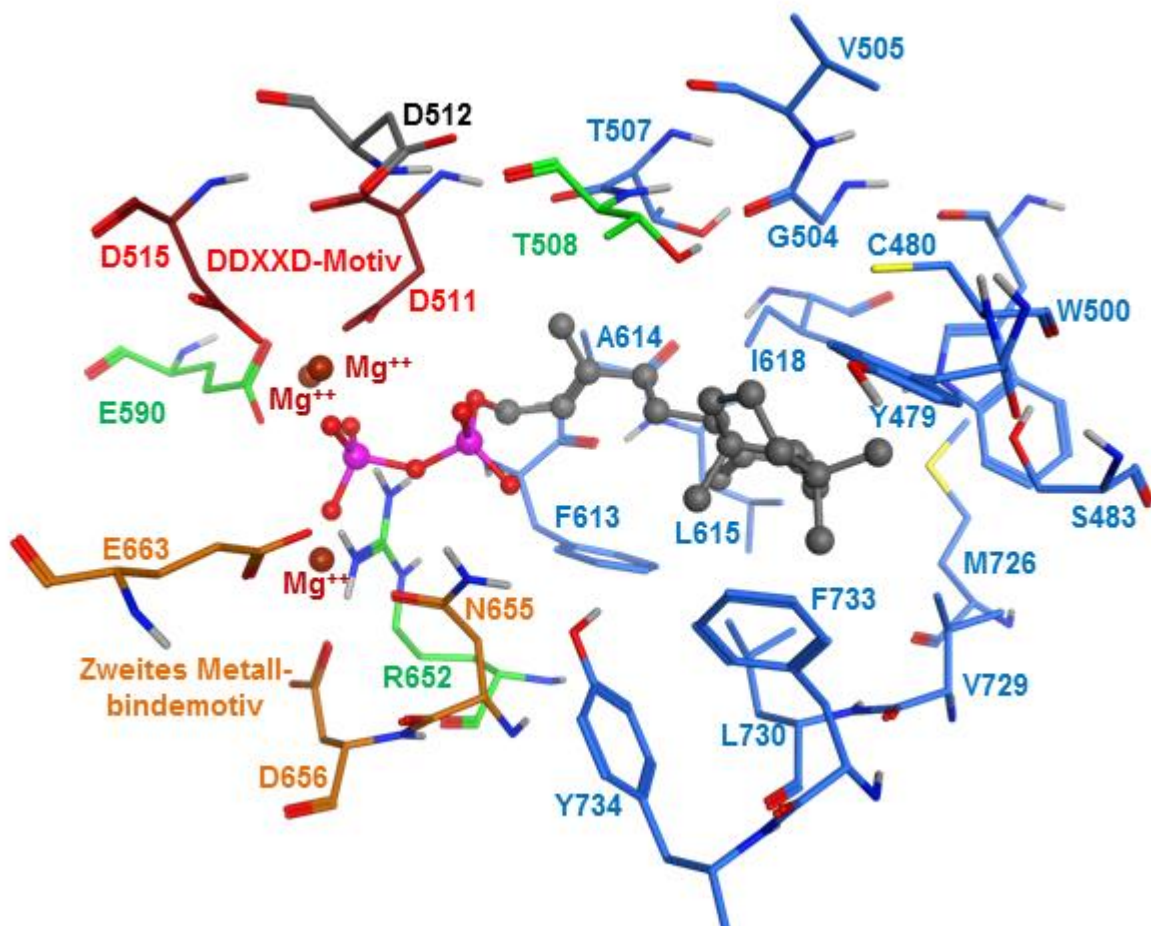


Abbildung 43: Substratbindetasche im AtKS-Modell mit *ent*-Copolyldiphosphat und Magnesiumionen. Die Farbkodierung ist in Abbildung 40 (Seite 74) definiert.

AtKS katalysiert die Ionisierung des Substrates *ent*-Copolyldiphosphat und die beiden anschließenden Zyklisierungen zu *ent*-Kauren. Die abschließende Deprotonierung muss, wenn es nicht zu signifikanten Bewegungen der Intermediate in der Bindetasche kommt, in der Nähe des komplexierten Diphosphatrestes stattfinden. Die gefundene Diphosphatkomplexierung ist insofern ungewöhnlich, als dass alle drei Magnesiumionen mit dem β -

Phosphat interagieren, wie in Abbildung 43 dargestellt. Eine alternative Positionierung des Diphosphations mit je einem Magnesiumion komplexiert an α - und β -Phosphat und dem dritten Magnesiumion zwischen α - und β -Phosphat scheint anhand des Modells möglich. Es kann sich bei der vorliegenden Positionierung des Diphosphatrestes und der Metallionen durchaus um ein Artefakt der Modellierung handeln. Allerdings gibt es Röntgenkristallstrukturmodelle von Klasse-I-Terpensynthasen mit komplexierten Diphosphatresten, in denen die ionischen Metall-Phosphat-Wechselwirkungen durch Wassermoleküle überbrückt werden und davon abgesehen ähnliche Geometrien wie in Abbildung 43 zu Stande kommen (beispielsweise bei 5-*epi*-Aristolochensynthase, PDB-Identifikator 3M02, ⁴⁵).

Zur Stabilisierung von intermediären Zuständen kommt in diesem Modell vor allem F613 in Frage, da es mit der Lage seines aromatischen Systems potenziell stark mit intermediären kationischen Spezies interagieren kann. T508 könnte durch die freien Elektronen der Hydroxylgruppe mit Intermediaten wechselwirken. Die zahlreichen anderen hydrophoben und aromatischen Aminosäuren der Bindetasche sind in diesem Enzym vermutlich für die Substratspezifität mitverantwortlich, aber eine Involvierung in die katalysierte Reaktion ist abgesehen von sterischen und hydrophoben bzw. van-der-Waals-Interaktionen unwahrscheinlich.

In der in Abbildung 43 gezeigten Konformation des Liganden im AtKS-Modell ist eine Beteiligung von Y734 an der Protonabstraktion unwahrscheinlich, weil der Abstand zum Substratmolekül über 5 Å beträgt.

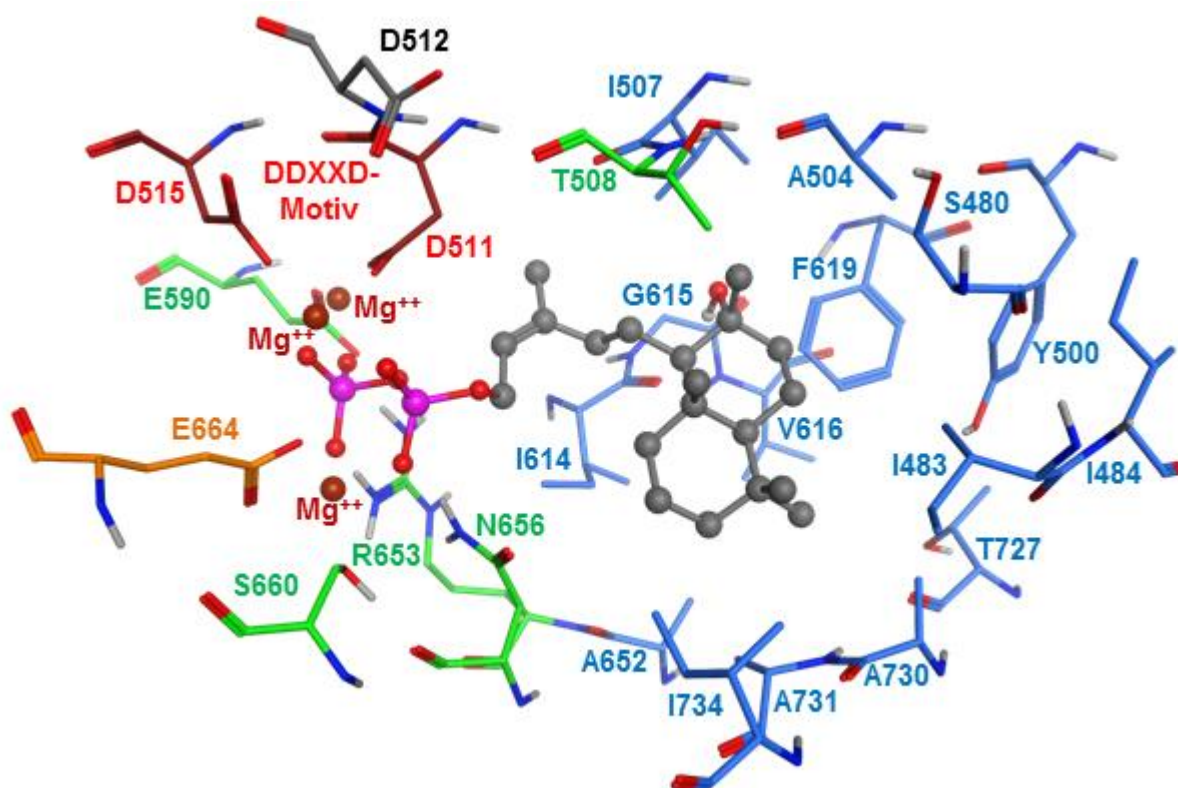


Abbildung 44: Bindetasche im NtABS-Modell mit 8-Hydroxy-*syn*-Copolyldiphosphat und Magnesiumionen. Die Farbkodierung ist in Abbildung 40 (Seite 74) definiert.

Die Darstellung der NtABS-Bindetasche in Abbildung 44 zeigt eine Komplexierung der Diphosphatgruppe, in welcher zwei Magnesiumionen jeweils zwischen α - und β -Phosphat positioniert sind und die Sauerstoffatome der beiden Phosphatgruppen quasi-ekliptisch angeordnet sind. Erwähnenswert ist die Lage der Seitenkette von N656: Während in den anderen Modellen ihre Ketogruppe an der Metallionkomplexierung beteiligt ist, wird diese Funktion hier durch das α -Phosphation übernommen. Zusätzlich nimmt im NtABS-Modell die Hydroxylgruppe von S660 an der Magnesiumerkennung teil. In den anderen drei erstellten KSL-Modellen übernimmt diese Funktion das durchgehend konservierte D657.

Die enzymatische Reaktion besteht in NtABS aus der ionischen Abspaltung des Diphosphatrestes, einer Hydridumlagerung und abschließender Deprotonierung mit Ausbildung eines vieratomigen π -konjugierten Systems (siehe Abbildung 15, Seite 30). Je nach Positionierung des intermediären Kations und des Diphosphats sind D511 und das Diphosphat selbst wahrscheinliche Kandidaten als Protonakzeptoren. Als hydrophile Wechselwirkung im hydrophoben Bereich der Substratbindetasche ist die Wasserstoffbrücke zwischen der Ketogruppe von I614 und der Hydroxylgruppe des Substrates zu erwähnen. Die restlichen Wechselwirkungen im hydrophoben Bereich der Bindetasche definieren vermutlich eine dreidimensionale Form, welche zur spezifischen Substraterkennung beiträgt oder sogar dafür notwendig ist. Die für Terpensynthasen relativ einfache Reaktion der NtABS ist möglicherweise nicht auf eine explizite Stabilisierung des intermediären Kations angewiesen, sobald die Diphosphathydrolyse stattgefunden hat.

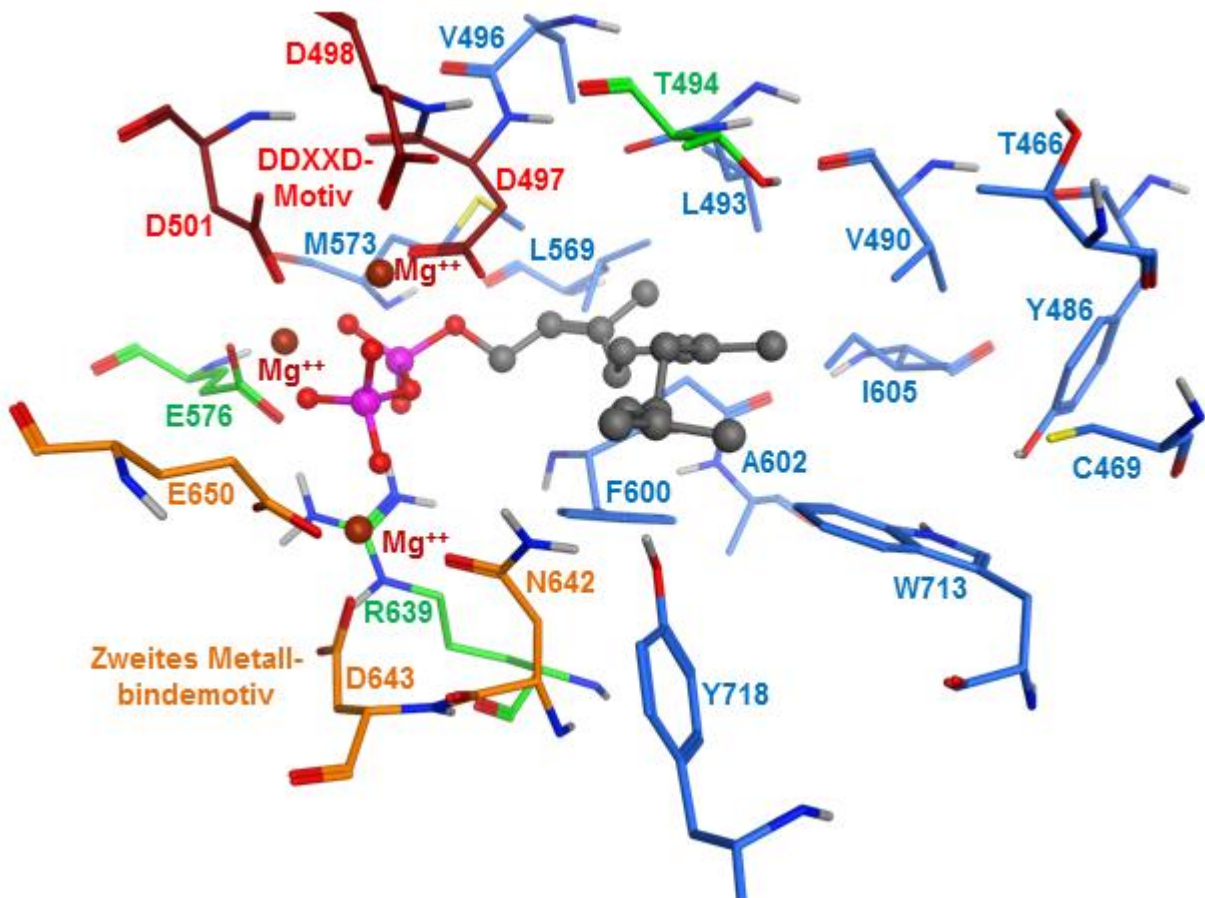


Abbildung 45: Bindetasche im ShSBS-Modell mit (Z,Z)-FPP und Magnesiumionen. Die Farbkodierung ist in Abbildung 40 (Seite 74) definiert.

Die Katalysen von ShSBS führen hingegen zu einer Fülle von Produkten (siehe Tabelle 20, Seite 89), die mit einem kraftfeldbasierten, starren Homologiemodell allein nicht zu erklären ist. Die Metallionenkomplexierung in diesem Modell (Abbildung 45) geht im Wesentlichen konform mit der Literatur.^{27,45} Der Diphosphatrest ist jedoch ähnlich zur Situation im AtKS-Modell gebunden: Das β -Phosphat wechselwirkt mit zwei Magnesiumionen und das dritte Magnesiumion liegt zwischen α - und β -Phosphat. Wie im AtKS-Modell ist auch hier aufgrund der methodischen Grenzen eine andere Lage des Diphosphatrestes zu den Metallionen denkbar. Die hydrophobe Bindetasche ist dominiert von den leicht zugänglichen aromatischen Systemen von F600 und W713, die potenziell intermediäre Kationen stabilisieren. Die Hydroxylgruppen der Aminosäuren T494 und Y718 sind zusätzlich in der Lage, elektrostatische Wechselwirkungen mit Intermediaten einzugehen.

Die Modulation der Produktbildung muss auch die Umsetzung von NPP zu beispielsweise Phellandren erlauben, so dass davon ausgegangen werden muss, dass der hydrophobe Teil der Bindetasche bei der Rekrutierung dieses kleineren Substrates teilweise kollabiert, sich mit Lösungsmittel auffüllt oder einige Seitenketten drastische Konformationsänderungen durchlaufen. Es ist unmöglich, aus dem vorliegenden Modell konkrete Hypothesen zu solchen Prozessen abzuleiten. Allerdings kann das Modell als Ausgangspunkt für weitere Untersuchungen gelten, deren Beschreibungskraft über Kraftfelder hinausgeht. Die Verwendung solcher Methoden ist notwendig, da die Reaktionsverläufe sehr fein moduliert werden (siehe hierzu auch 0, Seite 84 ff.). Dies zeigen insbesondere aufwändige quantenchemische Berechnungen zur Produktbildung in Terpensynthesen, in denen die Einflüsse der jeweiligen Enzyme vernachlässigt wurden.^{207,410} Eine Spekulation zum allgemeinen Mechanismus der Produktbildung ist die Stabilisierung von Intermediaten durch die aromatischen Aminosäuren F600 und W713 im unteren Teil von Abbildung 45. Die flexibleren Aminosäuren V490, T494, V496 und L569 im oberen Teil der Abbildung könnten entsprechende Konformationen von Intermediaten modulieren und so die Produktbildung beeinflussen.

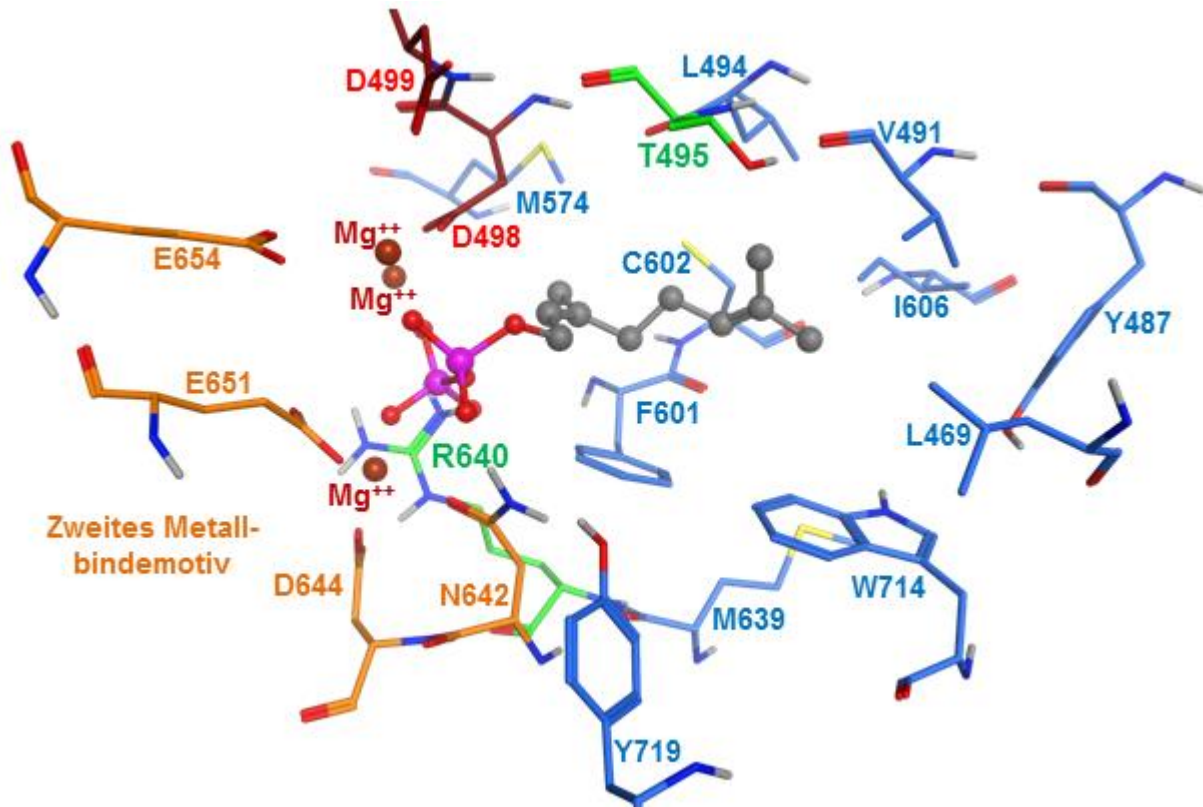


Abbildung 46: Bindetasche im SIPHS-Modell mit Neryldiphosphat und Magnesiumionen. Die Farbkodierung ist in Abbildung 40 (Seite 74) definiert.

Die Bindetasche des erstellten SIPHS-Modells ist in Abbildung 46 dargestellt. Die Komplexierung der Metallionen und des Diphosphatrestes entspricht dem Stand der Literatur.^{27,45} Wie in ShSBS sind auch hier eine Phenylalanin- und eine Tryptophanseitenkette Bestandteil des hydrophoben Teils der Bindetasche und theoretisch in der Lage, intermediäre Karbokationen zu stabilisieren. Die katalysierte Reaktion der Bildung von β -Phellandren aus Neryldiphosphat erfordert eine Konformation des Liganden, aus welcher die Zyklisierung zwischen dem allylischen Kation und der verbliebenen Doppelbindung des Prenylrestes geschehen kann. Wie beim ShSBS-Modell ist denkbar, dass die relativ geräumige Bindetasche mit Lösungsmittelmolekülen aufgefüllt ist oder Konformationsänderungen dazu führen, dass das Substrat vom Lösungsmittel abgeschirmt wird. Da Wassermoleküle mit dem intermediären Kation reagieren können ist davon auszugehen, dass entweder Wassermoleküle aus der Bindetasche ausgeschlossen werden oder ein zumindest teilweise konzertierter Prozess der Ionisierung und Zyklisierung des Substrates stattfindet, der den Angriff von Wassermolekülen ausschließt. Es ist wahrscheinlich, dass NPP in der Bindetasche eine die Zyklisierung bevorzugende Konformation einnimmt bevor die hydrolytische Abspaltung des Diphosphates von Statten geht um einen nukleophilen Angriff von Wasser zu verhindern. Bei der Umsetzung von GPP katalysiert SIPHS in kleinen Mengen die Bildung von Linalool, aber mit NPP als Substrat konnten keine Terpenalkohole im Produktspektrum nachgewiesen werden.¹⁰² Dies ist ein experimenteller Hinweis darauf, dass die Konformation des Substrates Einfluss auf die Zugänglichkeit der Bindetasche für Wassermoleküle hat.

4.2.2.3 Mutationsvorschläge für NtABS und ShSBS

Die Enzyme NtABS und ShSBS stehen im Mittelpunkt der Laborexperimente von Romy Töpfer am Leibniz-Institut für Pflanzenbiochemie in Halle (Saale). Eine konkrete vergleichende Analyse ist daher interessant, um im Licht von Aktivitätsuntersuchungen Unterschiede in Substrat- und Produktspektren auf Sequenzunterschiede herunterbrechen zu können und weiterführende Hypothesen abzuleiten. Beim Vergleich der Bindetaschen beider Enzyme bestehen wesentliche Unterschiede entsprechend Tabelle 16.

Tabelle 16: Im *Alignment* einander entsprechende Aminosäuren der Bindetaschen von NtABS und ShSBS. In Klammern ist die absolute Position in der jeweiligen kompletten Aminosäuresequenz der Enzyme angegeben.

NtABS	I483 (511)	A504 (532)	I614 (642)	V616 (644)	F619 (647)	A652 (680)	A730 (758)
ShSBS	C469 (502)	V490 (523)	F600 (633)	A602 (635)	I605 (638)	L638 (671)	W713 (746)

Die Differenz der Aminosäurepositionen in Tabelle 16 zwischen den *Alignment*-Positionen und den absoluten Positionen kommt durch Leerstellen im Sequenz-*Alignment* der KSL-Enzyme zustande.

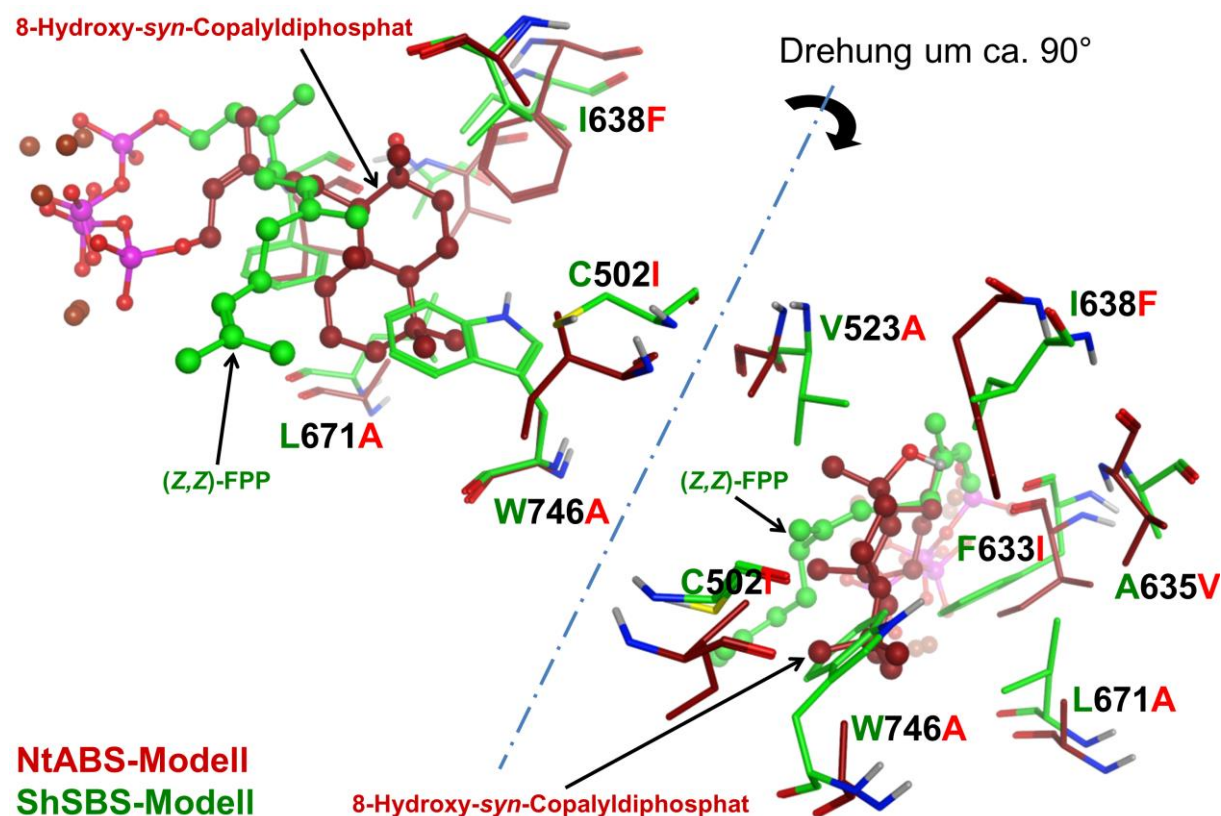


Abbildung 47: Ausschnittsdarstellung eines Struktur-*Alignment* zwischen NtABS- und ShSBS-Modell. Das Struktur-*Alignment* und die Abbildung wurden unter Benutzung von MOE erstellt. Die Ligandenatome und Metallionen sind durch Kugeln gekennzeichnet. Die Nummerierung der Aminosäuren entspricht den Absolutpositionen in der ShSBS-Sequenz.

Entsprechend des visuellen Vergleichs der Lage der Bindetaschen-Aminosäuren in Abbildung 47 wurden Mutationsvorschläge für ShSBS wie folgt rationalisiert (bezüglich Absolutpositionen in der ShSBS-Sequenz), siehe Tabelle 17.

Tabelle 17: Mutationsvorschläge für ShSBS und hypothetische Konsequenzen

W746A	größeres Bindetaschenvolumen, mindert Stabilisierung intermediärer Kationen
I638F	stärkere Stabilisierung intermediärer Kationen
F633I	größeres Bindetaschenvolumen, mindert Stabilisierung intermediärer Kationen
C502I	geringere Stabilisierung intermediärer Kationen
L671A	vergrößertes Bindetaschenvolumen

Vermutlich von untergeordneter Bedeutung, aber eventuell trotzdem in die Modulation der Produktbildung involviert sind die Positionen V523A und A635V, die interessanterweise paarweise vertauscht sind. Für eine schrittweise Umfunktionalisierung von ShSBS zur NtABS-Funktion bieten sich als Kcontroloption die entsprechenden Gegenmutationen in NtABS an. Solche gegenseitigen Umfunktionalisierungen wurden an verwandten Terpensynthasen bereits durchgeführt.⁴¹¹ Allerdings können auch Aminosäuren und Sekundärstrukturelemente in der Nähe der Bindetasche die Flexibilität und räumliche Kontur der an sich unreaktiven Aminosäuren des hydrophoben Teils der Terpensynthasebindetaschen wesentlich beeinflussen. Ohne weiterführende Veränderungen der Enzyme wäre in solchen Fällen eine beiderseitige Umfunktionalisierung nicht möglich.⁴¹²

4.2.3 Ergebnisse der experimentellen Umsetzung der Mutationsvorschläge

Romy Töpfer führte am Leibniz-Institut für Pflanzenbiochemie die vorgeschlagenen Mutationen an NtABS und ShSBS (Expressionssystem: *E. coli*) durch. Auszüge aus den Resultaten der anschließend von ihr ausgeführten Aktivitätstests (unveröffentlichte Ergebnisse) zeigen ein differenziertes Bild der Umsetzung verschiedener Substrate, wie die folgenden beiden Tabellen zusammenfassen. Die angegebenen optischen Aktivitäten wurden dabei nicht explizit überprüft, sondern entsprechend der Literatur entnommen.

Tabelle 18: Ergebnisse von Umsetzungsversuchen an ShSBS und ShSBS-Mutanten mit verschiedenen Substraten. Die erste Spalte benennt die verwendete Enzymmutante. In den weiteren Spalten sind nach Substraten getrennt die beobachteten Produkte von oben nach unten in absteigender Ausbeute angeordnet. Die Nomenklatur folgt Sallaud *et al.*,¹⁰¹. Die Zahlen in Klammern beziehen sich auf die Strukturdarstellungen in Abbildung 49 und Abbildung 50, Seite 87 f.

¹ A und B stehen für zwei nicht identifizierte Monoterpene.

² Diese Ergebnisse stammen jeweils aus Einfachmessungen von *E. coli*-Rohextrakten.

ShSBS	NPP	(Z,Z)-FPP	
		Rekombinantes Enzym	Transienter Test
Wildtyp	Limonen Terpinolen A ¹ B ¹ Linalool	(+)- α -Santalen (2) (+)- <i>endo</i> - β -Bergamoten (5) (-)- <i>endo</i> - α -Bergamoten (6) (-)- <i>exo</i> - α -Bergamoten (8) (-)- <i>epi</i> - β -Santalen (4) β -Farnesen β -Bisabolen β -Sesquiphellandren	(+)- α -Santalen (2) (+)- <i>endo</i> - β -Bergamoten (5) (-)- <i>exo</i> - α -Bergamoten (8) (-)- <i>endo</i> - α -Bergamoten (6) (-)- <i>epi</i> - β -Santalen (4) β -Farnesen β -Sesquiphellandren β -Bisabolen
C502I	Limonen Terpinolen A ¹	(+)- <i>endo</i> - β -Bergamoten (5) (-)- <i>endo</i> - α -Bergamoten (6) (+)- α -Santalen (2)	(+)- <i>endo</i> - β -Bergamoten (5) (-)- <i>exo</i> - α -Bergamoten (8) (+)- α -Santalen (2)

ShSBS	NPP	(Z,Z)-FPP	
		Rekombinantes Enzym	Transienter Test
	B ¹ Linalool	(-)- <i>exo</i> - α -Bergamoten (8) (-)- <i>epi</i> - β -Santalene (4) β -Farnesen β -Bisabolene β -Sesquiphellandrene	β -Sesquiphellandrene β -Bisabolene (-)- <i>endo</i> - α -Bergamoten (6) (-)- <i>epi</i> - β -Santalene (4) β -Farnesen
W746A	Limonene Linalool Terpineol	Keine Umsetzung	Farnesol β -Bisabolene β -Farnesen Sesquiphellandrene
F633I	nicht durchgeführt	β -Bisabolene ²	β -Bisabolene (+)- <i>endo</i> - β -Bergamoten (5) (-)- <i>exo</i> - α -Bergamoten (8) (-)- <i>epi</i> - β -Santalene (4) (+)- α -Santalene (2) (-)- <i>endo</i> - α -Bergamoten (6) β -Farnesen β -Sesquiphellandrene
I638F	nicht durchgeführt	β -Bisabolene ²	(-)- <i>exo</i> - α -Bergamoten (8) (+)- α -Santalene (2) (-)- <i>endo</i> - α -Bergamoten (6) (+)- <i>endo</i> - β -Bergamoten (5) β -Bisabolene β -Farnesen β -Sesquiphellandrene (-)- <i>epi</i> - β -Santalene (4)
L671A	nicht durchgeführt	β -Bisabolene ²	(+)- α -Santalene (2) (+)- <i>endo</i> - β -Bergamoten (5) β -Farnesen (-)- <i>endo</i> - α -Bergamoten (6) (-)- <i>exo</i> - α -Bergamoten (8) (-)- <i>epi</i> - β -Santalene (4) β -Bisabolene β -Sesquiphellandrene

Die transienten Enzymaktivitätstests an ShSBS wurden zweifach durchgeführt und die in Tabelle 18 aufgeführten Ergebnisse wurden reproduziert. Die Unterschiede zwischen den Resultaten der Aktivitätstests mit rekombinantem Enzym aus *E. coli* und aus dem transienten Test mit *N. benthamiana* können verschiedene Ursachen haben. Beispielsweise können Interferenzen mit anderen Enzymen des jeweiligen Expressionsorganismus auftreten. Ferner ist es möglich, dass organismusspezifische Glykosylierungsmuster Enzymaktivitäten modulieren. Grundsätzlich sind die Ergebnisse der transienten Tests vertrauenswürdiger, weil sie im pflanzlichen Expressionssystem zustande kamen und reproduziert wurden. Entsprechende Negativkontrollen der transienten Tests mit leerem Vektor belegen die Verantwortlichkeit der entsprechend klonierten Gene für die Anreicherung der Substanzen, die in Tabelle 8 aufgeführt sind.

Weitere Versuche erbrachten, dass alle ShSBS-Varianten keine Umsetzung von GGPP katalysieren. Das Wildtyp-Enzym NtABS ist weder in der Lage (Z,Z)-FPP noch (E,E)-FPP zu Sesquiterpenen umzusetzen. Weitere Ergebnisse zur Umsetzung von NPP und 8-Hydroxy-*syn*-CPP mit NtABS sind Tabelle 19 zu entnehmen. Diese Ergebnisse werden im Abschnitt 4.2.5 (Seite 90ff.) gemeinsam mit den theoretischen Betrachtungen zu den Enzymkatalysen diskutiert.

Tabelle 19: Ergebnisse von Umsetzungsversuchen an NtABS und NtABS-Mutanten mit verschiedenen Substraten. Bei der Umsetzung mit 8-Hydroxy-*syn*-CPP wurde ein gekoppelter Enzymtest mit weiteren Enzymen benutzt, um das Substrat bereitzustellen.

¹ B steht für ein nicht identifiziertes Monoterpen.

NtABS	NPP	8-Hydroxy- <i>syn</i> -CPP
Wildtyp	Limonen Nerol Terpineol B ¹ Linalool β-Myrcen Terpinolen	<i>cis</i> -Abienol
I486C	nicht durchgeführt	<i>cis</i> -Abienol <i>trans</i> -Abienol
F621I	nicht durchgeführt	<i>cis</i> -Abienol <i>trans</i> -Abienol
A654L	nicht durchgeführt	<i>cis</i> -Abienol <i>trans</i> -Abienol
A732W	nicht durchgeführt	<i>cis</i> -Abienol
I616F	nicht durchgeführt	Keine Umsetzung

4.2.4 Quantenchemische Berechnungen zur Bildung von Santalen und Bergamoten

Berechnungen molekularer Geometrien mit *ab initio*-Methoden sind derzeit wegen des hohen Rechenaufwandes auf kleine Molekülszenarien begrenzt. In den hier angestellten Untersuchungen wurden daher weder Lösungsmittelmoleküle noch Enzyme modelliert. Allerdings können anhand von Berechnungen an isolierten Gasphasemolekülen oft trotzdem Aussagen darüber getroffen werden, welche Funktionen das Enzym übernehmen muss, um die Produktbildung zu leiten. Im Rahmen dieser Arbeit wurden quantenchemische Berechnungen zur Bildung der ShSBS-Produkte und zu den Wechselwirkungen zwischen Prenylkationen und ausgewählten Aminosäureseitenketten durchgeführt.

Jones und Kollegen schlugen 2006 Reaktionsmechanismen für die Bildung von Bergamotenen und Santalenen aus (*E,E*)-FPP vor.³⁹⁷ Analog kann die Bildung dieser Substanzen auch aus (*Z,Z*)-FPP erfolgen, wobei formal die initiale Umlagerung des Diphosphatrestes und die Isomerisierung der Prenylkette im (*E,E*)-FPP-Fall durch die ionische Abspaltung des Diphosphatrestes im Fall von (*Z,Z*)-FPP als Substrat ersetzt werden können. Die hier untersuchten Abläufe orientieren sich an den Ausführungen zur (*Z,Z*)-FPP-Umsetzung in der Publikation von Sallaud *et al.*,¹⁰¹.

4.2.4.1 Erste Zyklisierung

Die Enantiomere des Bisabolylkations (Abbildung 48) gelten in Sesquiterpensynthesen entsprechend dem α-Terpinylkation in Monoterpensynthesen als Ausgangspunkt für weitere Transformationen im Reaktionsverlauf.

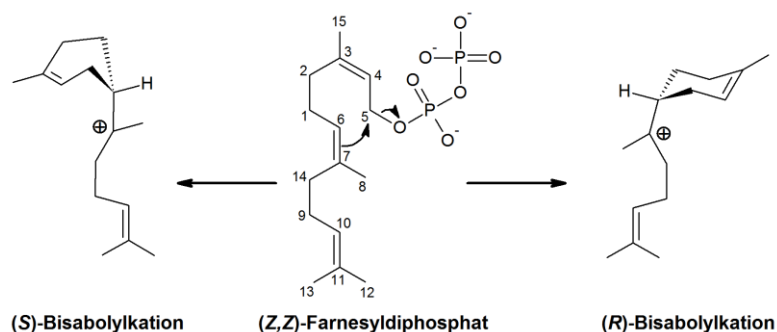


Abbildung 48: Mögliche erste Schritte in der enzymatischen Produktion von Santalenen und Bergamotenen. Die Nummerierung der Kohlenstoffatome entspricht ¹⁰¹. Der Diphosphatrest verbleibt im Lauf der Enzymkatalyse vermutlich in der Bindetasche des Enzyms.

Für Terpensynthesen gilt die Hypothese, dass die enzymgebundene Konformation des Substratmoleküls entscheidenden Einfluss auf die Produktbildung hat.^{34,413,414} Dies kann ausgehend vom (Z,Z)-FPP sowohl zum (R)- als auch zum (S)-Bisabolylkation führen. Die im Rahmen dieser Arbeit durchgeführten Berechnungen lieferten mit entsprechenden Startkonformationen die erwarteten Bisabolylkationenantimere als Zwischenzustände. Eine umfangreiche theoretische Studie zur Bildung von Santalenen und Bergamotenen, die nach der Durchführung der Berechnungen in dieser Arbeit veröffentlicht wurde, postuliert, dass auch die nachträgliche Epimerisierung zwischen den beiden Enantiomeren möglich sein könnte: Die Energiebarriere für eine 6,7-Hydridumlagerung ist kleiner als 20 kJ/mol; die Rotationsbarriere gilt als noch geringer.⁴¹⁰ Die ausgehend von den Bisabolylkationen in den folgenden Reaktionsschemata angeführten 6,7-Rotationen sind für die Bildung der jeweiligen Reaktionsprodukte nicht unbedingt notwendig, aber trotzdem relevant, da bei Bindung des Substrates in der Bindetasche korrespondierende Konformationen vorgebildet sein können.

4.2.4.2 Folgezyklisierungen

Die berechneten flexiblen Koordinaten-Scans der 7,3-Zyklisierung des (R)-Bisabolylkations zeigen in der abschließenden Optimierung ohne Nebenbedingung, dass die dargestellte 4,3-4,2-Umlagerung als Anschluss energetisch favorisiert ist (die beiden rechten Pfade in Abbildung 49, Seite 86). Dies umgeht die Bildung sekundärer Kationen, was die Ergebnisse früherer theoretischer Untersuchungen bestätigt.²⁰⁵

Die Bildung der α -Santalene **1** und **2** (links in Abbildung 49) konnte durch die Benutzung instabiler Intermediatstrukturen als Ausgangspunkt für flexible Koordinaten-Scans simuliert werden. Die dreigliedrigen Ringe kamen hierbei jeweils unter Protonabstraktion durch ein manuell positioniertes Essigsäureanion zu Stande, was auf einen konzertierten Mechanismus der Zyklisierung und Deprotonierung hinweist.

Ein Nachteil der durchgeführten flexiblen Koordinaten-Scans ist, dass die modellierten Reaktionen entlang des vorbestimmten Pfades gezwungen werden. So besteht die Möglichkeit mechanistische Details, welche sich mit mächtigeren Methoden offenbaren oder welche konformationsabhängig ablaufen, zu übersehen.

Die Frage, ob in der Bildung der Santalene wirklich eine initiale 7,3-Zyklisierung stattfindet, haben Hong und Tantillo mit der Berechnung intrinsischer Reaktionskoordinaten in der Gasphase adressiert.²⁰⁷ Sie kommen zu dem Schluss, dass subtile Unterschiede in der Enzymumgebung der ablaufenden Reaktion Einfluss auf die Stabilität eines Übergangszustandes

haben können, der nach einer initialen 7,2-Zyklisierung nachfolgend die Bildung von Santalenen und Bergamotenen erlaubt.

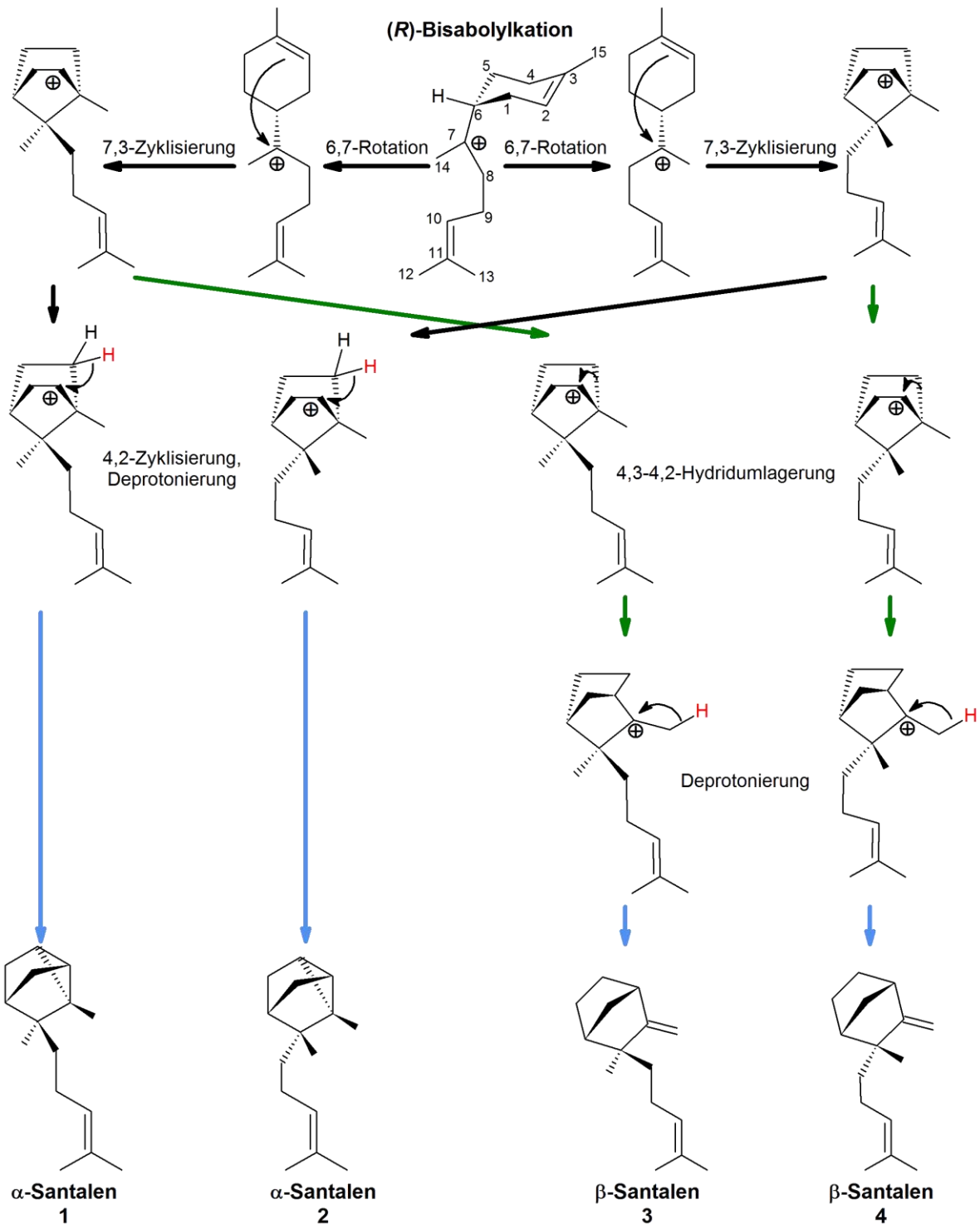


Abbildung 49: Theoretische formale Reaktionswege zu α - und β -Santalenen. Die Protonen der rot hervorgehobenen Wasserstoffatome werden abstrahiert. Grüne Pfeile: Schritte liefen in der Modellierung anschließend an den jeweils vorherigen Schritt spontan ab. Blaue Pfeile: Ablauf unter entsprechender Positionierung eines Essigsäureanions; Nomenklatur siehe Tabelle 20, Seite 89

Jindal und Sunoj schlagen folgende leicht abweichende Mechanistik für die Bildung der Santalene vor: Anstelle einer initialen 7,3-Zyklisierung des Bisabolylkations findet für Santalene und Bergamotene gleichermaßen eine 7,2-Zyklisierung statt, der im Falle der Bildung von Santalenen eine 7,2-7,3-Umlagerung folgt.⁴¹⁰ Sie postulieren ferner eine dyotrope Kopplung dieser 7,2-7,3-Umlagerung mit der nachfolgenden 4,3-4,2-Umlagerung, die sich auch in den hier durchgeführten Berechnungen als favorisiert darstellte.

Durch den spontanen Ablauf der 4,3-4,2-Hydridumlagerung, die in Abbildung 49 auf die 7,3-Zyklisierung folgt und zur Vorbildung der β -Santalene (ausgenommen Protonabstraktion) führt, ist davon auszugehen, dass ShSBS die Bildung von α -Santalenen aktiv über die Reaktionseinleitung hinaus beeinflusst. Dies kann zum Beispiel durch eine Positionierung des 7,3-zyklisierten Intermediates geschehen, die eine Protonabstraktion zur Folge hat, wie sie in der linken Hälfte von Abbildung 49 dargestellt ist. Ein solcher Protonentzug führt möglicherweise konzertiert mit der 4,2-Zyklisierung zur Bildung der α -Santalene.

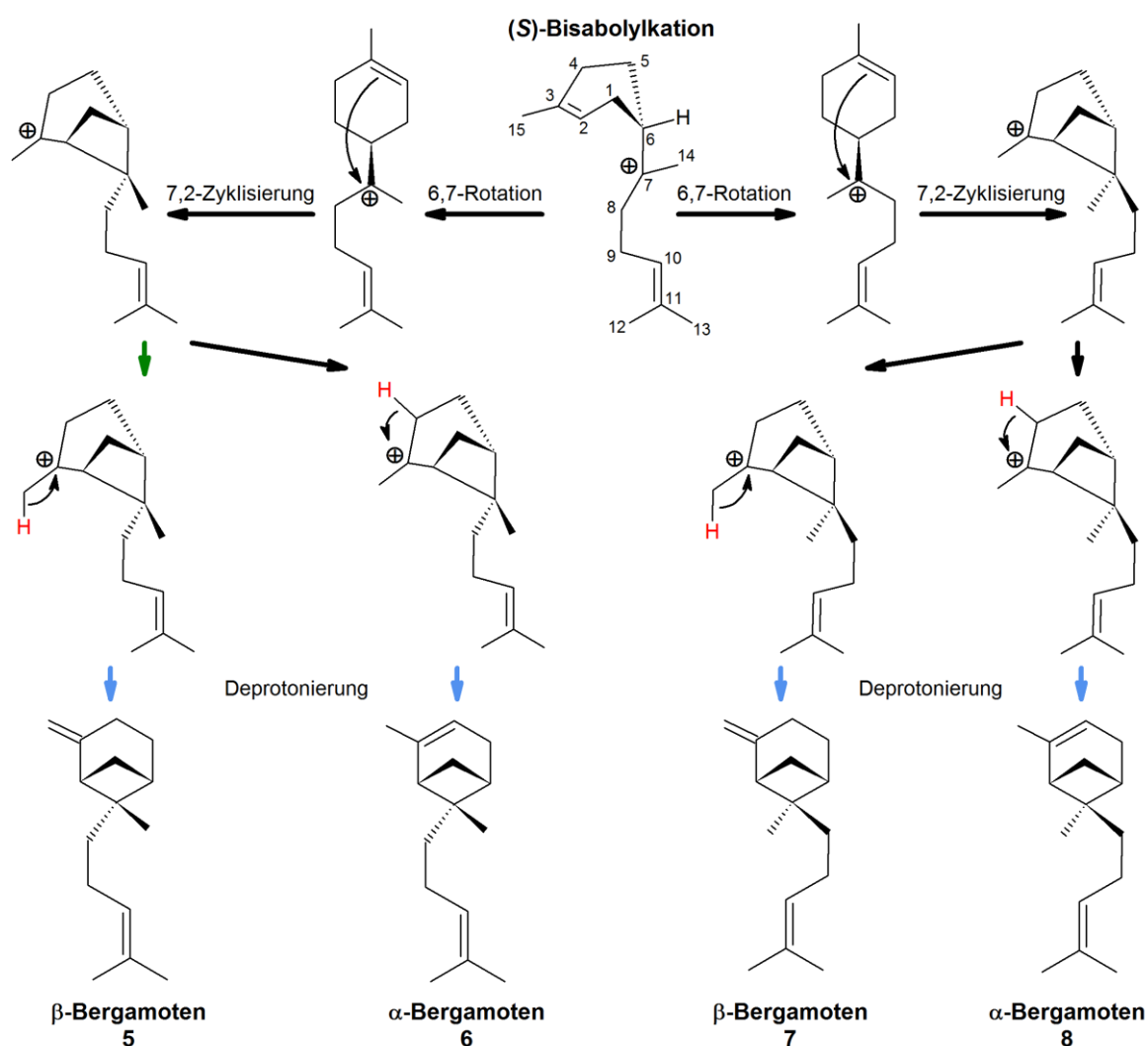


Abbildung 50: Theoretische formale Reaktionswege zu α - und β -Bergamotenen. Die Protonen der jeweils rot hervorgehobenen Wasserstoffatome werden abstrahiert. Grüner Pfeil: Schritt lief in der Modellierung anschließend an den vorherigen Schritt spontan ab. Blaue Pfeile: Ablauf unter entsprechender Positionierung eines Essigsäureanions; Nomenklatur siehe Tabelle 20, Seite 89

Da jedoch die β -Santalene ebenfalls abschließend deprotoniert werden, sind hier vermutlich sehr geringe relative Konformationsunterschiede zwischen Intermediat und Substratbindetasche für die Differenzierung zwischen α - und β -Santalenen verantwortlich.

Die flexiblen Koordinaten-Scans zur Bildung der Bergamotenprodukte lieferten nach der 7,2-Zyklisierung Intermediate, die analog zur Santalenebildung in Abhängigkeit einer entsprechend positionierten Base (Essigsäureanion) zu **5-8** deprotonierten (siehe Abbildung 50 und Tabelle 20, Seite 89). Auch hier muss eine relativ geringfügige Änderung der Prädisposition zur Protonabstraktion für die Differenzierung zwischen den verschiedenen Produkten verantwortlich sein. Die durch die 7,2-Zyklisierung gebildeten Intermediate unterlaufen allerdings im Gegensatz zu den Santalenen keine weiteren Umlagerungen und Zyklisierungen. Die Bildung der Bergamotene wird demnach ausschließlich vom Angriff zur Protonabstraktion geleitet.

ShSBS katalysiert hauptsächlich die Bildung von **2, 4, 5, 6** und **8** aus (*Z,Z*)-FPP.¹⁰¹ Bei **2** und **4** tritt das C6-Atom hierbei in *R*-Konfiguration auf, bei **5, 6** und **8** in *S*-Konfiguration. Das C6-Atom ist nach der 6,1-Zyklisierung zum (*R*)- oder (*S*)-Bisabolykation am weiteren Reaktionsverlauf nicht beteiligt. Ausgehend von der Hypothese, dass das Enzym nur eine bestimmte Konformation (entweder prä-*S* oder prä-*R*) in der Bindetasche akzeptiert, stellt sich die Frage, wie **2** und **4** spezifisch gegenüber **1** und **3** zu Stande kommen. Analog kann man fragen, warum die Produkte **5, 6** und **8**, aber nicht **7** gebildet werden. Die berechneten quantenmechanischen Energien zeigen deutliche Unterschiede zwischen α - und β -Santalenen und Bergamotenen, können aber die Bildung von **2, 4, 5, 6** und **8** gegenüber **1, 3** und **7** nicht erklären (siehe Abbildung 51 A). Ohne die Berechnung von intrinsischen Reaktionskoordinaten und die Ermittlung von Übergangszuständen und Energieprofilen sind diese Fragen somit schwer zu beantworten. Sowohl Hong und Tantillo⁽²⁰⁷⁾ als auch Jindal und Sunoj⁽⁴¹⁰⁾ bieten mit derartigen Modellierungen Einblicke in Gasphasen-Reaktionsverläufe. Letztere kommen zu dem Schluss, dass Produktspektren mit Reaktionsprodukten aus dem (*R*)- und dem (*S*)-Bisabolykationweg wahrscheinlich durch die bereits geschilderte Epimerisierung zu Stande kommen. Eine Differenzierung innerhalb der Gruppen muss demnach durch Einwirkung des Enzyms geschehen.

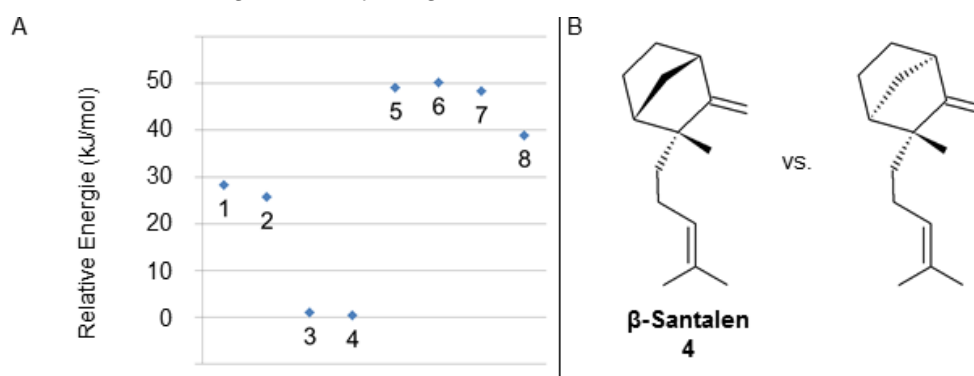


Abbildung 51: A) Relative quantenmechanische Energien der betrachteten Santalene und Bergamotene, B) Epimer von **4** („(-)- β -Santalene“ nach Jindal und Sunoj,⁴¹⁰)

Es ist hervorhebenswert, dass die quantenmechanischen Energien der β -Santalene **3** und **4** niedriger liegen als die der α -Santalene **1** und **2**. Die β -Santalene kamen in den Gasphasenberechnungen nach der 7,3-Zyklisierung spontan zustande (abgesehen von der Protonabstraktion). Dass ShSBS trotzdem überwiegend α -Santalene produziert, lässt darauf schließen, dass die Intermediate nach Deprotonierung nicht rückreagieren, weil sonst die

thermodynamisch bevorzugten β -Santalene gebildet würden. Vermutlich ist demnach die Deprotonierung auch mit Konformationsänderungen der Bindetasche verbunden, welche die entsprechende Rückreaktion verhindern.

Für die Betrachtung der Zyklisierungsmöglichkeiten ist es unabdingbar, auf Unterschiede bzw. Unvollständigkeiten und Inkonsistenzen bei der Nomenklatur der Santalene und Bergamotene aufmerksam zu machen:

Tabelle 20: Die Nomenklatur von Santalenen und Bergamotenen variiert in der Literatur; Spalte B: Bezeichnung der Strukturen in dieser Arbeit (vergleiche Abbildung 49 und Abbildung 50, Seite 86 bzw. 87). Die Substanzen 2, 3, 4, 5 und 8 sind die Hauptprodukte der Umsetzung von (Z,Z)-FPP mit ShSBS.¹⁰¹

B	Sallaud <i>et al.</i> ¹⁰¹	Hong & Tantillo ²⁰⁷	Jindal & Sunoj ⁴¹⁰	andere
1	-	<i>ent</i> - α -Santalene	-	-
2	(+)- α -Santalene	α -Santalene	(+)- α -Santalene	α -Santalene ³⁹⁷ (+)- α -Santalene ²⁵⁴
3	-	β -Santalene	(+)- β -Santalene	-
4	(-)- <i>epi</i> - β -Santalene	<i>epi</i> - β -Santalene	(+)- <i>epi</i> - β -Santalene	<i>epi</i> - β -Santalene ³⁹⁷ (+)- <i>epi</i> - β -Santalene ²⁵⁴
5	(+)- <i>endo</i> - β -Bergamotene	-	(+)- <i>endo</i> - β -Bergamotene	-
6	(-)- <i>endo</i> - α -Bergamotene	-	(-)- <i>endo</i> - α -Bergamotene	<i>endo</i> - α -Bergamotene ²⁵⁴
7	-	-	-	β - <i>trans</i> -Bergamotene ⁴¹⁵ α -Bergamotene ³⁹⁷
8	(-)- <i>exo</i> - α -Bergamotene	-	(-)- <i>exo</i> - α -Bergamotene	α - <i>trans</i> -Bergamotene ⁴¹⁵ <i>trans</i> - α -Bergamotene ⁴¹⁶ (-)- α - <i>exo</i> -Bergamotene ²⁵⁴

Zusätzlich zu den in Tabelle 20 dargestellten Nomenklaturen existieren weitere Stereoisomere von Bergamotenen und β -Santalenen, die zum Teil bei Hong und Tantillo als *trans*- α -Bergamotene, *cis*- α -Bergamotene, *trans*- β -Bergamotene und *cis*- β -Bergamotene bezeichnet werden.²⁰⁷ Jindal und Sunoj zeigen (-)-*epi*- β -Santalene, welches wie (-)- β -Santalene auch bei Jones *et al.* vorkommt (vgl. Abbildung 51 B).²⁵⁴ Es wird in diesen Publikationen nicht deutlich, ob es sich um natürlich vorkommende Isomere handelt. Sie blieben im Rahmen dieser Arbeit unberücksichtigt. In allen diesen Fällen liegt das den 6-Ring überbrückende Kohlenstoffatom auf der anderen Seite des Ringes als bei **3-8**. Consoli *et al.* berichten eine langsame, temperatur- oder lichtabhängige Isomerisierung von **7** zu **8**.⁴¹⁵ Das trifft vermutlich analog auf **5** und **6** zu.

4.2.4.3 Abschließende Deprotonierung

Die finale Deprotonierung der exozyklischen Methylgruppe zu den β -Formen bzw. die endozyklische Deprotonierung zu den α -Formen der Reaktionsprodukte konnte durch manuelle Positionierung eines Essigsäureanions jeweils simuliert werden und führte zu den acht angeführten Produkten (Tabelle 20). Es gibt verschiedene Theorien, welche Base das Proton im letzten Teilschritt abstrahiert:

- eine basische Aminosäureseitenkette (direkt oder über eine katalytische Diade)³⁶⁷
- ein Lösungsmittelmolekül³¹
- der Diphosphatrest^{30,410}

Dass das verwendete Essigsäureanion bei sinnvoller Positionierung jeweils die finale Deprotonierung durchführen kann, ist nicht überraschend und liefert ohne Betrachtung der Enzymstruktur auch keine neuen Erkenntnisse über die Verhältnisse im Enzym. Interessanterweise zeigten Garms und Kollegen an Sesquiterpensynthasen, dass die Deprotonierung auch durch Austausch hydrophober Aminosäuren in der Bindetasche – in ihrem Beispiel Leucin → Valin – beeinflusst werden kann und so das Produktspektrum moduliert wird.⁴⁰⁹

4.2.5 Integrierte Zusammenfassung und Diskussion der theoretischen und experimentellen Ergebnisse zu KSL-Enzymen

Methoden der Molekülmodellierung erlauben Einblicke in molekulare Strukturen und Prozesse, die mit Laborexperimenten nur schwer oder gar nicht zu erhalten sind. Die präsentierten Homologiemodelle sind ein Beispiel dafür und wurden als Ausgangspunkte benutzt, um die entsprechenden Biokatalysemechanismen zu plausibilisieren und zielgerichtete Mutagenesen vorzuschlagen. Die große Sequenzhomologie zu bekannten Proteinstrukturen und das vorhandene Wissen über Terpensynthasen rechtfertigen ein hohes Vertrauen in die erstellten Modelle.^{6,417} Die Homologiemodelle waren von großem Nutzen, um rationale Mutationsvorschläge für NtABS und ShSBS zu entwickeln. Die einzelnen zielgerichteten Mutationen führten zum Teil zu Veränderungen des Produkt- bzw. Substratspektrums und erlauben eine Diskussion der bisher erzielten Effekte. Die jeweiligen Mutationen entsprechen im Folgenden der Nomenklatur von Abschnitt 4.2.3. Die Diskussion der ShSBS-Mutanten bezieht sich vorrangig auf die reproduzierten Ergebnisse des transienten Enzymtests, weil diese als verlässlicher beurteilt werden als die Resultate der Rohextraktmessungen.

Die größte Veränderung auf die Enzymkatalyse bringt die Mutation des Tryptophan in der ShSBS-Bindetasche zu Alanin (W746A). Die Folge ist in ShSBS ein kompletter Verlust der Aktivität sekundärer Zyklisierung bei Umsetzung von (*Z,Z*)-FPP. Dies untermauert die Hypothese der Wichtigkeit dieser aromatischen Aminosäure für die Stabilisierung kationischer Übergangszustände. Interessanterweise produziert das mutierte Enzym mit NPP noch die Monoterpene Limonen und Linalool wie beim Wildtyp, aber zusätzlich auch Terpeneol als Monoterpenalkohol. Dazu korrespondierend produziert diese Mutante mit (*Z,Z*)-FPP Farnesol. Dies lässt den Schluss zu, dass das vergrößerte Bindetaschenvolumen das verstärkte Eindringen von Wasser erlaubt oder eventuell vorhandenen Wassermolekülen mehr Freiheitsgrade bietet. Trotzdem bildet die Enzymmutante mit (*Z,Z*)-FPP noch die einfach zyklisierten Produkte β -Bisabolen und Sesquiphellandren.

Die Mutation C502I in ShSBS führt nicht zur Änderung der Produktspektren, aber zu einer geringeren Produktion von (+)- α -Santalol. Die Aminosäureseitenkette an dieser Stelle hat somit offenbar einen gewissen Einfluss auf die Zugänglichkeit diverser Substratkonformationen, welche zu verschiedenen Produkten führen. Für eine konkrete Diskussion der Funktion von C502 in ShSBS sind jedoch weitere Untersuchungen zur Produktbildung notwendig. Es müsste beispielsweise durch Mehrfachmutationen ausschließbar sein, dass die potenzielle Funktion von C502 in der Isoleucin-Mutante durch eine benachbarte Aminosäure übernommen wird.

Interessant ist der Effekt der Mutationen F633I, I638F und L671A in ShSBS, der bei Umsetzung mit (*Z,Z*)-FPP das Produktspektrum in den *E. coli*-Rohextrakten komplett zu β -

Bisabolen kanalisiert. Es ist allerdings möglich, dass die Bildung des jeweils nachgewiesenen β -Bisabolens durch Enzyme aus *E. coli* katalysiert wird und die durchgeführten Mutationen ShSBS inaktivieren oder auf eine Weise wirken, deren Konsequenzen durch *E. coli*-Enzyme überlagert werden. Falls *E. coli*-Enzyme keinen Einfluss auf diese Ergebnisse haben, würde die gemessene Produktbildung darauf hindeuten, dass jede einzelne der entsprechenden Aminosäuren im Wildtyp dazu beiträgt, Übergangszustände zu stabilisieren oder in Konformation so zu positionieren, dass sie durch andere Aminosäuren stabilisiert werden. Die Bildung des Bisabolylkations ist der erste Schritt in der enzymatischen Bildung von Santalenen und Bergamotenen (siehe Abschnitt 0, Seite 84). Der konsequente Reaktionsabbruch, der dann offenbar in diesen Mutanten aufträte, kann theoretisch nur dadurch zu Stande kommen, dass das intermediäre Bisabolylkation eine Konformation bzw. relative Lage in der Bindetasche einnimmt, die eine frühzeitige Deprotonierung zur Folge hat. Diese Hypothese lässt sich auch für die Ergebnisse des transienten Tests nachvollziehen, aus dem β -Bisabolen als Hauptprodukt resultiert. Die untergeordnete Bildung von Bergamotenen und Santalenen relativiert allerdings die Stärke des vermuteten Effekts. Konkreten Aufschluss über stabilisierende oder sterische Rollen der einzelnen Aminosäuren könnten Doppelmутanten und anschließende Umsetzungsversuche an den aufgereinigten Enzymen geben.

Die Mutationen I486C, F621I und A654L an NtABS führen zu höheren Ausbeuten von *cis*-Abienol als im Wildtypenzym und führen gleichzeitig zum Nachweis von *trans*-Abienol als Nebenprodukt. Während I486C und F621I den für das Substrat zugänglichen Konformationsraum in der Bindetasche vergrößern sollten, müsste A654L einen gegenteiligen Effekt haben. Die wahrscheinlichste Ursache für höhere Ausbeuten ist eine effektivere Zugänglichkeit intermediärer Spezies zu Konformationen, die zur Produktbildung führen. Möglicherweise bewirken alle drei Mutationen einen solchen Effekt. Die scheinbar enzymatische Bildung von *trans*-Abienol ist auf die Isomerisierung von *cis*- zu *trans*-Abienol im Prozess der Gaschromatographieanalytik zurückzuführen.⁴¹⁸

Die Mutation A732W in NtABS hat lediglich einen geringen Effekt auf die Umsetzung von 8-Hydroxy-*ent*-CPP zu *cis*-Abienol. Dies ist erstaunlich, weil allein die räumliche Ausdehnung der Seitenkette des Tryptophans wesentlich größer ist als die des Alanins. Die überlagerten Bindetaschen in Abbildung 47 (Seite 81) zeigen auch eine sterische Überlagerung des Tryptophan mit der Position des 8-Hydroxy-*ent*-CPP. Ein möglicher Schluss besteht darin, dass es zu Konformationsänderungen in der NtABS-Mutante A732W kommt, welche die Substratbindung weiterhin erlauben und die Katalyse nicht wesentlich beeinflussen.

Der Austausch I616F in NtABS bewirkt, dass 8-Hydroxy-*ent*-CPP nicht mehr umgesetzt wird. Diese Wirkung des großen, relativ starren Phenylalanin statt eines kleineren und flexibleren Isoleucin lässt sich hypothetisch einfach begründen. Offenbar verhindert der Phenylring die Enzymkatalyse komplett. Die wahrscheinlichste Erklärung dafür ist, dass der Ligand davon abgehalten wird, eine Konformation relativ zur Bindetasche einzunehmen, aus welcher die Reaktion eingeleitet werden kann.

Die durchgeführten Aktivitätstests bieten zwar kein klares Bild aller Funktionen der variierten Aminosäuren oder der Modulation der Produktspezifität. Allerdings tragen diese Ergebnisse zum Verständnis der involvierten Mechanismen bei und können als sehr gute Grundlage für weitere Experimente gelten. Zukünftige Laborversuche könnten die Herstellung und Testung von Enzymchimären aus NtABS und ShSBS umfassen, aber auch die Untersuchung von Mehrfachmutanten ist vielversprechend.

Die durchgeführten quantenchemischen Berechnungen liefern deutliche Hinweise darauf, dass die Produktbildung in KSL-Enzymen sehr fein durch Rezeptor-Ligand-Wechselwirkungen moduliert wird. Folgende Schlussfolgerungen wurden abgeleitet:

- Sekundäre Karbokationen werden durch konzertierte Mechanismen bzw. spontane Umlagerungen als Intermediate weitgehend vermieden.
- Wechselwirkungen zwischen intermediären Karbokationen und Aminosäureseitenketten können energetisch im Bereich der Energie von Wasserstoffbrückenbindungen und darüber liegen und während der Enzymkatalyse Einfluss auf die Produktbildung haben.
- Die relative Positionierung von Intermediaten in der Bindetasche hat direkte Auswirkung auf die Produktdetermination. Besonders im Fall der Bergamotenbildung durch ShSBS wird deutlich, dass die Richtung des nukleophilen Angriffs zur Deprotonierung die Produktbildung stark beeinflusst.
- Eine rein thermodynamische Kontrolle der Produktbildung ist auszuschließen. Dies ist zum Beispiel erkennbar an der offenbar enzymbeeinflussten Bildung der thermodynamisch nicht bevorzugten α -Santalene nach der 7,3-Zyklisierung des (*R*)-Bisabolylkations durch ShSBS.

Die Aussagekraft der verwendeten quantenmechanischen Methoden ist hoch: Jindal und Sunoj rechnen zwar in der Gasphase, verwenden aber Diethylether als dielektrisches Kontinuum zur Modellierung der Enzymumgebung. Sie finden die gleichen relativen Aussagen wie bei Berechnungen in der Gasphase.⁴¹⁰ Allerdings schlagen sie einen leicht anderen Verlauf der Enzymkatalyse für Santalene vor. Ferner führen sie Untersuchungen zur Variabilität der Ergebnisse unter Benutzung verschiedener Dichtefunktionale an und beschreiben eine generelle Ähnlichkeit der Resultate. Hong und Tantillo bestätigen, dass verschiedene Theorieebenen zwar unterschiedliche absolute Energien liefern, aber die qualitativen Aussagen gleich bleiben und resultierende Geometrien sich stark ähneln.²⁰⁷ Weniger aufwändige semiempirische Methoden wurden bereits in Kombination mit Kraftfeldern verwendet, um die Zyklisierung in Terpensynthesen zu modellieren.⁸⁹ Wie jedoch von Hong und Tantillo und auch von Jindal und Sunoj gezeigt wurde, ist die Energiehyperfläche des Reaktionsverlaufs bei der Bildung von Santalenen und Bergamotenen sehr empfindlich gegenüber chemischen Einflüssen und bereits kleine Unterschiede in der Stabilisierung kationischer Zwischenzustände können zu verschiedenen Produkten führen. Eine sehr präzise Modellierung der beteiligten Wechselwirkungen zwischen Karbokation und umgebenden Aminosäuren ist vonnöten, um Reaktionsverläufe unter Einbezug der Enzymstruktur oder Teile derer zu untersuchen. Die zum jetzigen Zeitpunkt verfügbaren Rechner und Rechenprogramme sind von einer *ab initio*-Modellierung einer solchen Szenerie im Allgemeinen überfordert. Kombinierte Quanten- und Molekülmechanikansätze (QM/MM) wie von Rajamani und Gao (⁸⁹) zur Aufklärung von Katalysemechanismen in Terpensynthesen sind der momentan vielversprechendste Ansatz, um auf theoretischer Ebene neue Erkenntnisse zu gewinnen. Aufgrund der Größe der Substratbindetaschen von Sesquiterpensynthesen ist auch diese Methodik herausgefordert, die beteiligten Prozesse mit unterschiedlichen Verfahren mit der nötigen Genauigkeit bei möglichst geringer Rechen-dauer zu modellieren.

4.3 Molekulares *Docking* von Reaktionsintermediaten in *Mentha spicata* Limonensynthase und *Solanum habrochaites* Santalen- und Bergamotensynthase

Um das molekulare *Docking* von Reaktionsintermediaten untersuchen zu können, ist es notwendig, die konzeptionelle Beschreibung solcher Strukturen durch *Docking*-Programme zu betrachten und entsprechend anzupassen. Im Folgenden wird die Anpassung des *Docking*-Programms PLANTS mit Voruntersuchungen motiviert. Anschließend werden die umfangreichen durchgeführten *Docking*-Experimente analysiert und diskutiert.

4.3.1 Das Geranyl kation in verschiedenen Kraftfeldern

Die Modellierung des Geranyl kations mit verschiedenen Kraftfeldern zeigt, dass kein Kraftfeld außer TAFF das formal geladene Kohlenstoffatom des Geranyl kations als sp^2 -hybridisiert modelliert. Bis auf TAFF liefern alle Kraftfelder in MOE eine Fehlermeldung, dass nicht parametrisierte Atome vorliegen und somit keine Optimierung möglich ist. Mit dem TAFF-Kraftfeld wurde eine entsprechende Parametrisierung gefunden. Die korrespondierende Abbildung 52 zeigt, dass mit der Gasteiger-Methode zur Berechnung von Partialladungen (⁴¹⁹) aufgrund der Atomtypen und -konnektivität eine intramolekulare Polarisierung modelliert wird. Aufgrund dieser Polarisierung wird den direkt benachbarten Wasserstoffatomen des Kohlenstoffatoms, dem die positive Formalladung angewiesen wurde, eine hohe positive Ladung zugewiesen. Gleichzeitig bleiben alle Wasserstoffatome an tertiären Kohlenstoffatomen ohne Partialladung. Die Nachbaratome der Kohlenstoff-Kohlenstoff-Doppelbindungen werden positiv polarisiert. Das allylisch delokalisierte Kation ist allerdings stark asymmetrisch geladen, was der herkömmlichen Theorie zur Ladungsverteilung widerspricht. Das ist durch Anwendung einer einfachen Partialladungsberechnung im Kraftfeld nicht anders zu erwarten.

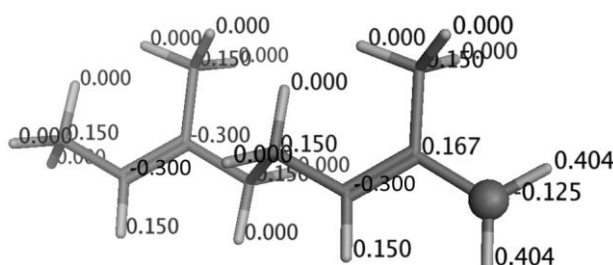


Abbildung 52: Geranyl kation mit Partialladungen nach Struktur-Optimierung mit TAFF-Kraftfeld in MOE. Die Position der positiven Formalladung ist durch eine Sphäre hervorgehoben; die angegebenen Partialladungen sind Gasteiger-Ladungen.

Weil Kraftfelder auf die Modellierung von Elektronen verzichten, kann das Konzept der Delokalisierung nur eingeschränkt modelliert werden. Allein für die in biochemischen Modellierungen häufig auftretenden Sauerstoffatome in Phosphat- und Carboxylat-Anionen wurde zum Beispiel in der TRIPOS-Atomtypen-Definition eine eigene Atomtypisierung eingeführt. Der TRIPOS-Atomtyp *C.cat* ist von den Entwicklern nur für die Anwendung in Guanidinium-Ionen gedacht. Solche Typisierungen erlauben eine spezielle Modellierung, können aber nicht die komplette Bandbreite physikalisch-chemischer Eigenschaften abbilden. Folglich leidet der Realismus von Kraftfeldmodellierungen bereits unter der Kategorisierung nach Atomtypen.

4.3.2 Quantenchemische Berechnungen zur Wechselwirkung zwischen Karbokationen und Aromaten bzw. Sulfiden

Es wurden flexible Koordinaten-Scans zur Stabilisierung des Terpinylkations durchgeführt um zu zeigen, dass die betrachteten Wechselwirkungen eine Stärke haben, die potenziell großen Einfluss auf enzymatisch katalysierte Reaktionsverläufe haben kann. Die Ergebnisse der Optimierungen mit der Hartree-Fock-Methode (Basissatz 6-31g) sind in Abbildung 53 veranschaulicht.

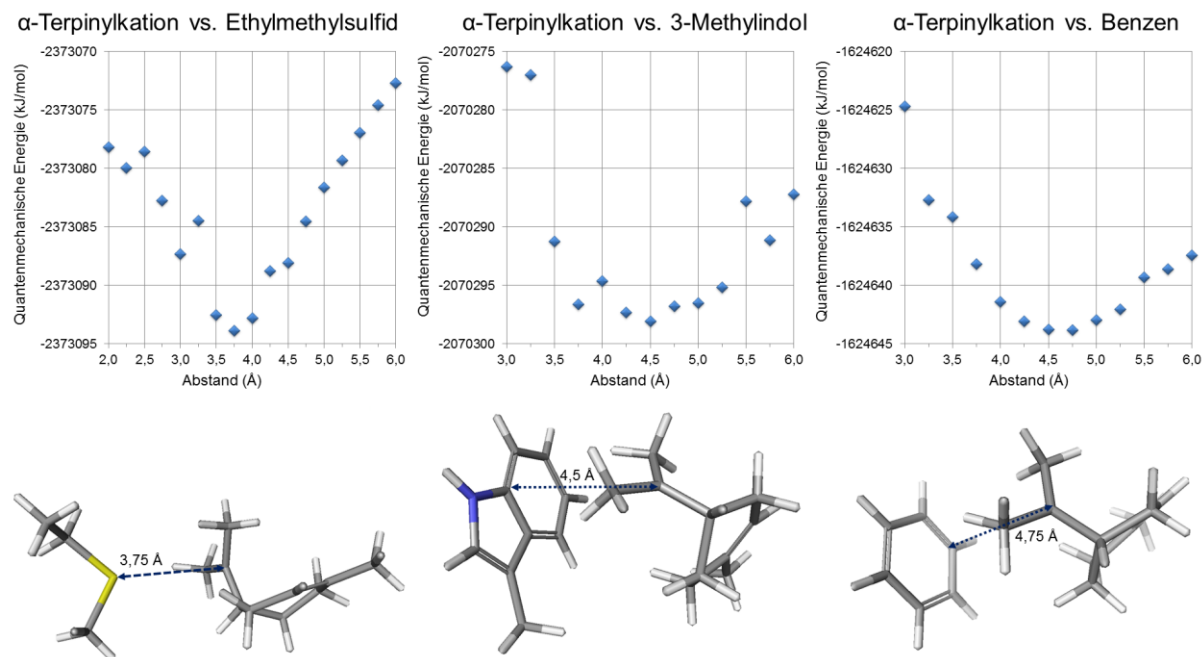


Abbildung 53: Ergebnisse flexibler Koordinaten-Scans – oben: Quantenmechanische Energien bei der Annäherung des α -Terpinylkations an verschiedene Modellmoleküle entsprechend Abbildung 22, Seite 55; unten: Wechselwirkungsgeometrien mit stärkster Wechselwirkung (Abbildung erstellt unter Benutzung von PyMOL)

Die Wechselwirkungsgeometrien zeigen unterschiedliche Interaktionsarten. Im Falle des Ethylmethylsulfids wechselwirken die freien Elektronenpaare des Schwefelatoms mit dem unbesetzten nichtbindenden π -Orbital des allylischen Kations. Sie interagieren gleichzeitig mit den positivierten Wasserstoffatomen der Methylgruppen am Allylkation. Die Polarisation dieser Methylgruppenwasserstoffatome beruht auf Hyperkonjugation: Die C-H-Bindungen dieser Methylgruppen geben Elektronendichte an das unbesetzte nichtbindende π -Orbital des Allylsystems ab. Dies führt einerseits zur Stabilisierung des allylischen Systems und andererseits zur erwähnten Positivierung der Wasserstoffe der benachbarten Methylgruppen. Deren Wechselwirkung mit den freien Schwefelelektronen stabilisiert die dargestellte Molekülgeometrie zusätzlich (Abbildung 53). Die Attraktion zwischen Ethylmethylsulfid und dem α -Terpinylkation beträgt mit der verwendeten Hartree-Fock-Methode mit Basissatz 6-31g in der dargestellten Geometrie über 20 kJ/mol. Die Berechnung eines genaueren Wertes wurde hier weder durch leistungsfähigere Berechnungsmethoden, beispielsweise unter Einbezug der Elektronenkorrelation, noch durch eine Erweiterung des betrachteten *Scan*-Abstandes weiter verfolgt, weil das Kerninteresse auf den grundsätzlichen qualitativen Gegebenheiten der untersuchten Wechselwirkungen liegt.

Für die beiden Fälle der aromatischen Wechselwirkungen des α -Terpinylkations sind die Wechselwirkungsenergien niedriger. In der dargestellten Wechselwirkungsgeometrie mit 3-Methylindol ist eine C-H-Bindung einer Methylgruppe des tertiären Karbokations direkt auf das π -System des Indol ausgerichtet. Auch hier ist von hyperkonjugativer Elektronendichteververschiebung in das nichtbindende π -Orbital des allylischen Kations auszugehen. Die positivierten Wasserstoffatome interagieren daher analog mit dem Indol- π -System. Eine entsprechende Beschreibung trifft auch auf die dargestellte Wechselwirkungsgeometrie des α -Terpinylkations mit Benzen zu, auch wenn die Wechselwirkungsenergie in diesem Fall geringer ausfällt. Dies kann sowohl auf die geringere Größe des aromatischen π -Systems als auch die geringere Elektronendichte im Vergleich zum stickstoffhaltigen Indolsystem zurückzuführen sein.

Die Abstandsoptima für die betrachteten Wechselwirkungen liegen zwischen 3,5 Å und 5,0 Å. Diese Erkenntnis ist wesentlich für die nachfolgenden Modifikationen an *Scoring*-Funktionen des PLANTS-Programms (Abschnitt 4.3.4, Seite 97 ff.).

Bräuer und Kollegen berichten für die Wechselwirkung zwischen einem Allylkation und Indol eine attraktive Energie von 30,6 kJ/mol.⁶⁴ Brandt *et al.* berechneten ferner den Wert -85 kJ/mol als Wechselwirkungsenergie zwischen Tryptophan und einem Allylkation.³⁶⁷ Eine weitere Studie zu Kation- π -Wechselwirkungen stammt von Biot und Kollegen.³⁴⁸ Sie kalkulierten unter anderem paarweise Wechselwirkungsenergien von Adenin mit Lysin bzw. Arginin in verschiedenen Lösungsmitteln. Die Attraktion zwischen Adenin und Arginin ist dabei -28 kJ/mol und stärker und auch die Wechselwirkungen zwischen Adenin und Lysin sind mit mindestens -8 kJ/mol sehr attraktiv.

Ähnliche, wenn auch nichtionische, Wechselwirkungen zwischen Tryptophan- und Methioninseitenketten wurden als ein strukturbildendes Motiv in Proteinen identifiziert.⁴²⁰ Auch in AtKS und SIPHS sind in der Substratbindetasche diese Aminosäuren so benachbart, dass starke Wechselwirkungen zwischen ihnen auftreten können wie von Duan *et al.* charakterisiert.⁴²¹ Es besteht daher die Möglichkeit, dass ein kationischer Zwischenzustand der jeweils katalysierten Reaktion in solchen Terpensynthesen stabilisiert wird, wie es in Abbildung 54 dargestellt ist.

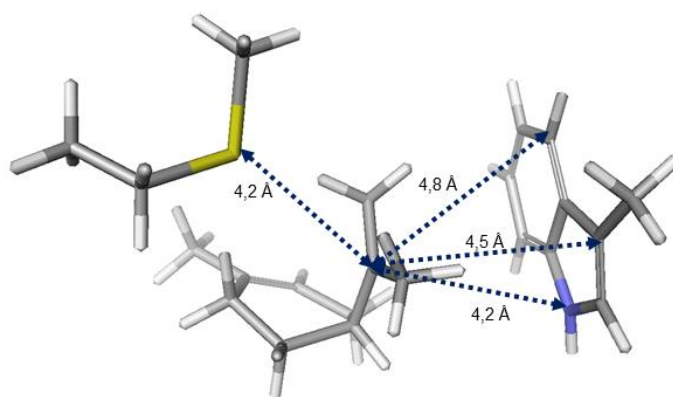


Abbildung 54: Quantenmechanisch optimierte Wechselwirkungsgeometrie des α -Terpinylkations zwischen 3-Methylindol und Ethylmethylsulfid (Abbildung erstellt unter Benutzung von PyMOL)

Die paarweisen Wechselwirkungsgeometrien ähneln den in Abbildung 53 (Seite 94) vorgestellten. Allerdings kommen in der in Abbildung 54 dargestellten Wechselwirkungsgeometrie zwei Methylgruppen des α -Terpinylkations dem Indol- π -System nahe, so dass hier

offenbar eine stärkere Attraktion vorliegt. Das könnte die Ursache dafür sein, dass andererseits die Wechselwirkung mit dem Sulfid schwächer ausfällt als bei der Geometrie in Abbildung 53. Dies spiegelt sich im bezeichneten Abstand wider. Der grundlegende Effekt, dass durch eine solche Sandwichstruktur die paarweisen Wechselwirkungen in Gasphase geringer sind als bei Betrachtung nur einer Wechselwirkung, wurde eingehend unter Verwendung theoretischer Methoden untersucht.^{422–424}

4.3.3 Versuch zur Abhängigkeit bei *Cluster-Docking*

Das Terpinylkation wurde mit PLANTS in die Bindetasche der Röntgenkristallstruktur der Limonensynthase aus *M. spicata* gedockt. Der Vergleich von 30 unabhängig voneinander erzeugten *Docking*-Posen mit 30 *Docking*-Posen, die mit aktivierter Option `cluster_structures = 30` erzeugt wurden, zeigt deutliche Unterschiede. Bei unabhängigen Durchläufen werden quasi identische Posen gefunden, aber diversitätsorientiertes *Docking* liefert wie angefordert unterschiedliche *Docking*-Posen (siehe Abbildung 55). Es gilt jedoch zu beachten, dass die Bindetasche starr gehalten wurde und das Terpinylkation in der verwendeten Definition nur über einen drehbaren Torsionswinkel verfügt. Es ist daher davon auszugehen, dass diversitätsorientiertes *Docking* für größere und flexiblere Liganden bei der Einstellung `cluster_rmsd = 2` relevante Unterschiede in den Ergebnisgeometrien liefert.

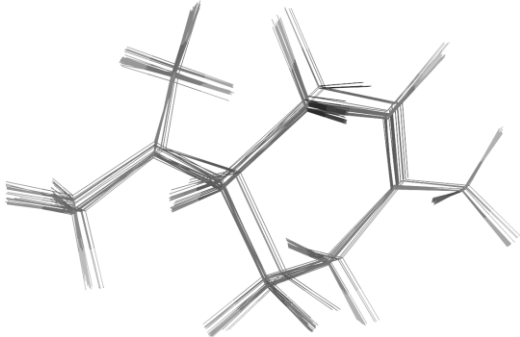

Unabhängiges <i>Docking</i>	versus	Diversitätsorientiertes <i>Docking</i>
<ul style="list-style-type: none"> • <code>cluster_structures = 1</code> • <code>cluster_rmsd = 2.0</code> (hier nicht maßgeblich) • 30-facher Start von PLANTS 		<ul style="list-style-type: none"> • <code>cluster_structures = 30</code> • <code>cluster_rmsd = 2.0</code> • einfacher Start von PLANTS
<u>Veranschaulichung der Ergebnisse</u>		
		
Die jeweils bestbewerteten Posen stimmen im Bindemodus überein und unterscheiden sich kaum voneinander.		Die resultierenden Posen streuen stark in der räumlichen Ausrichtung.

Abbildung 55: Einfluss der *Cluster*-Option auf die Variabilität von Ergebnissen am Beispiel des PLANTS-*Docking* des Terpinylkations in die starre Bindetasche der Limonensynthase aus *Mentha spicata* (PDB: 2ONH) mit der CHEMPLP-*Scoring*-Funktion; für die kompletten Konfigurationsdateien und Ligandendefinition siehe A 9, Seite 155

4.3.4 Modifikation der *Scoring*-Funktion in PLANTS

Die unter 3.3.3 (Seite 43 ff.) beschriebene Modifikation der CHEMPLP-*Scoring*-Funktion erweitert die Menge der explizit modellierten Wechselwirkungen. Hinzu kommen die Fälle der Wechselwirkungen zwischen Carbokationen und Schwefelatomen in Methioninseitenketten einerseits bzw. aromatischen Atomen von Aminosäureseitenketten andererseits. Die verwendete Modellierung variiert das PLP-förmige, rein abstandsabhängige Potenzial bezüglich des Parameters E für intermolekulare Atompaarinteraktionen (siehe auch Abbildung 18, Seite 41).

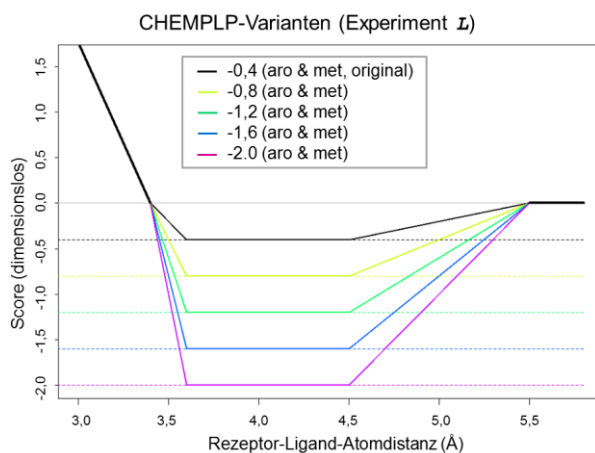


Abbildung 56: verwendete CHEMPLP-Varianten (im Experiment 1) für die beiden Wechselwirkungsarten zwischen Carbokationen und aromatischen Aminosäureseitenkettenatomen („aro“) bzw. Schwefelatomen von Methioninseitenketten („met“)

4.3.5 Auswertung der *Docking*-Experimente

Die Unterschiede zwischen den *Docking*-Ergebnissen aus den resultierenden verschiedenen CHEMPLP-Funktionen werden im Folgenden analysiert. Dabei wird nur ein Bruchteil der erhobenen Daten präsentiert. Es werden charakteristische Beispiele gegeben, anhand derer konkrete Verhältnisse erklärt werden.

Verschiedene geometrische Größen und Scores der umfangreichen *Docking*-Resultate werden im Folgenden teilweise statistisch analysiert. Die mittels des Kolmogorov-Smirnov-Tests errechneten p-Werte sollen beim Vergleich ermittelter Größen als grober Anhaltspunkt für die Ähnlichkeit zweier empirischer Verteilungen gelten. Sie geben an, wie wahrscheinlich es ist, genau die jeweiligen zwei Datenverteilungen zu finden, unter der Hypothese, dass sie demselben stochastischen Prozess entstammen. Es wurde kein Signifikanzniveau festgelegt, um diese Nullhypothese zu akzeptieren oder zurückzuweisen. Die Darstellungen beinhalten vielmehr die konkreten p-Werte, um einen Einblick in das Spektrum der Vergleichsergebnisse abzubilden. Der Kolmogorov-Smirnov-Test ist parameterfrei und bietet ferner den Vorteil, dass keine Annahme einer bestimmten Verteilung an die verglichenen Daten gestellt werden muss.

Die abgebildeten Dichteverteilungen der jeweils aufgetragenen Größen werden in der verwendeten Methode als Linearkombination von Gaußkernen berechnet. Sie ergänzen die Informationen der *Boxplot*-Darstellungen und erlauben eine detailliertere vergleichende Betrachtung von Verteilungen von Datenpunkten als die *Boxplots* allein. Die kontinuierliche Auftragung der diskreten Daten bietet eine übersichtliche Darstellungsform für mehrere Verteilungen in einer Grafik.

4.3.5.1 Bezeichnungen und Terminologie

Die neu hinzugefügten Potenziale werden abgekürzt bezeichnet: $E_{aro}-E_{met}$ steht für das modifizierte PLANTS-CHEMPLP, in dem die Potenzieltiefe (entspricht dem jeweiligen Minimum E in Abbildung 56) für Wechselwirkungen zwischen kationischen Kohlenstoffatomen und Aromaten E_{aro} und für Wechselwirkungen zwischen kationischen Kohlenstoffatomen und Schwefelatomen von Methioninen E_{met} beträgt. Ein Potenzial mit der Bezeichnung „-2_-0.4“ entspricht demnach einer CHEMPLP-Modifikation, in der karbo-kationisch-aromatische Interaktionen mit dem magentafarben dargestellten Potenzial in Abbildung 56 und Karbokation-Methioninschwefel-Interaktionen mit dem in Abbildung 56 schwarz dargestellten Potenzial modelliert werden. Das Potenzial „-0.4_-0.4“ kommt der Originalversion gleich und ist in den meisten *Docking*-Experimenten als Kontrollinstanz enthalten. Man beachte, dass in diesen Potenzialbezeichnungen und den folgenden Abbildungen das Dezimal-Komma durch einen Punkt ersetzt ist.

Eine *Docking*-Pose wird als besser oder im Rang höher (hochrangiger) als eine andere Pose bezeichnet, wenn ihr *Score* niedriger ist, da bei PLANTS negative *Scores* favorisierte Wechselwirkungen zum Ausdruck bringen. Bezüglich der Anzahl der betrachteten Posen bezeichnet „Top n Posen“ die n per *Score* bestbewerteten erzeugten *Docking*-Posen. Angegebene *Scores* aus PLANTS-*Dockings* sind dimensionslos.

Der Begriff der *Score*-Relevanz soll anzeigen, ob zu einer im Voraus über Atomnummern definierten potenziellen Wechselwirkung gemessene Distanzen in dem Intervall liegen, in dem sie durch die *Scoring*-Funktion als favorisiert bewertet werden. Dieser Distanzbereich (vgl. Abbildung 56, Seite 97) umfasst das Intervall [3,4; 5,5] Å.

Für die Parameterwahl zu den jeweils mit Identifikator $a...s$ bezeichneten *Docking*-Experimenten siehe Tabelle 10, Seite 51.

4.3.5.2 *Docking* des α -Terpinylkations in das Limonensynthesemodell

Unter Verwendung verschiedener Parameter für die beiden neu modellierten Wechselwirkungen wurden zuerst die Auswirkungen auf die Gesamt-*Scores* der resultierenden Posen geprüft.

Abbildung 57 veranschaulicht wesentliche Charakteristika der *Score*-Verteilungen zu diversitätsorientiertem *Docking* des α -Terpinylkations in das Limonensynthesemodell mit sechs Potenzialvarianten unterschiedlichen E_{aro} -Wertes. Es ist kein Trend erkennbar, dass die resultierenden *Scores* mit sinkendem Potenzialwert E_{aro} insgesamt sinken, obwohl in der Bindetasche der Limonensynthese sowohl Methionin als auch aromatische Aminosäuren vorkommen. Dieser Effekt ist auch nicht durch mangelndes *Sampling* der entsprechenden Wechselwirkungen wegen der Diversitätsorientierung zu erklären, denn er tritt unter den Top 10 Posen ebenfalls auf (Daten nicht gezeigt). Die p-Wert-Matrix deutet darauf hin, dass es zwar Unterschiede zwischen den *Score*-Verteilungen gibt, diese aber keiner Monotonie entsprechend des Potenzialwert E_{aro} folgen. Es ist anzumerken, dass die p-Werte für Potenzialpaare in anderen Potenzialreihen auch die Extremwerte null und eins umfassen. Eine monotone Regelmäßigkeit in Abhängigkeit des Potenzialwertes E_{aro} bzw. E_{met} ist jedoch nicht nachzuweisen.

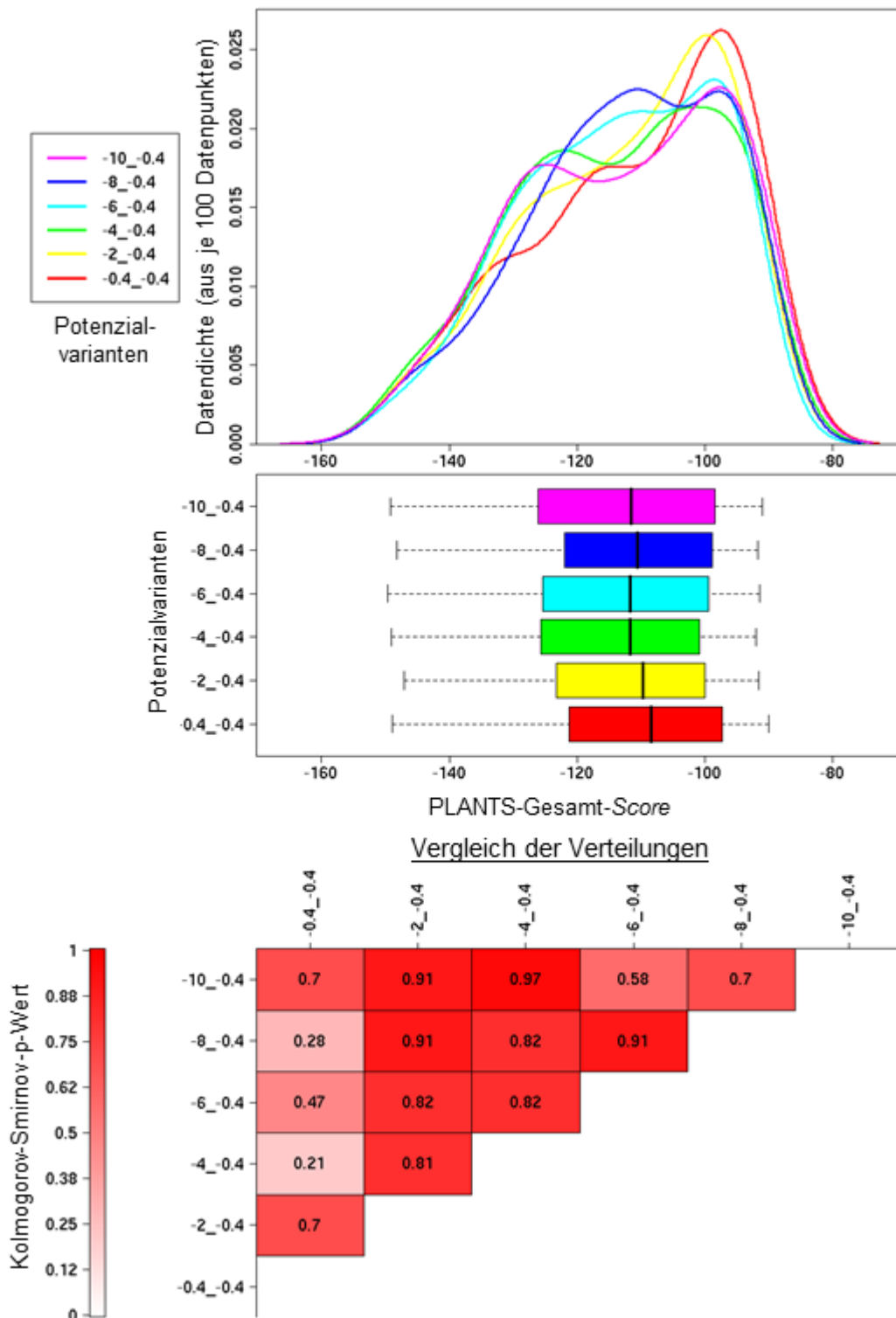


Abbildung 57: Dichte-Diagramm und *Boxplot* sowie p-Wert-Matrix der Verteilungen der Gesamt-Scores in ausgewählten (aro-) Potenzialvarianten zum Experiment a

Da die CHEMPLP-Scoring-Funktion eine Linearkombination verschiedener Beiträge ist und die Veränderung der Parameter sich lediglich in einem dieser Teile wiederfindet, ist das Verhalten dieses Score-Anteils (PLPpartsteric) bei Potenzialvariation besonders interessant. Die generelle Tendenz ist jedoch ähnlich der Verhältnisse bei den Gesamt-Scores, wie Abbildung 58 verdeutlicht. Im Allgemeinen lassen die erzeugten Darstellungen – wie hier am Beispiel deutlich wird – keinen monotonen Zusammenhang zwischen Potenzialwert und Gesamt-Score bzw. PLPpartsteric-Score-Anteil erkennen.

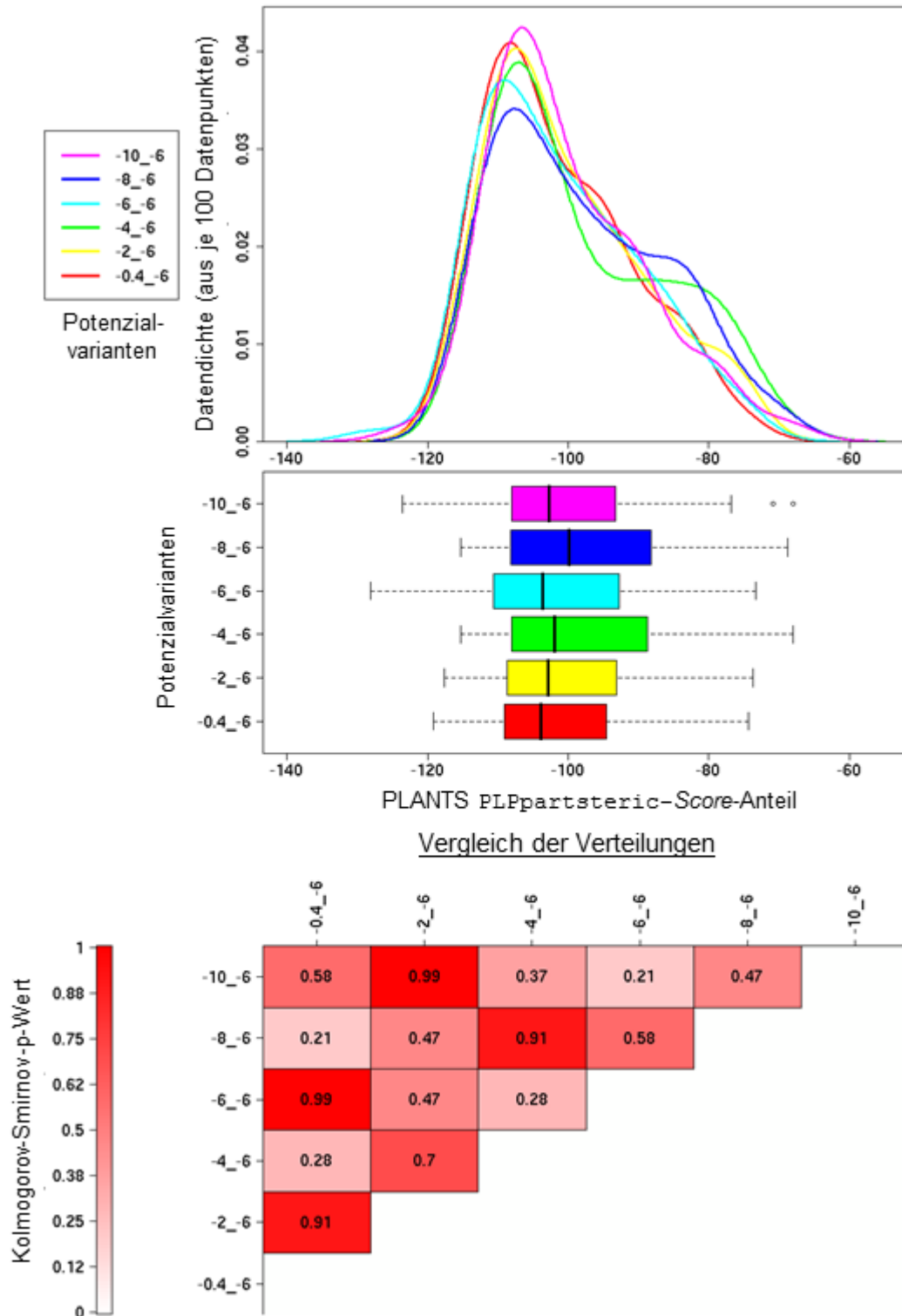


Abbildung 58: Dichte-Diagramm und *Boxplot* sowie p-Wert-Matrix der Verteilungen der PLPpartsteric-Score-Anteile in ausgewählten (aro-) Potenzialvarianten zum Experiment a

Um statt Unterschieden im *Scoring* differenzierbare Geometrien in den Wechselwirkungsposen identifizieren zu können, müssen geometrische Faktoren analysiert werden. Hierzu wurden manuell Aminosäuren in der Substratbindetasche ausgewählt, die potenziell direkt mit Liganden wechselwirken können. Für die ausgewählten Aminosäuren wurden Repräsentanzpunkte definiert, wie in Abbildung 59 angeführt. Diese entsprechen bei aromatischen Aminosäuren dem geometrischen Mittelpunkt des aromatischen Systems. Für nichtaromatische Aminosäure wurden die Repräsentanzpunkte auf charakteristische Atome

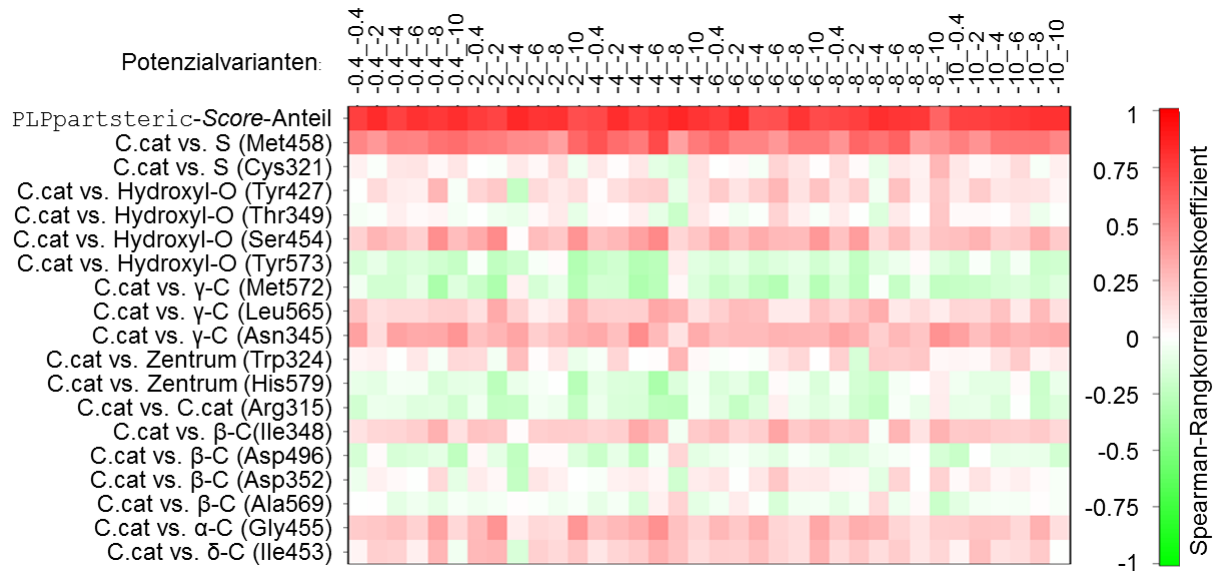


Abbildung 60: Rangkorrelation mit PLPpartstERIC und gemessenen Distanzen ausgewählter Wechselwirkungen im Experiment a (Bezeichnungen siehe Abbildung 59)

Auch ist die Rangkorrelation bezüglich des Abstandes des kationischen Atoms zu Trp324 wesentlich geringer, als die Score-Relevanz vermuten lässt. Eine mögliche Ursache hierfür ist, dass diese Wechselwirkung nahezu immer gefunden wird, was durch die relative Größe der Indolgruppe deutlich wahrscheinlicher ist als bei kleineren Aminosäureseitenketten, die weniger leicht für Wechselwirkungen zugänglich sind.

Als empirischer Beweis dafür, dass es keinen monotonen Zusammenhang zwischen gemessenen Distanzen bzw. dem Score-Anteil PLPpartstERIC und der Variation der Parameter E_{aro} und E_{met} gibt, können die in Abbildung 61 dargestellten Daten gelten. Auch wenn einige Korrelationskoeffizienten betragsmäßig größer als 0,8 sind, gibt es innerhalb der Matrix keinerlei offenbare Monotonie, wie sie zu erwarten wäre, wenn die Variation der Parameter sich stark in den gemessenen Größen niederschläge.

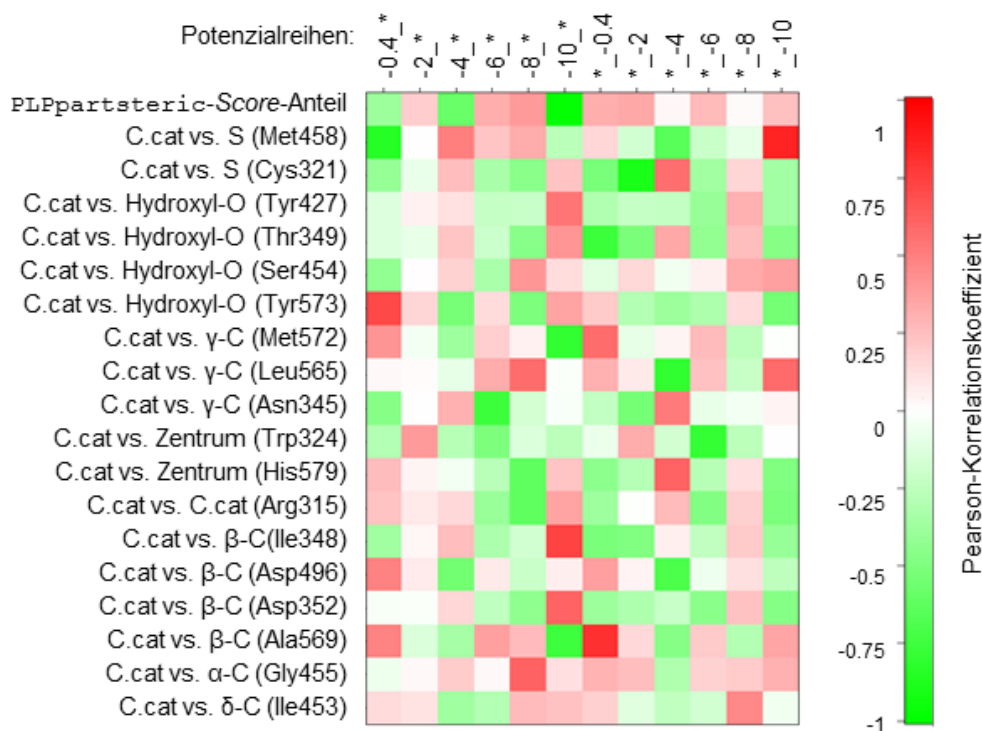


Abbildung 61: Korrelationen der Parametervariationen im Experiment **A** (hier die Folge (-0,4; -2; -4; -6; -8; -10)) mit Rangkorrelationskoeffizienten von PLPpartstERIC und gemessenen Distanzen ausgewählter Wechselwirkungen aus den entsprechenden Potenzialvarianten. Für die Bezeichnungslegende siehe Abbildung 59, Seite 101. Das *-Symbol steht in der Bezeichnung der Potenzialreihen für die Reihe der Parameter des jeweils anderen Parameterwertes, wie sie im Experiment verwendet wurden.

Ähnliche Verhältnisse herrschen bei diesen Korrelationen in den beiden anderen Experimenten mit *Docking* in die Limonensynthesestruktur, **B** und **E**. Daher werden diese Ergebnisse hier nicht präsentiert. Es ist jedoch anzumerken, dass aufgrund des diversitätsorientierten *Docking* in diesen drei Experimenten statistische Abhängigkeiten in der Generierung der *Docking*-Posen bestehen. Insbesondere führt jede akzeptierte *Docking*-Pose dazu, dass in Folgeschritten neue und andere Posen erzeugt werden, welche dann zumindest teilweise auf favorisierte Eigenschaften der früheren Pose verzichten müssen, um dem RMSD-Kriterium der Diversitätseinstellung zu genügen.

Das α -Terpinylkation besitzt nur eine frei drehbare Bindung. Mit lediglich zwei als flexibel eingestellten Aminosäuren in der Bindetaschenstruktur der Limonensynthese ist der Posenraum relativ klein im Vergleich zu vielen anderen praktischen *Docking*-Fragestellungen. Das erschöpfende *Docking* in Experiment **E** lieferte mindestens 1412 *Docking*-Posen pro Potenzialvariante bei einem Diversitäts-Parameter von $d_{clust} = 1 \text{ \AA}$. Durch die Diversitätsforderung verliert die Score-Relevanz wichtiger Aminosäuren in der Bindetasche relativ an Gewicht, wie Abbildung 62 verdeutlicht: Die beiden Experimente **A** und **B** wurden mit $d_{clust} = 2 \text{ \AA}$ gestartet, forderten aber pro Potenzialvariante nur 100 Posen. Im Fall des erschöpfenden *Docking* **E** liegen für keine der betrachteten Wechselwirkungen mehr als 50% der Posen im Score-relevanten Distanzbereich.

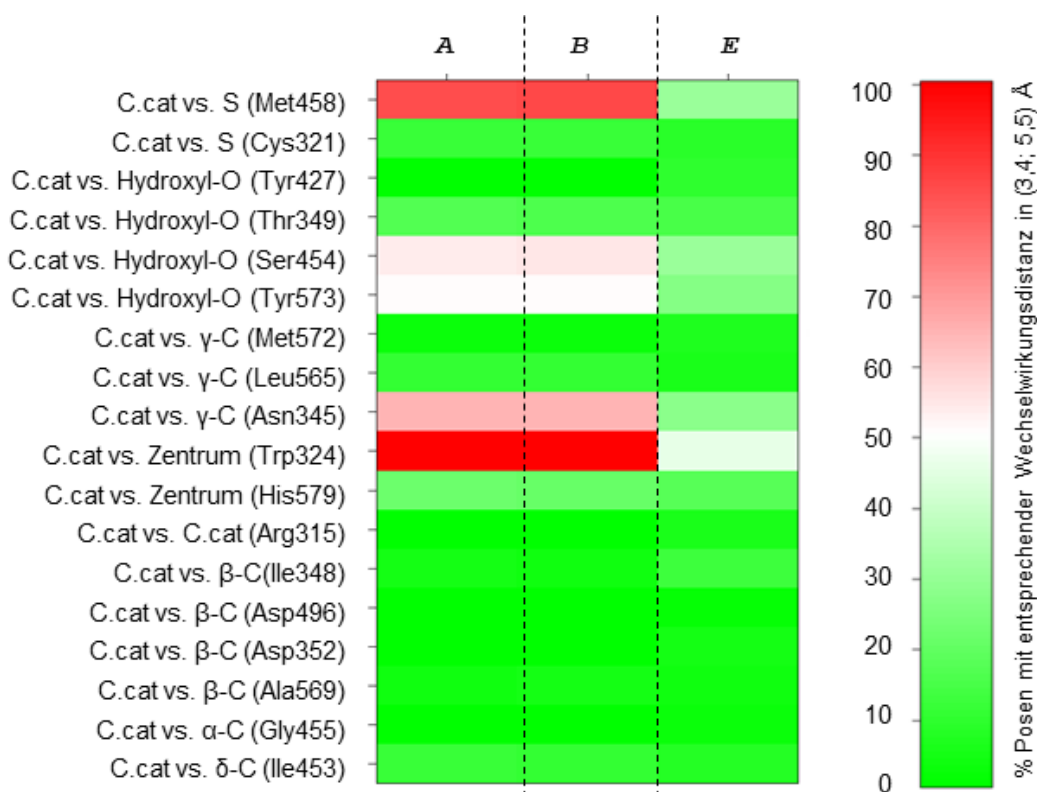


Abbildung 62: Farbliche Veranschaulichung des jeweiligen Posenanteils in den Experimenten *A*, *B* und *E*, bei dem die betrachteten Wechselwirkungen innerhalb des *Score*-relevanten Distanzbereichs liegen. Das *Docking* erfolgte jeweils in das Limonensynthesemodell und jeweils über alle Potenzialvariationen hinweg.

4.3.5.3 *Docking* des (Z,Z)-Farnesylkations in das ShSBS-Modell

Mit der deutlich größeren Anzahl an Freiheitsgraden beim *Docking* von (Z,Z)-FPP in das erstellte ShSBS-Modell sind komplexere Verhältnisse und bessere Sichtbarkeit der Einflüsse der variierten Parameter zu erwarten. Das erste *Docking*-Experiment (*c*) in das ShSBS-Modell war diversitätsorientiert mit $d_{clust} = 2 \text{ \AA}$ und forderte 1.000 Posen pro Potenzialvariante. Wahrscheinlich ist es der großen Posenanzahl geschuldet, dass die *Score*-Verteilungen für dieses Experiment sich statistisch wesentlich weniger ähneln als in den kleineren Experimenten zur Limonensynthese. Abbildung 63 zeigt beispielhaft, dass die p-Werte des Kolmogorov-Smirnov-Tests im Allgemeinen sehr klein sind. Interessant sind die hohen p-Werte für die *Score*-Verteilungspaare aus den Potenzialpaaren 0_-0.8 und 0_-0.4 sowie 0_-5 und 0_-0.6.

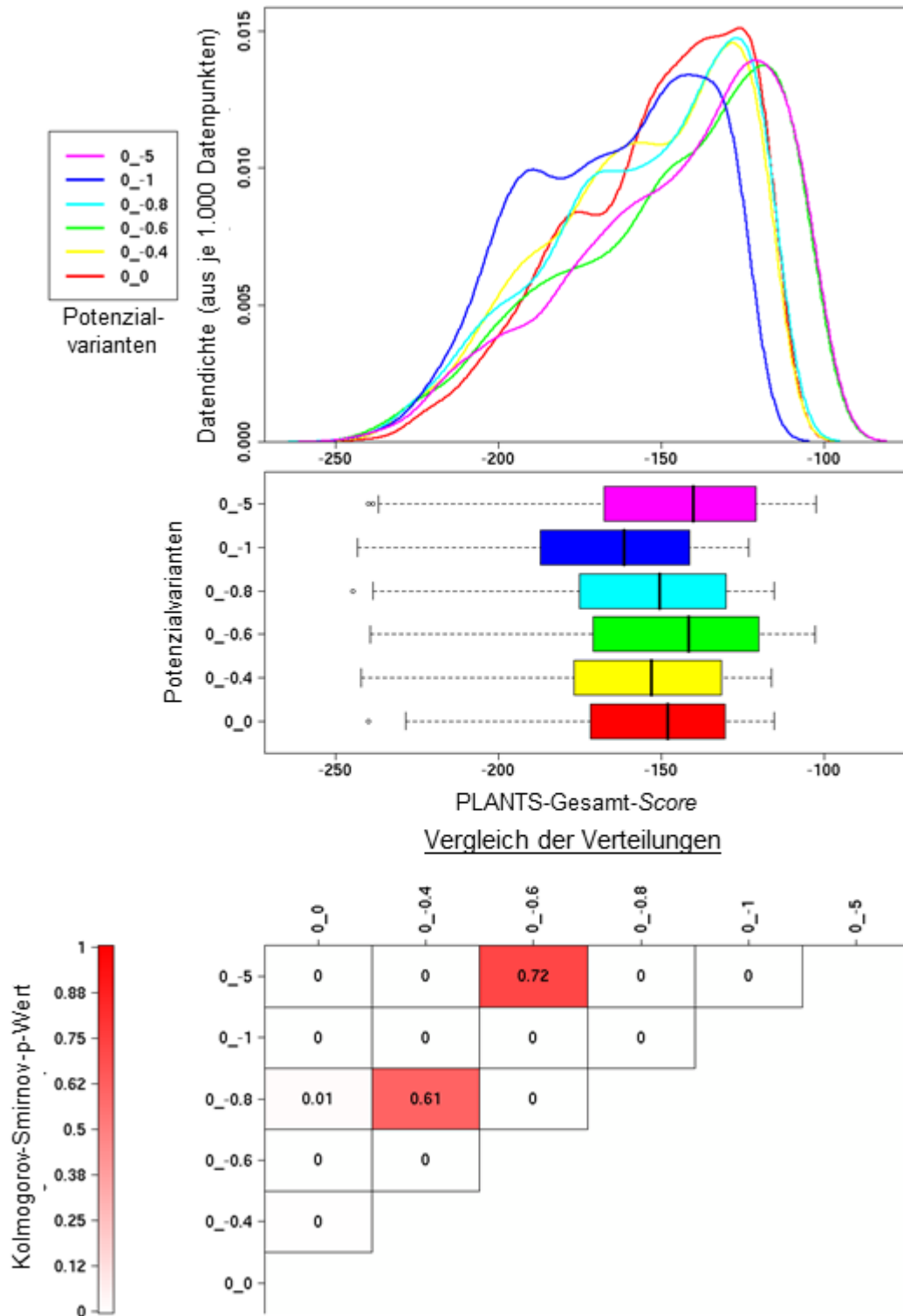


Abbildung 63: Dichte-Diagramm und *Boxplot* sowie p-Wert-Matrix der Verteilungen der Gesamt-Scores in ausgewählten (met-) Potenzialvarianten zum Experiment *c*

Ein Vergleich mit der entsprechenden Darstellung für den *PLPpartstERIC-Score*-Anteil (Abbildung 64) zeigt, dass hier nur im Fall 0_-5 versus 0_-0.6 ein hoher p-Wert auftritt. Es existiert demnach eine differenzierte Abhängigkeit der *Score*-Verteilungs-Ähnlichkeit bei verschiedenen Potenzialvarianten zwischen Gesamt-Score und *PLPpartstERIC*-Anteil.

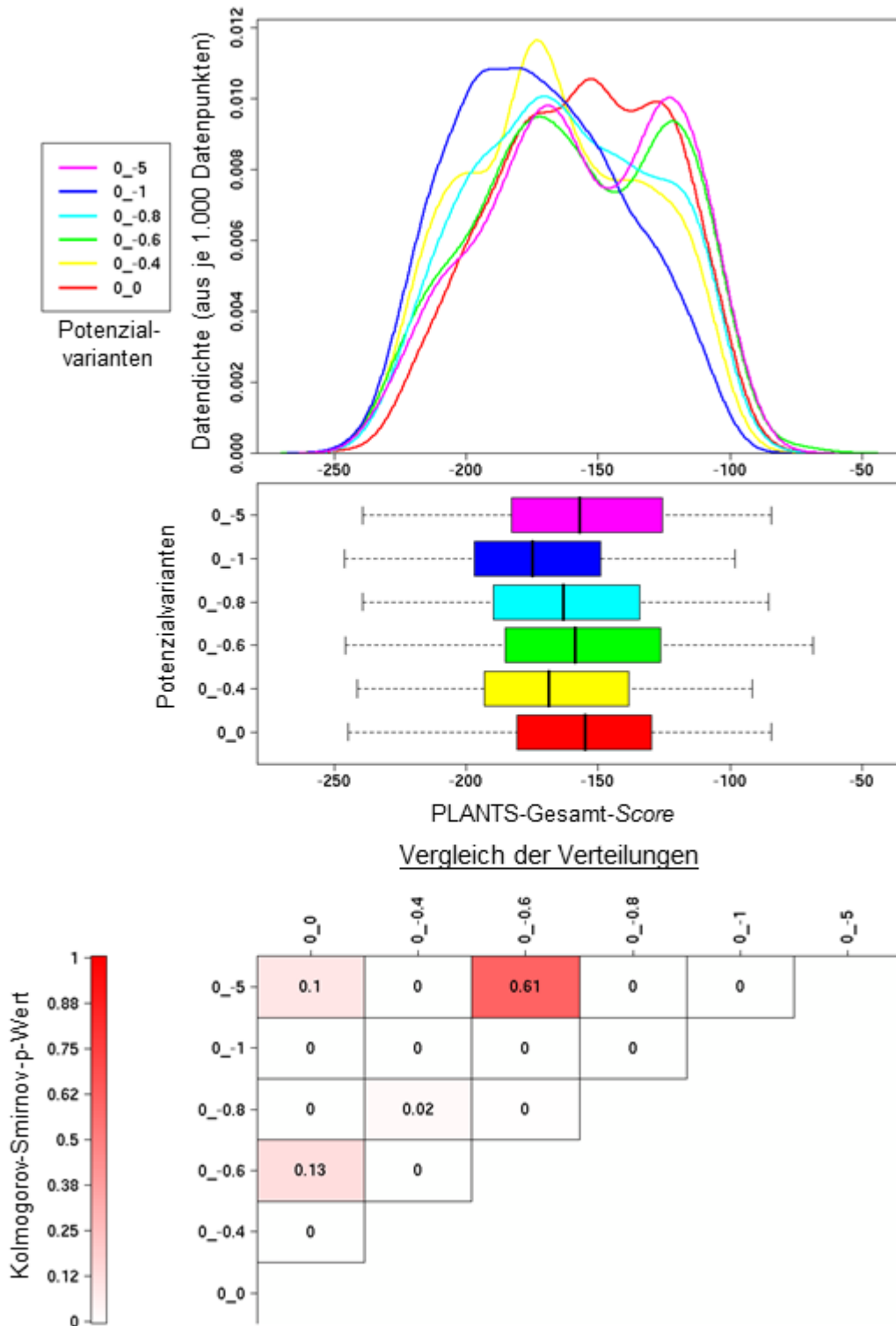


Abbildung 64: Dichte-Diagramm und *Boxplot* sowie p-Wert-Matrix der Verteilungen der PLPpartsteric-Score-Anteile in ausgewählten (met-) Potenzialvarianten zum Experiment *c* analog Abbildung 63

Eine Betrachtung der *Score*-Zusammensetzung bei 0_-0.6 bietet einen interessanten Einblick in die absoluten Werte von *Score*-Anteilen und deren Korrelationen zum Gesamt-Score (Abbildung 65). Die *Score*-Anteile PLPpartsteric und CHEMPLP_PLP_PROT haben mit Abstand die größten Absolutwerte. Der CHEMPLP_PLP_PROT-Term modelliert hierbei Interaktionen zwischen Rezeptoratomen. Die große Variabilität der Werte kommt unter

anderem durch die flexible Betrachtung von vier Aminosäuren in der ShSBS-Bindetasche zu Stande.

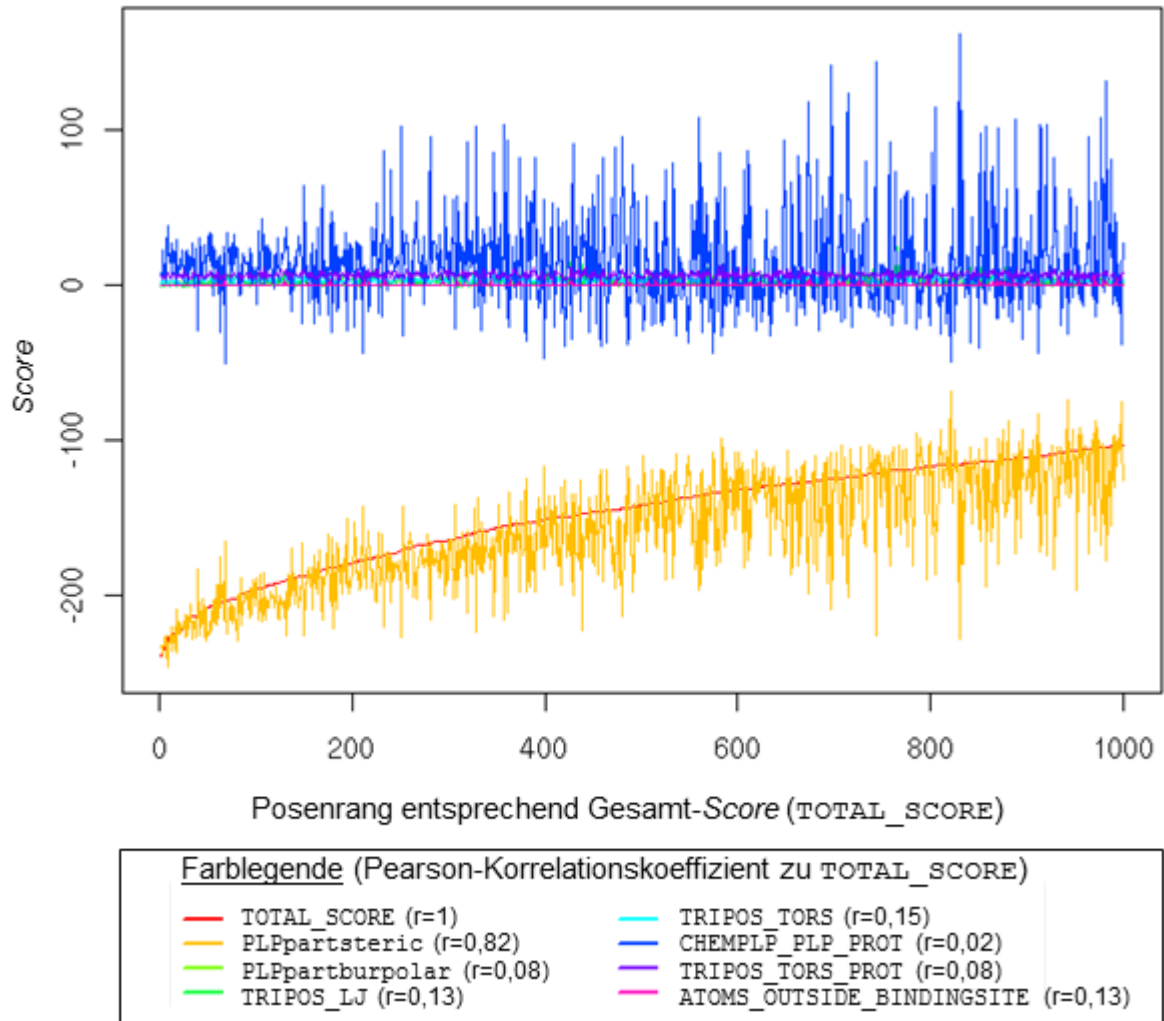


Abbildung 65: Auftragung der *Score*-Zusammensetzung für Posen, die mit dem Potenzial 0_-0.6 erzeugt wurden, und Pearson-Korrelationskoeffizienten zwischen *Score*-Anteilen und dem Gesamt-Score (Experiment c, zur besseren Verdeutlichung der Variabilität als kontinuierliche Auftragung). Bedeutung der Termbezeichnungen: TRIPOS_LJ – TRIPOS-Lennard-Jones-Potenzial, CHEMPLP_PLP_PROT – stückweise lineares Potenzial für intramolekulare Wechselwirkungen im Protein, TRIPOS_TORS_PROT – Torsionswinkelpotenzial für Aminosäureseitenketten; alle anderen siehe Gleichung 5, Seite 42.

Abbildung 65 und analoge Auftragungen für andere Potenzialvarianten, die hier nicht gezeigt werden, können als Hinweis darauf gelten, dass eine Veränderung im PLPpartsteric-Anteil, wie sie hier durchgeführt wird, Einfluss auf das *Ranking* von Posen hat. Die Korrelation zwischen dem Gesamt-Score und dem PLPpartsteric-Anteil liegt in allen untersuchten Potenzialvarianten zwischen 0,71 und 0,84 (unter Betrachtung aller 1.000 Posen der jeweiligen Potenzialvariante, Experiment c).

Eine Untersuchung der *Score*-Zusammensetzung für die Top 10 und Top 100 Posen zeigt allerdings, dass im Bereich dieser hochrangigen Posen zwischen Gesamt-Score und den einzelnen *Score*-Anteilen wesentlich andere Korrelationen vorliegen (ohne Abbildung): Das *Ranking* ist für die hochrangigen Posen wesentlich stärker durch unterschiedliche Einflüsse

der einzelnen *Score*-Anteile geprägt als in der großen Zahl der diversitätsorientiert erzeugten Posen – bis hin zu negativen Korrelationskoeffizienten für die Korrelation zwischen Gesamt-*Score* und *PLPpartsteric*-Anteil. Es liegt somit ein Abhängigkeitseffekt vor, in dem durch die Diversitätsforderung Posen zu ungünstigen Wechselwirkungen gezwungen werden.

Die Nummerierung der Aminosäuren des erstellten ShSBS-Modells hat sich im Laufe der Vorbereitungen als *Docking*-Rezeptor bei der Zwischenspeicherung in unterschiedlichen Dateiformaten verändert. Um in diesem Kapitel die Aminosäurepositionen mit den Positionen in Abschnitt 4.2 (Seite 71 ff.) zu identifizieren, muss von den hier angegebenen Positionen die Zahl 33 subtrahiert werden.

Die Darstellung der *Score*-Relevanz ausgewählter Wechselwirkungen in ShSBS hebt einige Aminosäuren in ihrer Bedeutung hervor, was aus Abbildung 66 deutlich wird.

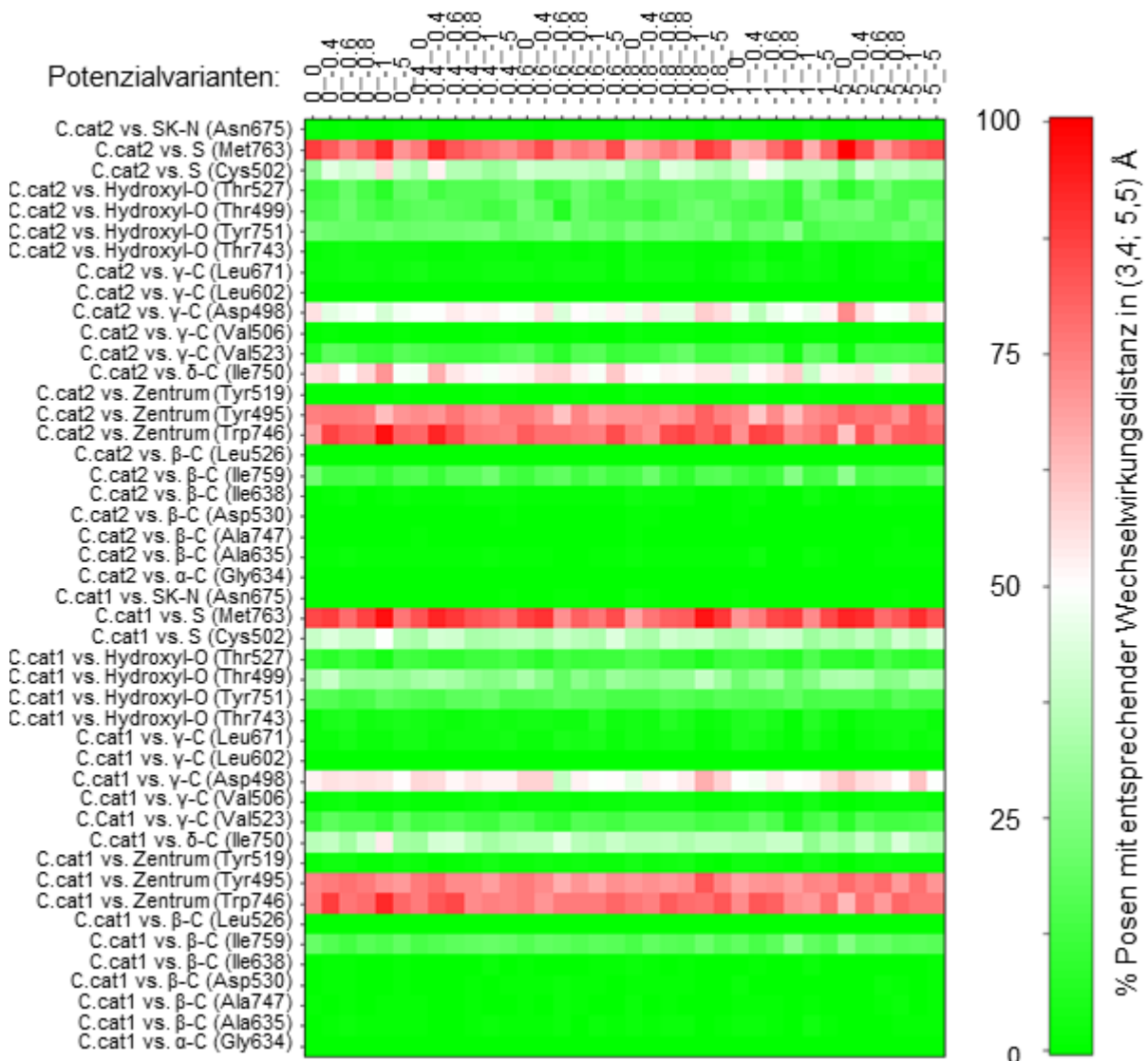


Abbildung 66: Farbliche Veranschaulichung des jeweiligen Posenanteils im Experiment *c*, bei dem die betrachteten Wechselwirkungen innerhalb des *Score*-relevanten Distanzbereichs liegen. (Abkürzungen: Siehe Abbildung 59, Seite 101, SK-N – Seitenketten-Stickstoffatom, C.cat1/2 – kationische Kohlenstoffatome des (Z,Z)-FPP, siehe Abbildung 19 B, Seite 52)

Die Aminosäuren Met763, Tyr495 und Trp746 interagieren in der Mehrheit aller *Docking*-Posen sowohl mit dem C.cat1- als auch dem C.cat2-Atom. Relativ häufig haben auch

Cys502, Asp498 und Ile750 Wechselwirkungen mit den beiden kationischen Kohlenstoffatomen des Liganden. Insbesondere ist für Ile750 zu erkennen, dass die Interaktion häufiger mit dem terminalen C.cat1-Atom als mit dem C.cat2-Atom zustande kommt. Umgekehrt verhält es sich für Thr499, welches insgesamt zwar selten, aber doch deutlich öfter mit dem C.cat2-Atom als mit dem C.cat1-Atom wechselwirkt. Weitere Wechselwirkungen treten untergeordnet auf. Insgesamt lassen sich keine auffälligen Regelmäßigkeiten über die Reihen der verwendeten Potenzialvarianten erkennen.

Auch die in Abbildung 67 dargestellten Rangkorrelationen der gemessenen Distanzen und des PLPpartsteric-Score-Anteils zum Gesamt-Score bieten ein differenziertes Bild.

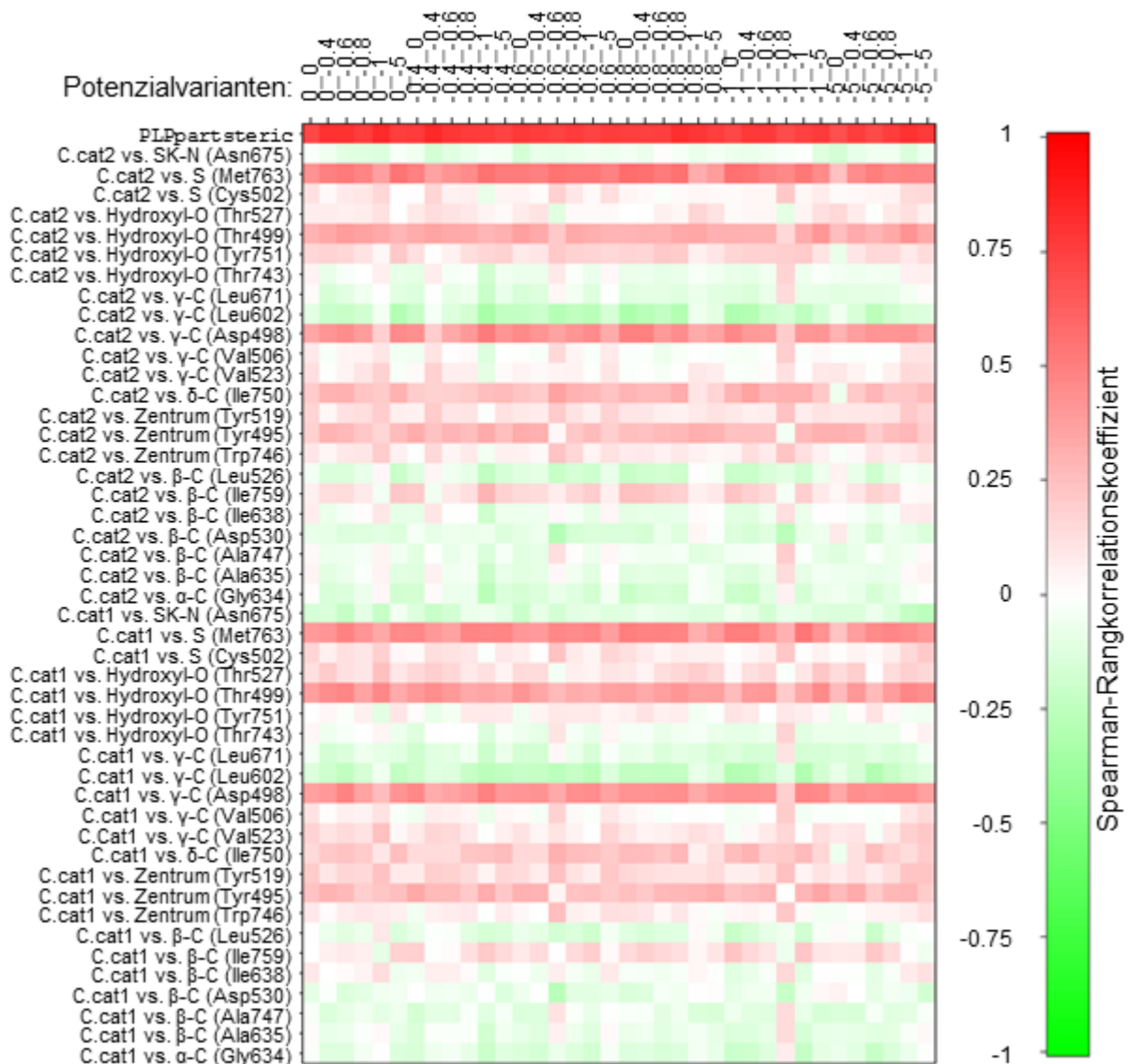


Abbildung 67: Rangkorrelation mit PLPpartsteric und gemessenen Distanzen ausgewählter Wechselwirkungen im Experiment c (Bezeichnungen siehe Abbildung 59, Seite 101 und Abbildung 66)

Es gibt in Abbildung 67 eine grundsätzliche Tendenz im Vergleich mit Abbildung 66, dass Wechselwirkungen mit sehr Score-relevanten Distanzen auch höhere Ränge belegen. Der Effekt ist allerdings nicht konsequent, wie beispielsweise die hohe Rangkorrelation der Wechselwirkungen von Tyr519 mit beiden kationischen Atomen zeigt, denn sie liegt nie im

Score-relevanten Distanzbereich. Das heißt, dass geringere Abstände zwischen Tyr519 und den kationischen Ligandenatomen tendenziell mit höheren Rängen auftreten, ohne dass diese Abstände dabei *Score*-relevant sind. Dieser Beobachtung liegt folgende Erklärung zu Grunde. Es ist möglich, dass die *Score*-Beiträge aus den Wechselwirkungen anderer Ligand- atome als der kationischen Ligandatome für Tyr519 *Score*-relevant sind. Dies kann dazu führen, dass zusätzlich zu diesen Wechselwirkungen auch die Distanzen zu den kationischen Atomen geringer sind als bei rein zufälliger Anordnung in der Bindetasche. Wenn in solchen Fällen die Wechselwirkungen insgesamt favorisierend sind, kommt es zu einer positiven Rangkorrelation zwischen den *Score*-irrelevanten Distanzen von Tyr519 zu den kationischen Ligandatomen. Dass dieser Effekt für Tyr519 besonders stark zu beobachten ist, liegt daran, dass Tyr519 im *Docking* flexibel gelassen wurde. Somit kann es potenziell mehr Konformationen einnehmen, die favorisierende *Scores* zur Folge haben als im *Docking* starre Aminosäuren. Das verdeutlicht, dass die Wahl der flexiblen Aminosäuren erheblichen Einfluss auf die *Docking*-Ergebnisse haben kann.

Interessant ist auch die überwiegend negative Rangkorrelation der Distanz von Leu602 mit dem $PLP_{partsteric}$ -*Score*-Anteil. Geringere Distanzen zwischen Leu602 und den kationischen Ligandatomen platzieren demnach die Posen tendenziell auf niedrigeren Rängen als größere Distanzen. Dabei sind diese Distanzen (siehe Abbildung 66, Seite 108) durchgehend *Score*-irrelevant. Daher muss die Erklärung für diese Beobachtung in indirekten Ursachen liegen. Offenbar treten in geringer Entfernung von Leu602 oft Wechselwirkungen zwischen Ligand- und Rezeptoratomen auf, die von der *Scoring*-Funktion als ungünstig bewertet werden. In der Tat ist Leu602 in der Nähe der polaren und geladenen Aminosäuren positioniert, welche die Metallionen komplexieren. Somit gestaltet Leu602 den Übergang zwischen dem hydrophoben und dem polaren Teil der Substratbindetasche von ShSBS (Leu602 in diesem Abschnitt entspricht L569 in Abbildung 45 auf Seite 78). Interaktionen des hydrophoben Liganden mit den polaren Atomen im diphosphatbindenden Teil der Bindetasche werden von der *Scoring*-Funktion mit ungünstigen positiven Beiträgen bewertet, was die dargelegte Beobachtung erklärt.

Der $PLP_{partsteric}$ -*Score*-Anteil selbst korreliert stark mit den Rängen der *Docking*- Posen, wie auch in Abbildung 65, Seite 107 schon dargestellt.

Einen interessanten Einblick in die geometrische Vielfalt der Interaktionen, die durch die Diversitätsforderung zu Stande kommen, bietet Abbildung 68. Gleichzeitig erlaubt die Darstellung eine Vorstellung von der *Sampling*-Dichte bei $d_{clust} = 2 \text{ \AA}$. Dabei ist zu beachten, dass sich Ligandpositionierungen und -konformationen stark ähneln können und die Lage der flexiblen Aminosäuren sich trotzdem stark unterscheiden kann, weil das d_{clust} -Kriterium nur auf den Liganden bezogen wird.

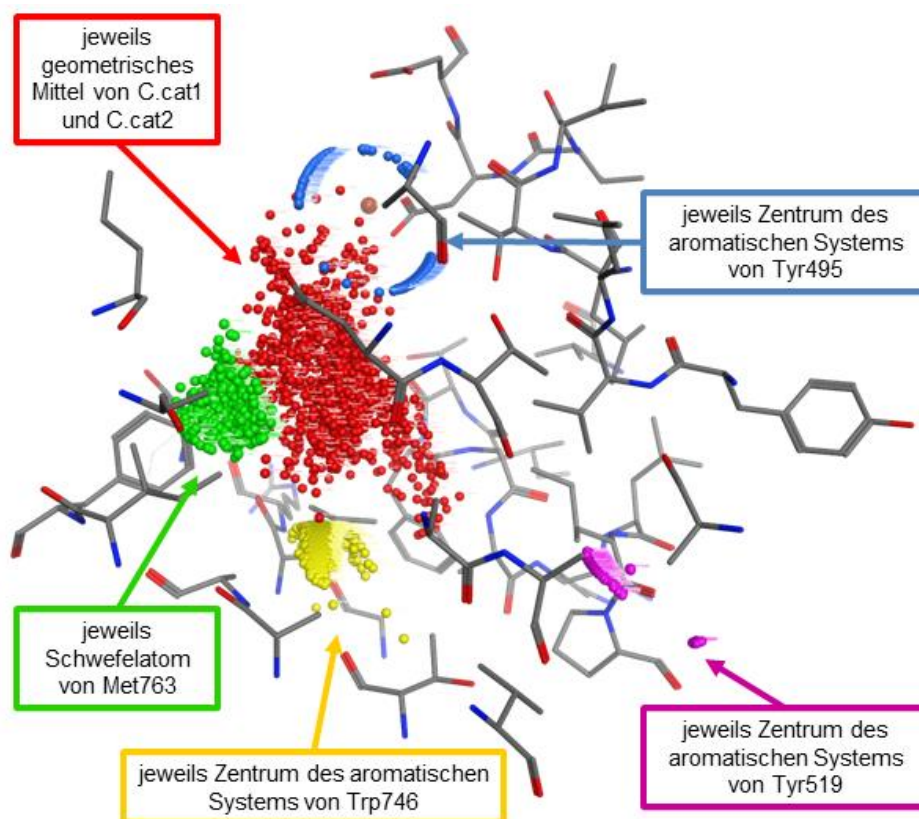


Abbildung 68: Räumliche Verteilung ausgewählter Repräsentanzpunkte in der ShSBS-Binde-tasche für alle 1000 Posen, die mit dem Potenzial $-0.4_{-0.4}$ aus dem Experiment *c* entstanden

Die zahlreichen niedrigen p-Werte im Kolmogorov-Smirnov-Test zu den Gesamt-Score- bzw. PLPpartsteric-Verteilungen (Abbildung 64, Seite 106) deuten darauf hin, dass verschiedene Potenzialvarianten zu unterschiedlichen Score-Verteilungen führen. In Abbildung 67 (Seite 109) wurde der Zusammenhang zwischen der Verwendung unterschiedlicher Potenzialvarianten und geometrischen Konsequenzen dargestellt. Allerdings wurden die Parameterwerte systematisch variiert, so dass zu erwarten wäre, dass die Änderungen von Parametern sich auch systematisch in resultierenden Interaktionsgeometrien widerspiegeln. Demzufolge müsste bei Variation eines der beiden Parameter und konstantem anderen Parameter ein monotoner Zusammenhang zwischen gemessenen Distanzen und den verwendeten Parameterwerten resultieren. Das ist aber für die Variationen von E_{met} entsprechend Abbildung 67 augenscheinlich nicht der Fall, wie die amonotonen zeilenweisen Farbverläufe in den Spalten gleichen E_{aro} -Wertes andeuten.

Eine vertiefende Untersuchung des Zusammenhanges zwischen Potenzialreihen und den Rangkorrelationen gemessener Wechselwirkungsdistanzen ist in Abbildung 69 veranschaulicht. In dieser Darstellung sind die durch * abgekürzten Parameterreihen (0; -0,4; -0,6; -0,8; -1; -5) korreliert gegen die Rangkorrelationskoeffizienten gemessener Distanzen (bzw. des PLPpartsteric-Score-Anteils). Wenn eine Wechselwirkung mit geringerer Distanz zu günstigeren Scores führt, so sollte sich dieser Zusammenhang über Potenzialreihen entweder monoton verstärken oder monoton abschwächen, und zwar für alle Potenzialreihen unter Variation jeweils eines der beiden Parameter E_{met} und E_{aro} . Demzufolge müssten sich die Farben der linken sechs Spalten bzw. der rechten sechs Spalten in Abbildung 69 zeilenweise ähneln. Dass dies nicht zu beobachten ist, beweist, dass die Geometrien der erzeugten Docking-Posen nicht ausschließlich von der

verwendeten *Scoring*-Funktion abhängig sind. Aus diesem Grund ist hier – wie im Fall der *Docking*-Experimente in das Limonensynthesemodell – davon auszugehen, dass die Diversitätsforderung in niedrigeren Rängen deutlich weniger favorisierte Wechselwirkungen bzw. Wechselwirkungskombinationen zur Folge hat, welche die theoretisch zu erwartenden Korrelationen stören.

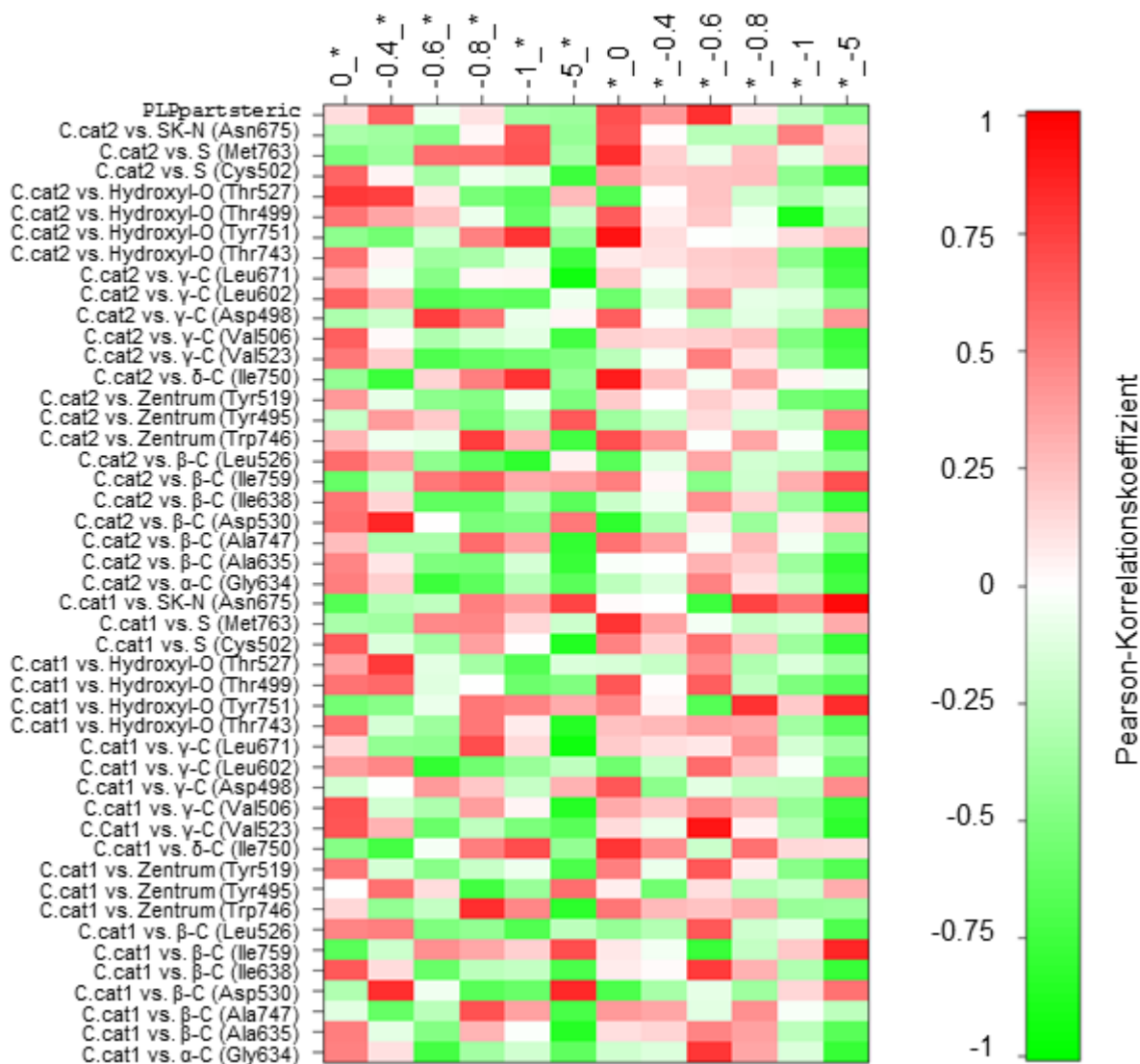


Abbildung 69: Korrelation der Parametervariation im Experiment *c* (die Folge (0; -0,4; -0,6; -0,8; -1; -5)) mit Rangkorrelationskoeffizienten von *PLPpartsteric* und gemessenen Distanzen ausgewählter Wechselwirkungen aus den entsprechenden Potenzialvarianten (Bezeichnungen siehe Abbildung 59, Seite 101 und Abbildung 66, Seite 108; das *-Symbol steht in der Bezeichnung der Potenzialreihen für die Reihe der Parameter des jeweils anderen Parameterwertes, wie sie im Experiment verwendet wurden)

In diesem Abschnitt wurde bisher das Experiment *c* diskutiert, in welchem weder komplett erschöpfend noch unabhängig gedockt wurde. Es ist daher davon auszugehen, dass die stark unterschiedlichen Korrelationen gemessener Distanzen mit dem Gesamt-Score über die Potenzialvarianten hinweg durch unterschiedliche Mengen von *Docking*-Posen zu Stande kommen. Insbesondere wird hier postuliert, dass in solchen großen Posenmengen die durch das *Docking* ausgewählte Untermenge der Grundgesamtheit stark abhängig von den ersten

erzeugten Posen ist. Allein die Auswahl der ersten Pose schränkt den Suchraum für weitere Posen durch das Diversitätskriterium ein. Dieser Effekt setzt sich in den Folgeiterationen fort.

Dass bei identischen *Docking*-Parametern trotzdem unterschiedliche Untermengen mit dem *Sampling* ausgewählt werden, liegt am stochastischen Verfahren der Optimierung. Einen Beweis für die Unterschiedlichkeit der Posenauswahl in *Docking*-Experimenten mit identischem Parametersatz bieten die beiden erschöpfenden *Docking*-Experimente \mathcal{H} und \mathcal{J} . Die Diversitätsforderung wurde auf $d_{clust} = 1 \text{ \AA}$ gesetzt und mit den verschiedenen Parameterkombinationen wurden sehr unterschiedliche Posenanzahlen erreicht, bis der PLANTS-Algorithmus terminiert, wie Tabelle 21 wiedergibt.

Tabelle 21: Anzahl der bei Termination des PLANTS-Algorithmus zurückgegebenen Docking-Posen (Experiment \mathcal{H} bzw. Experiment \mathcal{J})

$E_{aro} \setminus E_{met}$	-0,4	-0,45	-0,5	-0,55	-0,6
-0,4	4.491 4.221	4.141 3.794	4.204 4.005	4.593 4.136	4.404 4.399
-0,45	4.197 4.634	3.985 4.269	4.271 4.058	4.257 4.445	4.357 4.365
-0,5	4.215 4.194	4.256 4.424	4.514 4.556	4.371 4.129	4.645 4.359
-0,55	4.457 4.565	4.167 4.143	4.194 4.109	4.098 4.520	4.367 4.008
-0,6	4.358 4.616	3.966 4.269	4.036 4.019	3.895 4.576	4.076 4.360

Ähnliche relative Abweichungen in der Anzahl der erzeugten *Docking*-Posen wurden auch in den anderen erschöpfenden *Docking*-Experimenten gefunden. Hinweise auf die quantitative Natur der Abhängigkeit erzeugter Posen von im Algorithmus vorher erzeugten Posen liefern Datenvisualisierungen wie Abbildung 70. Der $PLP_{partsteric}$ -Score-Anteil bezieht sich in dieser Art von Darstellung nicht auf die konkrete Wechselwirkung, sondern auf den entsprechenden Term des Gesamt-Score der jeweiligen Posen. Auch wenn die $PLP_{partsteric}$ -Datenpunkte teilweise die Abstandsdatenpunkte verdecken, ist die deutliche Tendenz zu erkennen, dass die untersuchte Wechselwirkung erst ab Rängen von ca. 1.000 abwärts Score-relevante Distanzen aufweist. Es handelt sich hierbei offenbar um eine der Wechselwirkungen, die nur dann Score-relevante Distanzen zeigt, wenn die Diversitätsforderung keine anderen Möglichkeiten lässt. Abbildung 70 ist charakteristisch für Wechselwirkungen, die in hochrangigen Posen keine Rolle spielen. Dies wird in diesem Fall durch die negative Rangkorrelation der Distanzen von -0,42 unterstrichen. Die Rangkorrelation des $PLP_{partsteric}$ -Score-Anteils ist im Allgemeinen für derartige Daten größer als 0,8 (hier 0,91).

Analysen der Ähnlichkeit von Verteilungen ausgewählter Abstände mit dem Kolmogorov-Smirnov-Test über Potenzialvarianten hinweg deuten darauf hin, dass sich die zu Grunde liegenden stochastischen Prozesse sehr wahrscheinlich unterscheiden (ohne Abbildung). Erneut kann jedoch die Begründung der Abhängigkeit von Posen untereinander durch die Diversitätsforderung angeführt werden.

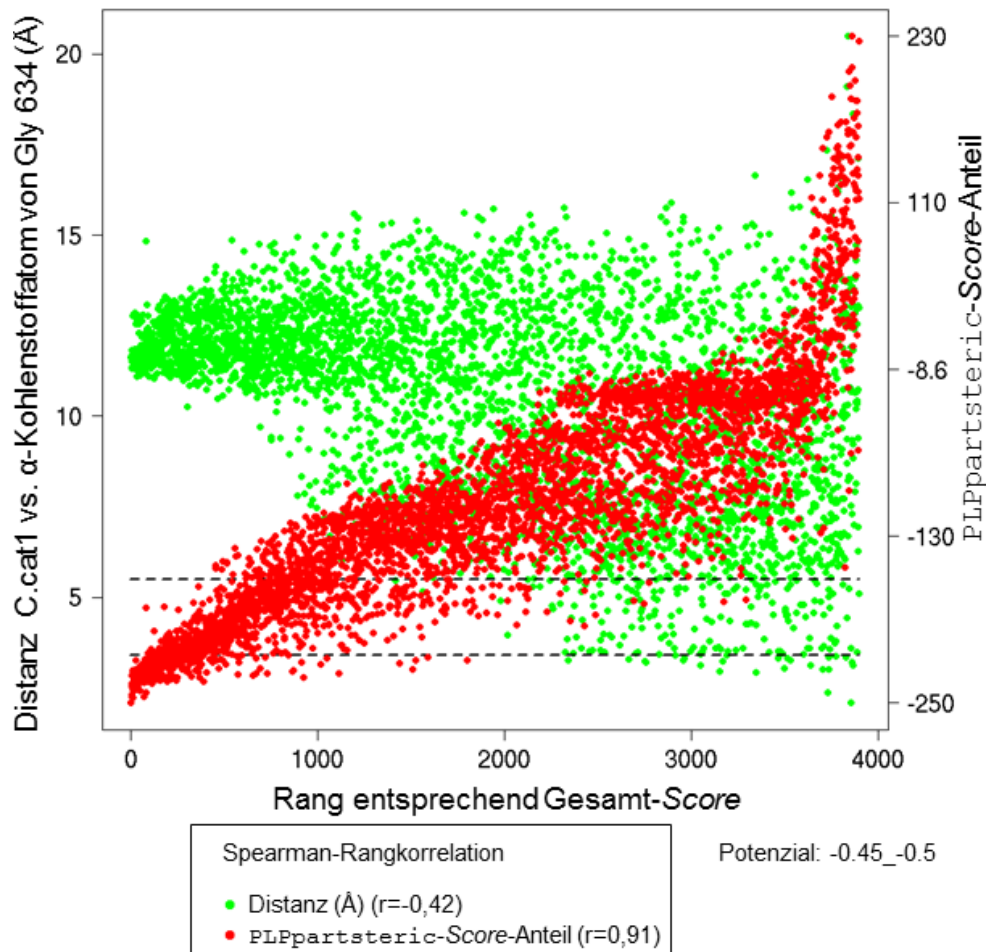


Abbildung 70: Distanzen zwischen C.cat1 und dem α -Kohlenstoffatom von Gly634 und PLPpartsteric-Score-Anteile entsprechend der Rangsortierung (Experiment #). Der Bereich zwischen den beiden gestrichelten Linien entspricht dem Score-relevanten Distanzbereich. Man beachte die unterschiedlichen Skalen der vertikalen Achsen.

Abbildung 71 hingegen ist charakteristisch für Wechselwirkungen, die in hochrangigen Posen Score-relevant sind. Die Rangkorrelation der Distanzen kann hierfür als Indikator gelten.

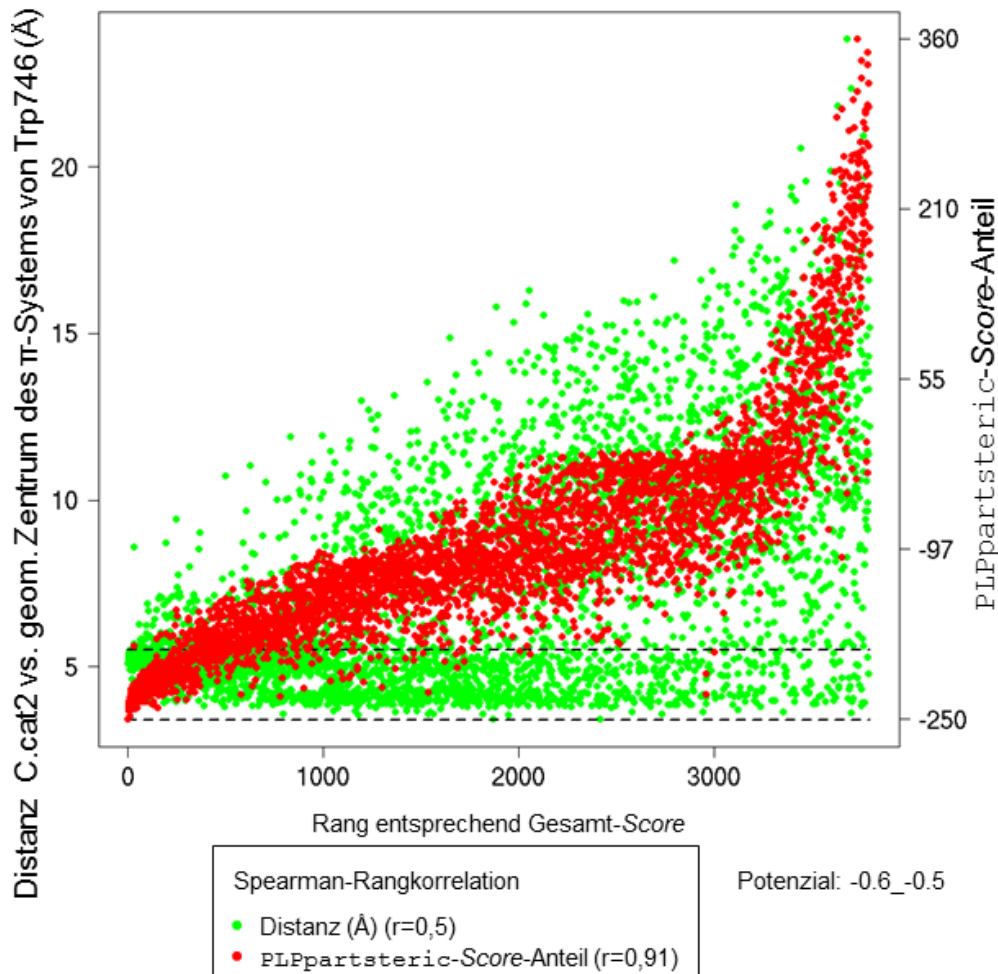


Abbildung 71: Distanzen zwischen C.cat2 und Trp746 und PLPpartsteric-Score-Anteile entsprechend der Rangsortierung (Experiment \mathcal{J}). Der Bereich zwischen den beiden gestrichelten Linien entspricht dem Score-relevanten Distanzbereich. Man beachte die unterschiedlichen Skalen der vertikalen Achsen.

Ein interessanter Sachverhalt wird in Abbildung 72 deutlich, in der die Score-Anteile für die Top 100 Posen des Experiments \mathcal{J} für ein ausgewähltes Potenzial aufgetragen sind: Große Abweichungen vom mittleren PLPpartsteric-Score-Anteil werden durch große, entgegengesetzte Abweichungen des CHEMPLP_PLP_PROT-Score-Anteils kompensiert. Hierbei darf keine ursächliche oder direktionale Verknüpfung impliziert werden. Allerdings zeigt diese Darstellung, dass Aminosäureseitenketten in der Bindetasche des Proteins *per se* bezüglich der Scoring-Funktion ungünstige Konformationen einnehmen und stattdessen der Ligand besser bewertete Konformationen erhalten kann. Das entspricht einem kleinen Einblick in die multidimensionale Optimierung der Rezeptor-Ligand-Wechselwirkungen.

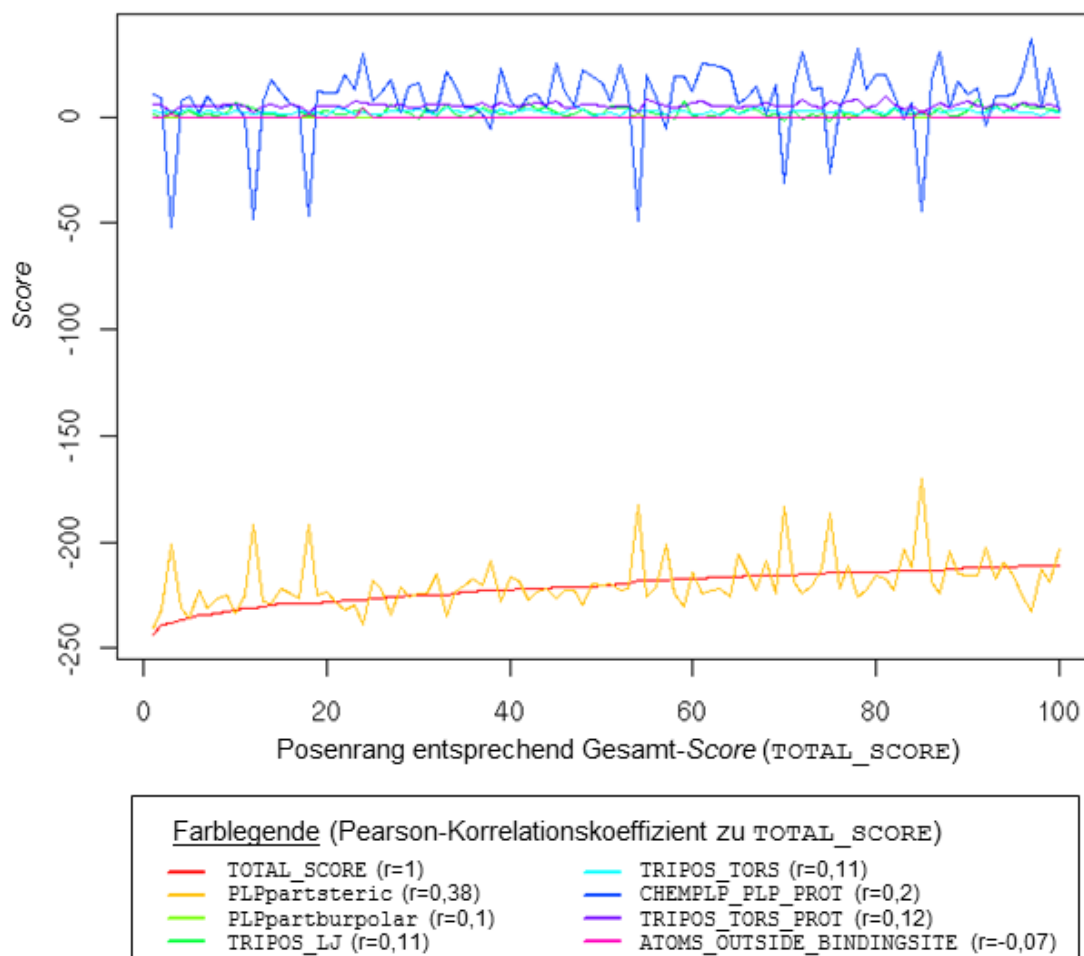


Abbildung 72: Auftragung der *Score*-Zusammensetzung für die Top 100 Posen, die mit dem Potenzial -0.45_-0.55 erzeugt wurden, und Pearson-Korrelationskoeffizienten zwischen *Score*-Anteilen und dem Gesamt-Score (Experiment \mathcal{J} , zur besseren Verdeutlichung der Variabilität als kontinuierliche Auftragung); Bedeutung der Termbezeichnungen: siehe Abbildung 65, Seite 107

Die bisher dokumentierten Ergebnisse deuten darauf hin, dass der Unterschied zwischen Wechselwirkungsdistanzen, die in hochrangigen Posen vorkommen und Wechselwirkungsdistanzen, die bevorzugt in Posen unterer Ränge auftreten, nicht mit der Wahl der variierten Parameter zusammenhängt. Vielmehr lässt sich aus Darstellungen wie Abbildung 67, Seite 109, schließen, dass im Wesentlichen sterische Zugänglichkeit für Wechselwirkungen zwischen dem Liganden und bestimmten Aminosäuren verantwortlich ist. Diesen Schluss implizieren alle bisher angeführten Ergebnissen aus diversitätsorientiertem *Docking*. Die absolute Größe der variierten Parameter hat ebenfalls höchstens untergeordneten Einfluss.

Ein Experiment mit unabhängigem *Docking* in ShSBS (Experiment \mathcal{F}) gibt Aufschluss darüber, welche Wechselwirkungen *Score*-relevant sind und zeigt gleichzeitig eine Homogenität zwischen verschiedenen Potenzialvarianten, wie sie auch bei abhängigem *Docking* (vgl. Abbildung 66, Seite 108) auftritt. Abbildung 73 verdeutlicht, dass häufig insbesondere Wechselwirkungen zwischen den kationischen Kohlenstoffatomen und dem Schwefelatom von Met763 sowie den aromatischen Systemen von Tyr495 und Trp746 gefunden werden. Der *Score*-Relevanz-Unterschied zwischen den Distanzen des Hydroxylgruppen-Sauerstoffatoms von Thr499 zu C.cat1 bzw. C.cat2 zeigt, dass C.cat2 diese

Wechselwirkung seltener eingetreten als C.cat1. Dies deutet daraufhin, dass der Ligand bezüglich Thr499 nicht beliebig positioniert werden kann, ohne dass *Score*-Einbußen folgen. Der gleiche Effekt ist auch für Val523 zu beobachten. Umgekehrt ist es bei der Distanz zwischen dem γ -Kohlenstoffatom von Asp498 zu den kationischen Kohlenstoffatomen des Liganden: Hier ist C.cat2 öfter in *Score*-relevanter Wechselwirkungsdistanz zu finden als C.cat1.

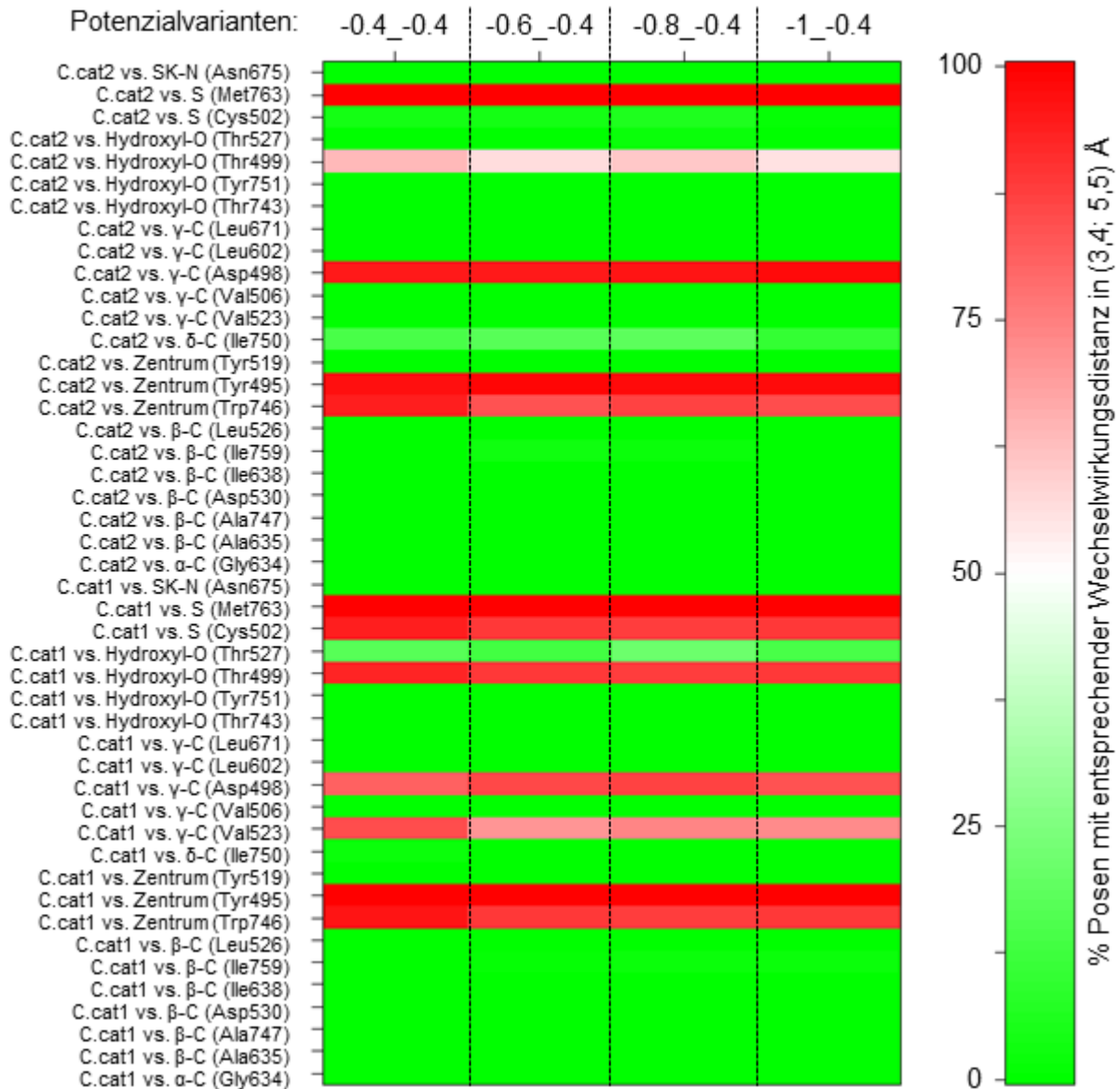


Abbildung 73: Farbliche Veranschaulichung des jeweiligen Posenanteils im Experiment \mathcal{E} , bei dem die betrachteten Wechselwirkungen innerhalb des *Score*-relevanten Distanzbereichs liegen. (Abkürzungen: siehe Abbildung 59, Seite 101 und Abbildung 66, Seite 108)

In Anbetracht der bisherigen Ergebnisse zu abhängigem *Docking* und dem in Abschnitt 4.3.1 (Seite 93) vorgestellten Vorversuch ist davon auszugehen, dass das *Sampling* im Fall der unabhängigen *Docking*-Experimente auch die Posen (bzw. geometrisch sehr ähnliche Posen) findet, die im diversitätsorientierten *Docking* hochrangig waren. Ein zusätzlicher Hinweis dafür ist der Vergleich von Gesamt-Scores in Tabelle 22.

Tabelle 22: Gesamt-Scores in diversitätsorientierten vs. unabhängigen Docking-Experimenten

Score von:	Rang 1		Rang 100
F (100-fach unabhängig)	-250,209	...	-230,79
H (erschöpfend mit $d_{clust} = 1 \text{ \AA}$)	-243,169	...	-210,071
J (erschöpfend mit $d_{clust} = 1 \text{ \AA}$, identisch zu H)	-236,104	...	-204.066

Dieses Beispiel zeigt, dass der Gesamt-Score der Pose bei einmaligem *Docking* im hochrangigen Bereich des diversitätsorientierten *Docking* liegt. Das ist ein empirischer Beweis für ein breites *Sampling* des Ergebnisraums durch den Ameisenalgorithmus. Unabhängiges *Docking* eignet sich somit besser als abhängiges *Docking* um systematische geometrische Konsequenzen für *Docking*-Posen bei Potenzialvariation zu untersuchen.

Die *Score*-Werte in Tabelle 22 deuten ferner darauf hin, dass das Diversitätskriterium in Kombination mit dem Finden lokaler Optima der *Scoring*-Funktion dazu führen kann, dass Posen mit noch besserer Bewertung nicht gefunden werden.

Erschöpfendes *Docking* ist in diesem Zusammenhang ein künstliches Konstrukt zur Untersuchung der Auswirkung von Potenzialveränderungen. In der Praxis ist es unüblich, aus einem *Docking*-Algorithmus mehr als 10 *Docking*-Posen pro Molekülstruktur zu fordern.

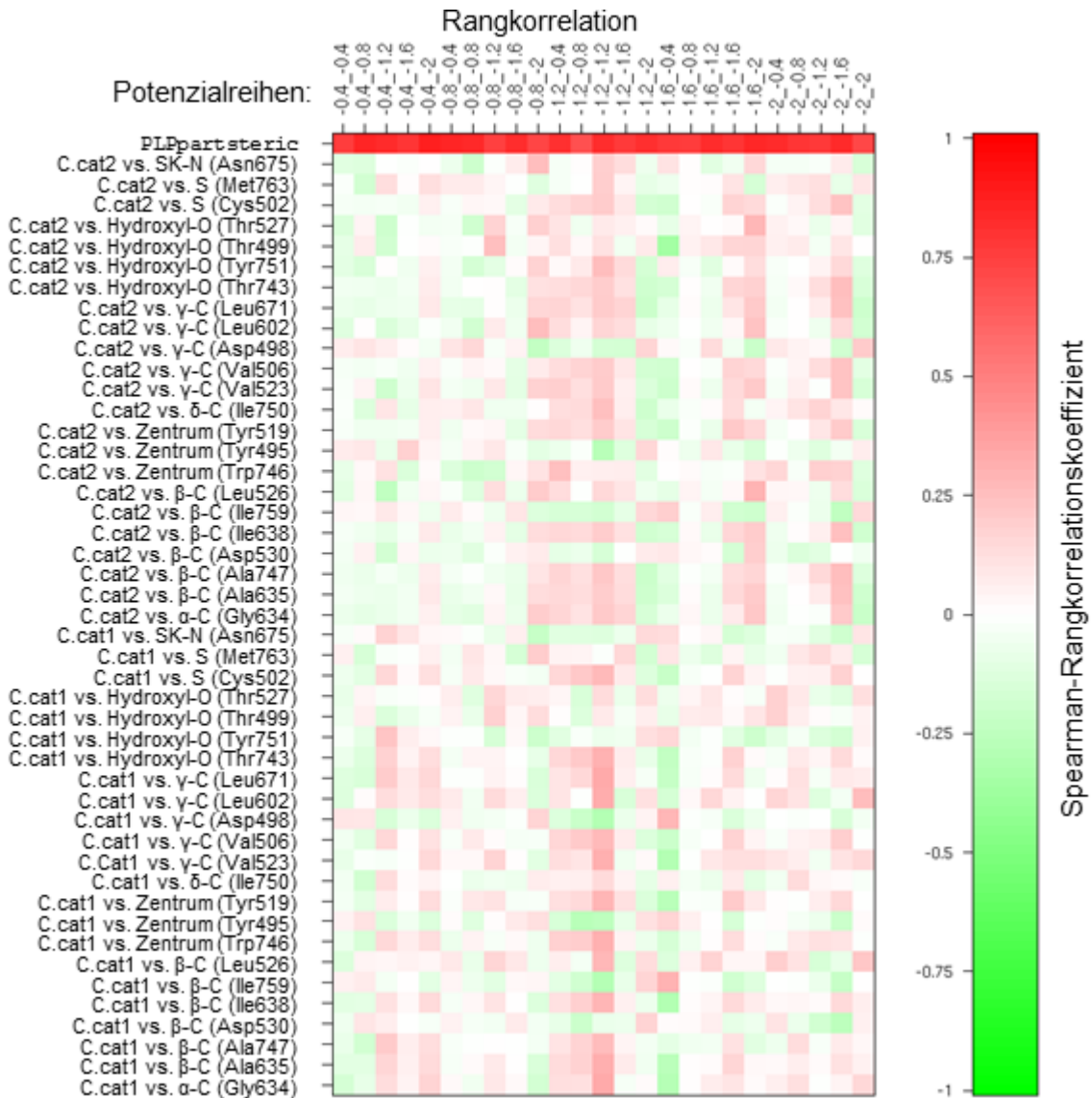


Abbildung 74: Rangkorrelation mit PLPpartstERIC-Score-Anteil und Distanzen ausgewählter Wechselwirkungen im Experiment R (Bezeichnungen siehe Abbildung 59, Seite 101 und Abbildung 66, Seite 108)

Die in Abbildung 74 dargestellte Rangkorrelation der beobachteten Distanzen ausgewählter Wechselwirkungen zeigt Variabilität der Korrelation auf niedrigem Niveau. Die Rangkorrelation des PLPpartstERIC-Score-Anteils ist durchweg sehr hoch und die Pearson-Korrelation zwischen Gesamt-Score und PLPpartstERIC-Score-Anteil liegt im Experiment R stets zwischen 0,63 und 0,84. Optisch heben sich einzelne Potenzialvarianten ab, wie beispielsweise -1.2_-1.2 als Potenzial mit relativ hoher Rangkorrelation über fast alle Wechselwirkungsdistanzen oder das Potenzial -1.6_-0.8, bei dem fast alle Korrelationskoeffizienten nahe Null sind. Um herauszufinden, ob diese Auffälligkeiten reproduziert werden können, wurde das *Docking*-Experiment wiederholt.

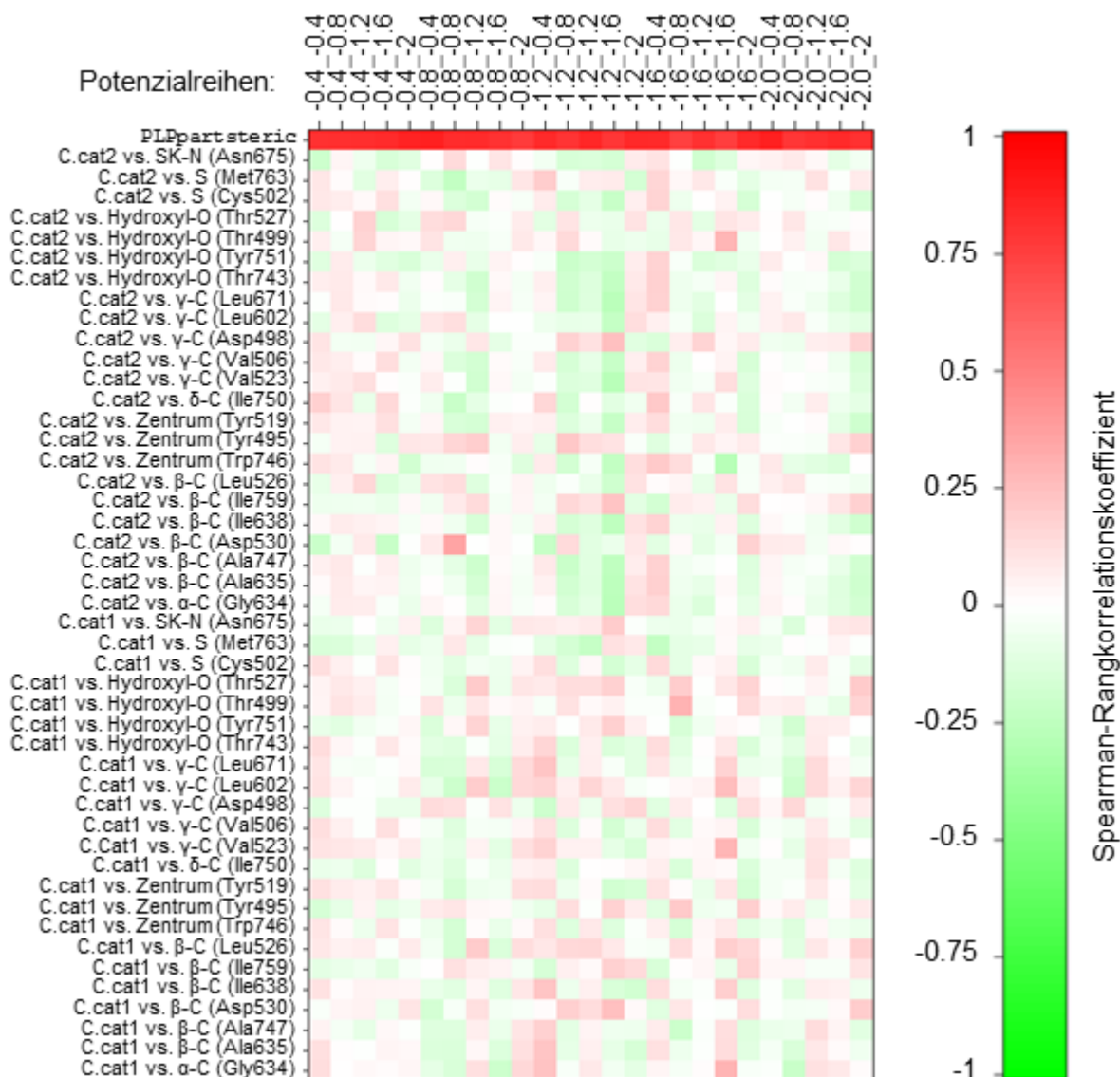


Abbildung 75: Rangkorrelation mit PLPpartstERIC-Score-Anteil und Distanzen ausgewählter Wechselwirkungen im Experiment *s* (Bezeichnungen siehe Abbildung 59, Seite 101 und Abbildung 66, Seite 108)

Das identische Experiment *s* zeigt ebenfalls eine allgemein geringe Rangkorrelation der Distanzen der ausgewählten Wechselwirkungen (Abbildung 75), bestätigt aber nicht die vermeintlichen Auffälligkeiten aus Abbildung 74. Aus dem Vergleich der Ergebnisvisualisierungen in Abbildung 74 und Abbildung 75 ist zu schlussfolgern, dass die Rangkorrelationen gemessener Distanzen über die große Zahl unabhängiger *Docking*-Posen wenig aussagekräftig sind. Die stochastischen Elemente des PLANTS-Programmes führen demzufolge zu einer Geometrievielfalt der Posen des gedockten (*Z,Z*)-Farnesylkations, die durch die verwendeten Potenzialvarianten nicht so stark eingeschränkt wird, dass sich die Unterschiede bei 1000-fach unabhängigem *Docking* in Rangkorrelationen der beobachteten Distanzen niederschlagen. Eine Quantifizierung dieser Schlussfolgerung wäre nur über zahlreiche weitere Durchführungen von *Docking*-Experimenten analog *R* und *s* zu erhalten.

Die Rangkorrelation des PLPpartstERIC-Score-Anteils ist allerdings erneut konsequent hoch und die Pearson-Korrelation zwischen Gesamt-Score und PLPpartstERIC-Score-Anteil liegt potenzialweise zwischen 0,67 und 0,82.

Insbesondere in Anbetracht der Veranschaulichungen in Abbildung 74 und Abbildung 75 (Seite 119 bzw. 120) ist davon auszugehen, dass die verwendeten Potenzialvarianten keine wesentlichen Änderungen der resultierenden *Docking*-Posen zur Folge haben, deren Ursache die Potenzialvarianten selbst sind. Die durchgeführten Parametervariationen von E_{aro} und E_{met} mit Werten im Intervall [-2; 0] gelten als realistischer Wertebereich, da andere Wechselwirkungen mit Werten in demselben Bereich modelliert werden. Der entsprechende Parameterwert E für Wasserstoffbrückenbindungen ist -1,0, für polar-unpolare Atomkontakte -0,1, für Metallkoordination -0,4 und für repulsive Wechselwirkungen +0,1 (Definition der Wechselwirkungen jeweils entsprechend Korb *et al.*, 2009, ²⁷⁹).

Die Kolmogorov-Smirnov-Tests zur Unterscheidung von Distanzverteilungen unter verschiedenen Potenzialvarianten sind bei unabhängigen *Docking*-Experimenten nicht schlüssig. Vielfach existieren p-Werte über 0,8, aber ohne erkennbare Regelmäßigkeit sind auch häufig p-Werte unter 0,1 anzutreffen. Eine Erklärung für dieses Verhalten kann in der Kolmogorov-Smirnov-Methode selbst liegen: Bei den Experimenten mit unabhängigem *Docking* ist die Anzahl der *Docking*-Posen im Vergleich zu diversitätsorientiertem *Docking* gering. Das kann dazu führen, dass der Kolmogorov-Smirnov-Test im Falle von unabhängigem *Docking* identische Verteilungen als wahrscheinlicher beurteilt als bei großen Stichproben wie im Fall des diversitätsorientierten *Docking*.

Als eine Beschreibung wesentlicher Charakteristika hochrangiger *Docking*-Posen des (Z,Z)-Farnesylkations in der Bindetasche des ShBS-Modell über alle Potenzialvarianten hinweg soll Abbildung 76 dienen. Die geometrische Positionierung des Liganden zwischen den drei dargestellten Aminosäureseitenketten macht deutlich, dass die realistische Modellierung von Kation- π -Wechselwirkungen durch ein stückweise lineares Potenzial limitiert ist. Die Modellierung des (Z,Z)-Farnesylkations mit zwei C.cat-Kohlenstoffatomen müsste als optimale Ausrichtung des Liganden zu Aromaten theoretisch eine etwa parallele Anordnung der Ebenen der π -Systeme zur Folge haben. Allerdings können andere Wechselwirkungen diese Geometrie stören – wie das Beispiel von Met763 zeigt – so dass keine Parallelität zu Stande kommt.

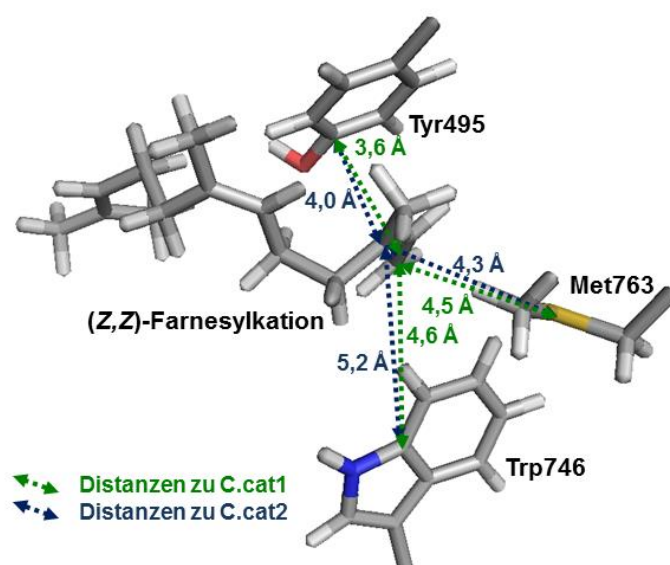


Abbildung 76: Ausschnitt aus der Top-*Docking*-Pose für das Potenzial -0.4_-0.4 im Experiment *s* (erstellt unter Benutzung von PyMOL). Das kationische Allylsystem ist aufgrund der Atomtypdefinition in PLANTS planar (siehe Abbildung 19 B, Seite 52).

Nicht gezeigte Darstellungen analog zu Abbildung 68 (Seite 111) über die Ergebnisse von unabhängigem *Docking* zeigen sehr deutlich, dass nur sehr geringe geometrische Abweichungen zwischen den Posen bestehen. Die wesentlichen Wechselwirkungen entsprechen fast ausschließlich den in Abbildung 76 veranschaulichten. Dies bestätigen auch die durchgeführten *Cluster*-Analysen zur Positionierung der flexiblen Aminosäureseitenketten und der kationischen Kohlenstoffatome des Liganden. Diese liefern dann getrennte *Cluster*, wenn die Distanz aller fünf flexiblen Koordinaten untereinander (geometrisches Mittel von C.cat1 und C.cat2, Schwefelatom von Met763, Zentren der aromatischen System von Tyr495, Tyr519 und Trp746) größer als eine gegebene Distanz ist. Diese Distanz wurde im Rahmen der Analysen in 1 Å-Schritten von 15 Å auf 1 Å gesenkt. Die Trennung in zwei *Cluster* erfolgt über alle Potenzialvarianten hinweg fast ausnahmslos beim Wechsel der Distanz von 6 Å auf 5 Å. Die beiden resultierenden *Cluster* liegen räumlich meist eng beieinander (ohne Abbildung, beispielhafte Daten: A 29, Seite 227). Für kleinere Distanzen werden bis zu fünf *Cluster* gebildet, die sich hauptsächlich in der Positionierung des Prenylrestes fern des kationischen Allylsystems des (Z,Z)-Farnesylkations unterscheiden.

Die konsequente Positionierung des kationischen Teils des Liganden im Dreieck der Aminosäuren Tyr495, Tyr519 und Trp746 ist unrealistisch, da die Diphosphathydrolyse am gegenüberliegenden Ende der Bindetasche stattfindet und eine umgekehrte Positionierung des Liganden herbeiführen würde. Im Hinblick auf folgende Zyklisierungsschritte der enzymkatalysierten Reaktion (siehe 0, Seite 84 ff.) sind *Docking*-Posen des (Z,Z)-Farnesylkations interessant, in denen eine Vorpositionierung der allylischen Kationatome C.cat1 und C.cat2 parallel zur mittleren Doppelbindung des Farnesylrestes vorliegt. Solche Posen werden im unabhängigen *Docking* nicht zurückgegeben und im diversitätsorientierten *Docking* signifikant schlechter bewertet (Daten nicht gezeigt). Eine Hauptursache für die umgekehrte Lage des kationischen Intermediats in der Bindetasche sind die summarisch sehr vorteilhaften Wechselwirkungen zu den drei genannten Aminosäuren. Die *Scores* dieser Wechselwirkungen summieren sich über mehrere aromatische Atome auf. Somit liefern sie einen betragsmäßig deutlich größeren *Score*-Beitrag, als einfache polare und elektrostatische Wechselwirkungen dies könnten.

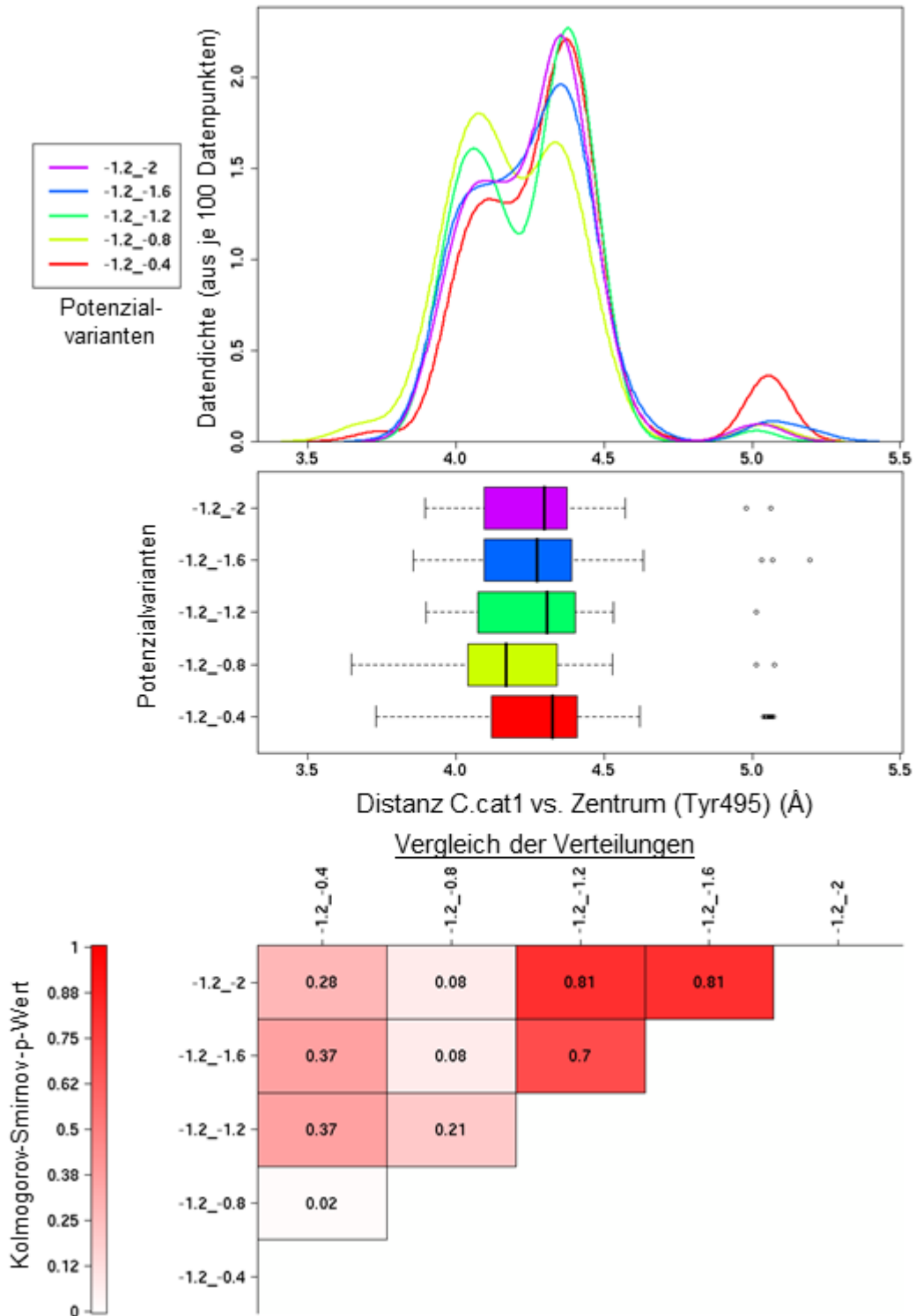


Abbildung 77: Vergleich von Distanzverteilungen über ausgewählte (met-) Potenzialvarianten der (potenziellen) Wechselwirkung zwischen C.cat1 und dem geometrischen Zentrum des π -Systems von Tyr495 (Experiment \mathcal{R}) – charakteristisches Beispiel für Score-relevante Interaktionen

Als allgemeine Charakterisierung der *Docking*-Posen des (Z,Z)-Farnesylkations in die ShSBS-Bindetasche dient Abbildung 73 (Seite 117). Dabei besteht für Score-relevante Wechselwirkungen ebenso wenig wie für Score-irrelevante Wechselwirkungen ein monotoner Zusammenhang zwischen verwendeten Parameterwerten und gemessenen Distanzen. Das verdeutlichen beispielhaft Abbildung 77 und Abbildung 78. In beiden Fällen

handelt es sich um die Veranschaulichung der Distanzverteilungen der Wechselwirkungen von C.cat1 mit den aromatischen Ringen von Tyrosinseitenketten. Deutlich tritt im Vergleich beider Abbildungen der *Score*-Relevanz-Unterschied hervor. Tyr495 und Tyr519 wurden beim *Docking* als flexibel behandelt, aber bereits das Potenzial $-0.4_{-0.4}$, welches die neu modellierten Wechselwirkungen am wenigsten gegenüber den anderen Potenzialen favorisiert, liefert konsistent die dreifache Erkennung des allylischen Kations entsprechend Abbildung 76. Da sich in der Umgebung von Tyr519 keine Wechselwirkungspartner für die karbokationischen Atomen des Liganden befinden, die eine bessere Bewertung zur Folge hätten, bleibt die Wechselwirkung zwischen den karbokationischen Atomen des Liganden und Tyr519 *Score*-irrelevant.

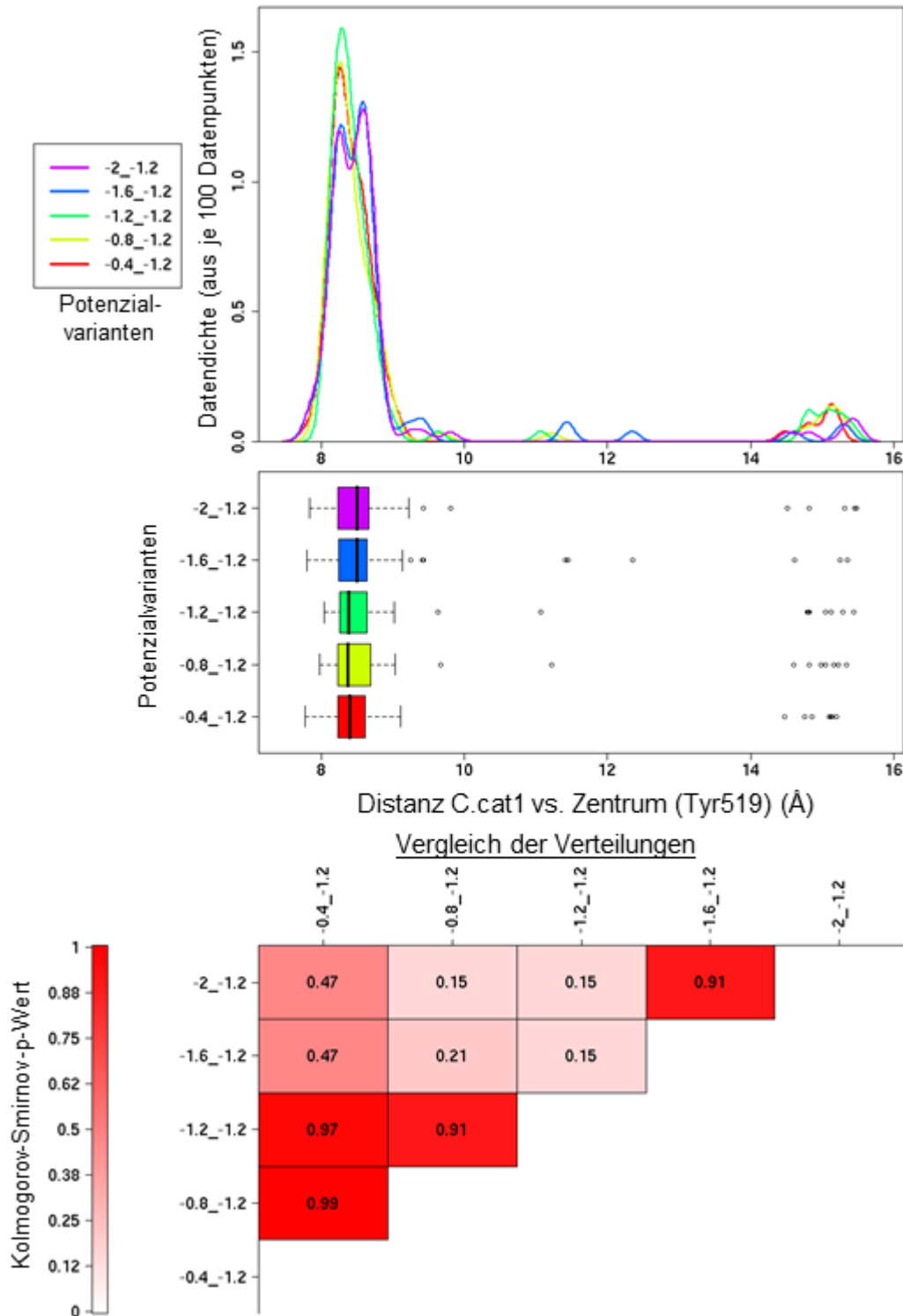


Abbildung 78: Vergleich von Distanzverteilungen über ausgewählte (aro-) Potenzialvarianten der (potenziellen) Wechselwirkung zwischen C.cat1 und dem geometrischen Zentrum des π -Systems von Tyr519 (Experiment \mathcal{R}) – charakteristisches Beispiel für *Score*-irrelevante Interaktionen

Die Korrelationen der Parametervariation in den Experimenten \mathcal{R} und \mathcal{S} mit gemessenen Distanzen und dem $PLP_{partsteric}$ -*Score*-Anteil zeigen keine Hinweise darauf, dass verschiedene Potentiale statistisch signifikant zu verschiedenen *Docking*-Posen führen (siehe A 29, Seite 227 und A 31, Seite 230).

4.3.6 Diskussion der *Docking*-Experimente und ihrer Ergebnisse

Im Rahmen dieser Arbeit wurden umfangreiche Beschreibungen der Folgen der Verwendung unterschiedlicher Potenzialvarianten für das *Docking* karbokationischer Intermediate in Terpensynthasebindetaschen vorgenommen. Die statistische Vorgehensweise ist hier notwendig, um einerseits im Fall des unabhängigen *Docking* die Variabilität der stochastisch beeinflusst erzeugten Ergebnisse zu untersuchen und um andererseits große Posenmengen aus diversitätsorientiertem *Docking* analysieren zu können. Abgesehen von den verwendeten *Scoring*-Funktion-Varianten gibt es im *Docking* kein objektives Maß dafür, wie realistisch eine *Docking*-Pose ist. Über verschiedene *Docking*-Programme hinweg gibt es auch unterschiedliche Ansprüche an die Qualitäten sinnvoller *Docking*-Posen. Das spiegelt sich in zahlreichen Versuchen wider, *Docking*-Programme miteinander zu vergleichen (beispielsweise ^{326,355–361,425,426}). Dabei ist die Definition der angewendeten Vergleichskriterien kritisch für die Bewertung der Resultate.^{355,427–430} Cole und Kollegen (³⁵⁵) nennen unter anderem folgende Herausforderungen, denen ein Vergleich von *Docking*-Programmen standhalten sollte:

- Um Unterschiede in Anreicherungsraten beim Durchsuchen von Datenbanken und bei der Erfolgsquote für Posenvorhersagen mit Signifikanzen zu untermauern, sind statistische Herangehensweisen notwendig.
- Einfache numerische Abstandsmaße wie RMSD müssen sehr sorgfältig interpretiert werden und können durch Wechselwirkungsmaße und visuelle Ergebnisinspektion ergänzt werden.
- Testdatensätze haben Anforderungen bezüglich ihrer Diversität und Zuverlässigkeit der Daten aus Laborexperimenten.
- Packungseffekte von aufgeklärten Kristallstrukturen können eine Rolle in der Zuweisung von Bindungsmodi spielen.
- Die Methode zur Erzeugung von initialen Rezeptor-Ligand-Komplexen kann drastische Auswirkungen auf die *Docking*-Ergebnisse haben.
- Für einen fairen Vergleich müssen verschiedene Programme gleich komplexe Probleme in vergleichbarer Zeit lösen.

Korb *et al.* stellten am Beispiel von PLANTS fest, dass *Scoring*-Funktionen nur dann objektiv bezüglich des Findens nativer Rezeptor-Ligand-Wechselwirkungsgeometrien miteinander verglichen werden können, wenn die Konformationserstellung bereits unter Verwendung der entsprechenden *Scoring*-Funktion stattfindet.⁴³¹ Cheng und Kollegen (⁴³²) definieren drei Größen, um basierend auf experimentellen Ergebnissen *Scoring*-Funktionen zu vergleichen:

1. *Docking Power*: Fähigkeit, die Pose des Röntgenkristallstruktur-Komplexes aus einer computergenerierten Auswahl von Rezeptor-Ligand-Posen herauszufinden
2. *Ranking Power*: Fähigkeit, verschiedene Liganden, die am gleichen Rezeptor binden, auf der Basis der bekannten Bindemodi entsprechend ihrer Bindeaffinität zu ordnen
3. *Scoring Power*: Fähigkeit, *Scores* zu berechnen, die bei bekannten Rezeptor-Ligand-Komplexen (vorzugsweise linear) mit experimentell ermittelten Bindeaffinitäten korrelieren

Für die in der vorliegenden Arbeit angestellten Untersuchungen kommen jedoch die Kriterien nach Cheng *et al.* für die Auswertung nicht in Frage, weil weder Bindeaffinitäten für Intermediate bekannt sind, noch experimentelle Rezeptor-Ligand-Strukturen existieren. Die

komparativen statistischen Analysen der modellierten Wechselwirkungen sind ein Beispiel für den Versuch einer objektiven Beschreibung des Verbesserungspotenzials von *Scoring*-Funktionen.

Die in der vorliegenden Arbeit beschriebenen *Docking*-Experimente zeigen geringe Auswirkungen auf die Positionierung des intermediären Kations in den resultierenden Posen unter Variation der *Scoring*-Funktion. Verschiedene Schwachpunkte der benutzten Modellierung wurden identifiziert:

- Sinnvolle Geometrien werden durch ein einfaches abstandsabhängiges Potenzial mit breitem Distanzbereich, in dem die Wechselwirkung als optimal bewertet und keine Differenzierung ermöglicht wird, nicht konsistent geliefert.
- Summation der *Score*-Anteile über aromatische Atome und ausreichendes *Sampling* führen zu geringer Variabilität in der Positionierung der kationischen Kohlenstoffatome.
- Diversitätsorientiertes und erschöpfendes *Docking* vergrößern den Posenraum, der im *Sampling* ausgewählt wird. Die zusätzlichen Posen werden aber schlechter bewertet als die meisten Posen aus unabhängigem *Docking*.
- Die Flexibilität ausgewählter Aminosäuren kann zur Bevorzugung oder Benachteiligung von Wechselwirkungen des Liganden mit diesen gegenüber starren Aminosäureseitenketten führen.
- Trotz Parametervariation in relevanten Intervallen sind weder signifikante Geometrie- noch *Scoring*-Unterschiede auszumachen.

Eine Herausforderung an das *Docking* von karbokationischen Intermediaten in Terpen-synthasebindetaschen ist die Homogenität der Liganden bezüglich der Atomtypisierung. Abgesehen von den kationisch modellierten Kohlenstoffatomen sind kaum spezifische Wechselwirkungen möglich, da alle anderen Kohlenstoffatome in PLANTS gleich behandelt werden und lediglich sterische Verhältnisse bzw. die Rotierbarkeit von Bindungen Einfluss haben können. Dabei spielt auch die flexible oder starre Handhabung der Aminosäuren während des *Docking* eine erhebliche Rolle. Über Seitenketten hinaus ist auch für Terpen-synthasen bekannt, dass bei Substratbindung wesentliche Konformationsänderungen im Enzym auftreten, welche zu einer starken Wechselwirkung mit dem Substrat führen.^{38,211} Die Benutzung des Homologiemodells als Rezeptor kann jedoch als gute Repräsentanz für die substratbindende Konformation der Bindetasche angesehen werden, weil das Homologie-modell auf einer experimentell aufgeklärten Proteinstruktur mit gebundenem Liganden beruht. PLANTS behandelt zyklische Strukturen als starr, so dass das α -Terpinylkation nur eine frei drehbare Bindung als Freiheitsgrad besitzt. Ein zusätzliches Konformations-*Sampling* vor dem *Docking* kann diesen Umstand beheben. Die in dieser Arbeit durchgeführten Analysen zielten jedoch auf die Unterschiedlichkeit von Geometrien und Interaktionen bei verschiedenen Potenzialvarianten, so dass die konkrete Konformation des Liganden eine untergeordnete Rolle spielte.

Es wurde gezeigt, dass es statistisch keine wesentlichen Unterschiede zwischen Geometrien aus unterschiedlichen Potenzialvarianten gibt. Dies beweist nicht, dass es kein geeignetes Potenzial für die Widerspiegelung der neu modellierten Wechselwirkungen gibt, weil weder eine vollständige Durchsuchung des Parameterraumes erfolgte, noch andere Potenzialformen probiert wurden. Konkrete Übergangszustandskonformationen – auch starr – zu docken kann trotzdem sinnvoll sein, wenn man an Geometrie-vorschlägen für die relative

Anordnung von Rezeptor und Ligand interessiert ist. Jedoch muss man sich in solchen Fällen bewusst sein, dass die Bewertung durch eine *Scoring*-Funktion keine Widerspiegelung der realen Bindungsenergetik (weder freie Bindungsenergie noch Rezeptor-Ligand-Wechselwirkungsenergie) leisten kann, weil die zugrunde liegenden Näherungen mit Fehlern in Größenordnungen behaftet sind, die eine Differenzierung verbieten.^{257,268,404,433} Die aktuellen *Scoring*-Funktionen werden im Allgemeinen als verbesserungswürdig angesehen.^{269,270,287,329} Unzureichendes *Sampling* durch mangelhafte Flexibilitätsmodellierung ist ein weiterer Nachteil von *Docking*-Programmen. Es stellt sich die Frage, ob es im Rahmen virtuellen *Screenings* notwendig ist, globale Optima für Rezeptor-Ligand-Wechselwirkungen zu finden. Für Untersuchungen zur Positionierung intermediärer Liganden – wie in dieser Arbeit – scheint eine solche Herangehensweise sinnvoll, wenn die Ligand-Eigenschaft gegeben ist. Ist aber, wie bei virtuellem *Screening*, nicht bekannt, ob ein Molekül an den Rezeptor bindet oder nicht, dann sollten auch entropische Terme und (De-)Solvatationsenergien in die Betrachtung einbezogen werden, welche schwierig zu modellieren sind.^{257,276,327,338} Bezüglich der Weiterentwicklung von molekularem *Docking* scheint eine dreiteilige Spezialisierung bezüglich unterschiedlicher Zwecke sinnvoll, wie sie von Huang *et al.* vorgestellt wurde (³²⁹, siehe Seite 27): Geometrievorhersage, Affinitätsvorhersage und virtuelles *Screening*. Zusätzlich zu diesem sehr nützlichen Ansatz ist eine realistischere Beschreibung des Begriffes „Bindemodus“ sinnvoll. Während auch mit dem Wort „Proteinmodell“ oft nur eine konkrete Molekülkonformation gemeint ist, wäre es hilfreich, eine Beschreibung zu entwickeln, welche der Flexibilität molekularer Strukturen gerecht wird. Dies hätte auch Implikationen für die *Sampling*-Algorithmen zukünftiger *Docking*-Programme, welche Ensembles von Rezeptorkonformationen durchsuchen müssten.

Die präsentierten Daten weisen für die Optimierung intermolekularer Interaktionen und die Berechnung von Wechselwirkungsenergien in Richtung von Methoden, welche über die einfache Struktur von Kraftfeldern hinausgehen und folgende Effekte besser bzw. überhaupt erst modellieren:

- intra- und intermolekulare Polarisierung
- Orbitalwechselwirkungen
- Umfassendes *Sampling* von Ligand- und Rezeptorkonformationen (inkl. Proteinerückgrat)

Diese Punkte sind nur mit wenigstens semiempirischen Berechnungsmethoden zu adressieren, welche für Systeme der Größe der meisten Enzymbindetaschen unpraktikabel lange Rechenzeiten benötigen. Die Herausforderungen, welchen sich die wissenschaftliche Gemeinschaft im Bereich des molekularen *Docking* widmen muss, um statt Wechselwirkungsenergien Bindeaffinitäten zu berechnen, sind noch weitaus komplexer, denn sie umfassen die ausreichend genaue Modellierung von Solvatations- und Desolvatationseffekten sowie entropischen Energiebeiträgen (siehe zum Beispiel ^{275,339,349,434} und insbesondere die Übersichtsartikel von Pons *et al.*,⁴³⁵ und Meng *et al.*,²⁶⁹).

Kraftfeldmethoden und *Docking* an Terpensynthasen zur Erklärung der Produktspezifität sind ein sehr geeignetes Testsystem für theoretische Methoden, weil diese experimentelle Resultate widerspiegeln können müssen, um als zweckmäßig gelten zu können. Auf der Basis von Homologiemodellierung, molekularem *Docking*, quantenmechanischen Berechnungen und detaillierten Vergleichen wurden für die KSL-Enzyme wesentliche Grund-

steine für biochemische Experimente zur Charakterisierung der Produktbildung in dieser Enzymgruppe gelegt. Allgemein hat das Zusammenspiel von Rechen- und Laborexperimenten für Terpensynthesen eine hervorgehobene Bedeutung, da sie als Paradebeispiel für fein modulierte Enzymkatalysen gelten können. Diese fordern sowohl experimentell als auch theoretisch anspruchsvolle Untersuchungen für ein umfassendes Verständnis der beteiligten Prozesse.

4.4 Schlussfolgerungen und Stellungnahme zu den in der Zielstellung aufgeworfenen Fragen

- Inwiefern kann molekulares *Docking* benutzt werden, um Intermediate einer enzymkatalysierten Reaktion sinnvoll in die Proteinbindetasche einzupassen?

Es mangelt an einer abgeschlossenen Beschreibung intermolekularer Wechselwirkungen, die realistisch genug ist, um die relevanten Wechselwirkungen differenziert zu modellieren. Auch wenn das *Sampling* durch unvollständige Flexibilitätsbetrachtung nicht perfekt ist, scheint die Entwicklung geeigneter *Scoring*-Funktionen für molekulares *Docking* die größere Herausforderung zu sein.

- Können aus *Docking*-Experimenten Vorschläge zu Übergangszuständen und deren Stabilisierung während der enzymatisch katalysierten Reaktion abgeleitet werden?

Ja, allerdings ist deren Bewertung nicht sehr vertrauenswürdig, weil die intermolekularen Wechselwirkungen mit kraftfeldbasierten Methoden nicht präzise und vollständig genug modelliert werden. Eine explizite Nachbetrachtung mit quantenchemischen Methoden wird empfohlen, ist aber in vielen Fällen durch die Größe des molekularen Systems herausfordernd.

- Ist das *Docking* von Intermediatstrukturen ein geeignetes Verfahren, um Inhibitoren zu finden?

Intermediatstrukturen, die Übergangszuständen ähneln, sollten starke Wechselwirkungen mit dem Rezeptor haben. *Docking* ist hier allerdings nur eingeschränkt nützlich, nämlich dann, wenn die verwendete *Scoring*-Funktion die relevanten Wechselwirkungen gut modelliert und dabei entropische Energiebeiträge sowie Solvatationsenergien ausreichend präzise modelliert werden oder keine Rolle spielen.

- Wie wird die Produktbildung in den ausgewählten KSL-Enzymen moduliert? Was sind die Ursachen für unterschiedliche Produkte in Enzymen, die sich in Primär- und Tertiärstruktur ähneln?

KSL-Enzyme modulieren die Produktbildung durch zusammenwirkende Faktoren:

- a) Formgebung des hydrophoben Teils der Bindetasche zur Substratselektion
- b) Flexibilität der Bindetasche zur Steuerung der Produktbildung (involviert auch periphere Aminosäuren und benachbarte Sekundärstrukturelemente)
- c) Differenzierte Stabilisierung von Intermediaten
- d) Modulation der Zugänglichkeit reaktiver Konformationen
- e) Bereitstellung bzw. Zugänglichkeit basischer Gruppen zur Protonabstraktion

Die hydrophobe Bindetasche von Terpensynthasen ist somit weit mehr als nur ein abgeschlossener Reaktionsraum.^{82,211,436} Die Vielzahl und Komplexität der einflussnehmenden Faktoren und auch deren Zusammenspiel verdeutlichen, dass KSL-Enzyme evolutionär hochspezialisiert sind. Noch zahlreiche und gründliche Untersuchungen werden notwendig sein, um die Produktbildung in Terpensynthasen vollständig zu verstehen.

5 ZUSAMMENFASSUNG

Terpenoide sind eine äußerst diverse Naturstoffklasse, deren Vertreter oft über komplexe Kohlenstoffgerüste verfügen. Viele dieser Substanzen sind unter anderem für Anwendungen in der Pharmazie oder der Geruchs- und Geschmacksstoffindustrie interessant. Terpensynthesen sind an ihrer Biosynthese beteiligt. Prenyltransferasen sind für die Übertragung von Prenylresten auf andere Moleküle verantwortlich. Beide Enzymklassen gemeinsam bilden die Grundlage einer breiten chemischen Diversität, die in zahlreichen metabolischen Prozessen Informationen vermittelt. Über ein tieferes Verständnis der involvierten Katalyseprozesse ist es möglich, diese Enzyme für biotechnologische Anwendungen gezielt auszuwählen, zu modifizieren und nutzbar zu machen. Die vorliegende Arbeit befasst sich mit der Modellierung von Prenyltransferasen und Terpensynthesen und dem molekularen *Docking* von Reaktionsintermediaten. Sie gliedert sich in drei Teile.

Zunächst wurden Homologiemodelle für die sieben DMAPP:AMP/ADP/ATP-Transferasen AtIPT1 und AtIPT3 bis AtIPT8 aus *Arabidopsis thaliana* erstellt. Anhand dieser Modelle wurde der aus der Literatur bekannte Reaktionsmechanismus nachvollzogen. Ein virtuelles *Screening* lieferte zahlreiche Vorschläge für potenzielle Liganden dieser Enzyme. Auf Grundlage dieser Ergebnisse wurden 17 potenzielle Inhibitoren zur weiteren Untersuchung ausgewählt. Diese wurden von Kooperationspartnern chemisch synthetisiert und anschließend am Enzym AtIPT1 auf ihre Aktivität getestet. Auf diese Weise wurden sechs Substanzen mit geringer inhibitorischer Potenz ($300 \mu\text{M} < \text{IC}_{50} < 1000 \mu\text{M}$) identifiziert.

Im zweiten Teil der Arbeit wurden für die vier Kaurensynthase-ähnlichen Enzyme

- *ent*-Kaurensynthase aus *Arabidopsis thaliana*
- *cis*-Abienolsynthase aus *Nicotiana tabacum*
- Santalen- und Bergamotensynthase aus *Solanum habrochaites*
- β -Phellandrensynthase aus *Solanum lycopersicum*

Homologiemodelle erzeugt. Mit molekularem *Docking* wurden Substratstrukturen in den Modellen positioniert. An den Enzym-Ligand-Komplexen werden Aspekte der jeweiligen Enzymkatalyse differenziert und vergleichend erörtert. Ergänzend zu den Modellierungen der Kaurensynthase-ähnlichen Enzyme wurden quantenchemische Berechnungen in Gasphase durchgeführt. Die Ergebnisse dieser Kalkulationen zeigen, dass intermediäre Kationen während der Enzymkatalyse stark attraktive Wechselwirkungen mit den freien Elektronenpaaren der Schwefelatome von Methioninseitenketten eingehen können. Gleiches gilt für die Interaktionen der kationischen Intermediate mit den π -Systemen aromatischer Aminosäuren.

Weitere quantenchemische Gasphase-Berechnungen zur Produktbildung in dem Multiproduktenzym Santalen- und Bergamotensynthase beweisen, dass dieser Prozess fein moduliert wird. Unterschiedliche Positionierungen von Reaktionsintermediaten in der Enzymbindetasche führen zur Bildung verschiedener Produkte. Auf der Basis der durchgeführten Modellierungen wurden für die Santalen- und Bergamotensynthase und die *cis*-Abienolsynthase Mutationsvorschläge unterbreitet, um die Funktion einzelner Aminosäuren der Substratbindetaschen aufzuklären und beide Enzyme schrittweise funktionell ineinander zu überführen. Zielgerichtete Mutagenesen und anschließende Aktivitätstests wurden durch Kooperationspartner durchgeführt. Sie liefern ein differenziertes Bild der Aktivitäten der Einfachmutanten der Santalen- und Bergamotensynthase und der *cis*-Abienolsynthase. Die Produktspektren beider Enzyme werden im Kontext der Proteinmodelle diskutiert.

Der dritte Teil der Arbeit hat die Modulation des Produktspektrums der Santalen- und Bergamotensynthese durch Wechselwirkungen von Liganden mit der Substratbindetasche zum Thema. Es wurde untersucht, ob molekulares *Docking* von kationischen Intermediaten der Enzymreaktionen Aufschluss über deren Positionierungen in den jeweiligen Substratbindetaschen geben kann. Zu diesem Zweck wurde das *Docking*-Programm PLANTS um die Modellierung spezifischer Wechselwirkungen zwischen den intermediären Kationen und bestimmten Rezeptoratomen erweitert. Diese Rezeptoratome umfassen Schwefelatome von Methioninseitenketten und aromatische Kohlenstoffatome. In umfangreichen *Docking*-Experimenten von Reaktionsintermediaten in die bekannte Struktur der *Mentha spicata* Limonensynthese und die modellierte Struktur der Santalen- und Bergamotensynthese wurden die neu modellierten Interaktionen untersucht. Unter Verwendung statistischer Methoden werden die Folgen der Verwendung unterschiedlicher Potenzialvarianten für die neu implementierten Wechselwirkungen dargelegt. Dabei werden sowohl geometrische Größen als auch Terme der Bewertungsfunktion von *Docking*-Posen analysiert. Die Analysen zeigen, dass die randomisierte Posengenerierung in PLANTS einen größeren Einfluss auf die Bewertung der *Docking*-Posen hat als die durchgeführte Variation der verwendeten Potenziale. Statistisch untermauert resultiert, dass es nicht sinnvoll ist, einzelne Kraftfeldparameter für molekulares *Docking* zu optimieren, ohne weitere wesentliche Aspekte intermolekularer Wechselwirkungen zu modellieren. Dazu gehören Rezeptorflexibilität, Entropie und Solvatation.

Die theoretischen Betrachtungen in dieser Arbeit schließen aus, dass die Mechanismen der Produktbildung in Terpensynthesen durch *Docking* oder quantenmechanische Berechnungen allein erklärt werden können. Vielmehr ist hierfür eine integrierte Betrachtung der Enzym-Ligand-Wechselwirkungen notwendig, die Molekülorbitale modelliert und den umfangreichen Konformationsraum realitätsnah abbildet.

6 LITERATURVERZEICHNIS

1. Gershenzon, J.; Dudareva, N. *Nat Chem Biol* **2007**, *3*, 408–414.
2. Lombard, J.; Moreira, D. *Mol Biol Evol* **2011**, *28*, 87–99.
3. Liang, P. H. et al. *Eur J Biochem* **2002**, *269*, 3339–3354.
4. Cory, E. J. et al. *J Am Chem Soc* **1966**, *88*, 4750–4751.
5. Ramos-Valdivia, A. C. et al. *Nat Prod Rep* **1997**, *14*, 591–603.
6. Degenhardt, J. et al. *Phytochemistry* **2009**, *70*, 1621–1637.
7. International Union of Biochemistry and Molecular Biology. Nomenclature Committee; Webb, E. C. *Enzyme Nomenclature 1992*; Academic Press: San Diego, California, 1992.
8. Wrolstad, R.; Jennings, W. *J Food Sci* **1965**, *30*, 274–279.
9. Croteau, R. B. *Planta Med* **1991**, *57*, S10–S14.
10. Sekiwa, Y. et al. *Nat Prod Lett* **2001**, *15*, 267–274.
11. Gaydou, E. M. et al. *J Agric Food Chem* **1989**, *37*, 1032–1037.
12. Hyatt, D. C. et al. *Proc Natl Acad Sci USA* **2007**, *104*, 5360–5365.
13. Jiang, L.; Kubota, K. *J Agric Food Chem* **2004**, *52*, 4197–4203.
14. Kapoor, I. P. et al. *J Agric Food Chem* **2009**, *57*, 5358–5364.
15. Weiss, E. R. et al. *J Agric Food Chem* **2003**, *51*, 2277–2282.
16. Behr, A.; Johnen, L. *ChemSusChem* **2009**, *2*, 1072–1095.
17. Chyau, C. C. et al. *J Agric Food Chem* **1996**, *44*, 1096–1099.
18. Zakharova, S. et al. *Arkivoc* **2004**, 79–96.
19. San Feliciano, A. et al. *Planta Med* **1993**, *59*, 485–490.
20. Teng, K. H.; Liang, P. H. *Bioorg Chem* **2011**.
21. Kellogg, B. A.; Poulter, C. D. *Curr Opin Chem Biol* **1997**, *1*, 570–578.
22. Ogura, K.; Koyama, T. *Chem Rev* **1998**, *98*, 1263–1276.
23. McGarvey, D. J.; Croteau, R. *Plant Cell* **1995**, *7*, 1015–1026.
24. Yang, T. et al. *Phytochemistry* **2005**, *66*, 285–293.
25. Iijima, Y. et al. *Plant Physiol* **2004**, *134*, 370–379.
26. Schnee, C. et al. *Plant Physiol* **2002**, *130*, 2049–2060.
27. Köksal, M. et al. *Nature* **2011**, *469*, 116–120.
28. Wendt, K. U.; Schulz, G. E. *Structure* **1998**, *6*, 127–133.
29. Oldfield, E.; Lin, F.-Y. *Angew Chem Int Ed* **2012**, *51*, 1124–1137.
30. Rynkiewicz, M. J. et al. *Proc Natl Acad Sci USA* **2001**, *98*, 13543–13548.
31. Whittington, D. A. et al. *Proc Natl Acad Sci USA* **2002**, *99*, 15375–15380.
32. Günnewich, N. Expression and characterization of terpene synthases from *Canabis sativa* L. and *Salvia sclarea* L., Martin-Luther-Universität Halle-Wittenberg: Halle (Saale), 2009.
33. Köksal, M. et al. *Biochemistry (Mosc.)* **2012**, *51*, 3011–3020.
34. Köllner, T. G. et al. *Plant Cell* **2004**, *16*, 1115–1131.
35. Burke, C. C. et al. *Proc Natl Acad Sci USA* **1999**, *96*, 13062–13067.
36. Tholl, D. *Plant Cell Online* **2004**, *16*, 977–992.
37. Schmidt, A. et al. *Plant Physiol* **2009**, *152*, 639–655.
38. Christianson, D. W. *Chem Rev* **2006**, *106*, 3412–3442.
39. Berman, H. M. et al. *Nucleic Acids Res* **2000**, *28*, 235–242.
40. Kampranis, S. C. et al. *Plant Cell* **2007**, *19*, 1994–2005.
41. Aaron, J. A. et al. *Biochemistry (Mosc.)* **2010**, *49*, 1787–1797.
42. Gennadios, H. A. et al. *Biochemistry (Mosc.)* **2009**, *48*, 6175–6183.
43. Shishova, E. Y. et al. *Biochemistry (Mosc.)* **2007**, *46*, 1941–1951.
44. Starks, C. M. et al. *Science* **1997**, *277*, 1815–1820.

45. Noel, J. P. et al. *ACS Chem Biol* **2010**, *5*, 377–392.
46. Zhou, K. et al. *J Biol Chem* **2012**, *287*, 6840–6850.
47. Köksal, M. et al. *Nat Chem Biol* **2011**, *7*, 431–433.
48. Wendt, K. U. et al. *Science* **1997**, *277*, 1811–1815.
49. Pandit, J. et al. *J Biol Chem* **2000**, *275*, 30610–30617.
50. Thoma, R. et al. *Nature* **2004**, *432*, 118–122.
51. Peters, R. J. et al. *Biochemistry (Mosc.)* **2000**, *39*, 15592–15602.
52. Cao, R. et al. *Proteins* **2010**, *78*, 2417–2432.
53. Oh, S. K. et al. *J Biol Chem* **2000**, *275*, 18482–18488.
54. Zhou, C.; Huang, R. H. *Proc Natl Acad Sci USA* **2008**, *105*, 16142–16147.
55. Chu, H. M. et al. *Nucleic Acids Res* **2010**, *38*, 1738–1748.
56. Strickland, C. L. et al. *Biochemistry (Mosc.)* **1998**, *37*, 16601–16611.
57. Tarshis, L. C. et al. *Biochemistry (Mosc.)* **1994**, *33*, 10871–10877.
58. Zhang, H. et al. *Structure Fold Des* **2000**, *8*, 241–251.
59. Chang, T. H. et al. *J Biol Chem* **2006**, *281*, 14991–15000.
60. Shao, H. et al. *Plant Cell* **2005**, *17*, 3141–3154.
61. Heide, L. *Curr Opin Chem Biol* **2009**, *13*, 171–179.
62. Metzger, U. et al. *J Mol Biol* **2010**, *404*, 611–626.
63. Bräuer, L. et al. *J Mol Mod* **2004**, *10*, 317–327.
64. Bräuer, L. et al. *Chembiochem* **2008**, *9*, 982–992.
65. Kuzuyama, T. et al. *Nature* **2005**, *435*, 983–987.
66. Sugawara, H. et al. *Proc Natl Acad Sci USA* **2008**, *105*, 2734–2739.
67. Maurer-Stroh, S. et al. *Genome Biol* **2003**, *4*.
68. Croteau, R. B.; Felton, M. *Arch Biochem Biophys* **1981**, *207*, 460–464.
69. Croteau, R. B. et al. *Arch Biochem Biophys* **1980**, *200*, 524–533.
70. Croteau, R. B. *Arch Biochem Biophys* **1986**, *251*, 777–782.
71. Croteau, R. B.; Venkatachalam, K. V. *Arch Biochem Biophys* **1986**, *249*, 306–315.
72. Croteau, R. B. et al. *J Biol Chem* **1986**, *261*, 7257–7263.
73. Croteau, R. B. et al. *J Biol Chem* **1986**, *261*, 13438–13445.
74. Wheeler, C. J.; Croteau, R. *Arch Biochem Biophys* **1986**, *246*, 733–742.
75. Back, K.; Chappell, J. *J Biol Chem* **1995**, *270*, 7375–7381.
76. Tholl, D. et al. *Plant J* **2005**, *42*, 757–771.
77. Martin, D. M. et al. *Plant Physiol* **2004**, *135*, 1908–1927.
78. Rising, K. A. et al. *J Am Chem Soc* **2000**, *122*, 1861–1866.
79. Sharon-Asa, L. et al. *Plant J* **2003**, *36*, 664–674.
80. Iijima, Y. et al. *Plant Physiol* **2004**, *136*, 3724–3736.
81. Köllner, T. G. et al. *Arch Biochem Biophys* **2006**, *448*, 83–92.
82. Miller, D. J.; Allemann, R. K. *Nat Prod Rep* **2012**, *29*, 60–71.
83. Vederas, J. C.; Leeper, F. J. *Biosynthesis: Aromatic Polyketides, Isoprenoids, Alkaloids*; Topics in Current Chemistry; Springer: Heidelberg, 2000.
84. Toyomasu, T.; Sassa, T. *Comprehensive Natural Products II*; Elsevier Science Ltd, 2010.
85. Peters, R. J.; Croteau, R. B. *Biochemistry (Mosc.)* **2002**, *41*, 1836–1842.
86. Peters, R. J.; Croteau, R. B. *Proc Natl Acad Sci USA* **2002**, *99*, 580–584.
87. Tokiwano, T. et al. *Org Biomol Chem* **2005**, *3*, 2713–2722.
88. Poralla, K. *Chem Biol* **2004**, *11*, 12–14.
89. Rajamani, R.; Gao, J. *J Am Chem Soc* **2003**, *125*, 12768–12781.
90. Koyama, T. *Biosci Biotechnol Biochem* **1999**, *63*, 1671–1676.
91. Brems, D. N.; Rilling, H. C. *J Am Chem Soc* **1977**, *99*, 8351–8352.
92. Takahashi, S.; Koyama, T. *Chem Rec New York N* **2006**, *6*, 194–205.

93. Lu, Y. P. et al. *Biochem Biophys Res Commun* **2009**, 379, 351–355.
94. Lu, Y. P. et al. *Biochem Biophys Res Commun* **2010**, 400, 758–762.
95. Greve, H.-H. In *Ullmann's Encyclopedia of Industrial Chemistry*; Wiley-VCH Verlag GmbH & Co. KGaA, Ed.; Wiley-VCH Verlag GmbH & Co. KGaA: Weinheim, Germany, 2000.
96. Frick, S. et al. *Proc Natl Acad Sci USA* **2013**, 110, 4194–4199.
97. Kale, T. A. et al. *Curr Top Med Chem* **2003**, 3, 1043–1074.
98. Chen, F. et al. *Plant J* **2011**, 66, 212–229.
99. Toyomasu, T. et al. *Biosci Biotechnol Biochem* **2000**, 64, 660–664.
100. Morrone, D. et al. *FEBS Lett* **2009**, 583, 475–480.
101. Sallaud, C. et al. *Plant Cell* **2009**, 21, 301–317.
102. Schillmiller, A. L. et al. *Proc Natl Acad Sci USA* **2009**, 106, 10865–10870.
103. Hillwig, M. L. et al. *Plant J* **2011**, 68, 1051–1060.
104. Dussourd, D. E.; Hoyle, A. M. *Chemoecology*, **2000**, 10, 11–16.
105. Martel, J. W.; Malcolm, S. B. *J Chem Ecol* **2004**, 30, 545–561.
106. Rasmann, S. et al. *Ecology* **2009**, 90, 2393–2404.
107. Kappers, I. F. et al. *Science* **2005**, 309, 2070–2072.
108. Erbilgin, N. et al. *Oecologia* **2006**, 148, 426–436.
109. Morrissey, J. P.; Osbourn, A. E. *Microbiol Mol Biol Rev* **1999**, 63, 708–724.
110. Papadopoulou, K. et al. *Proc Natl Acad Sci USA* **1999**, 96, 12923–12928.
111. Laurent, P. et al. *Eur J Org Chem* **2003**, 2003, 2733–2743.
112. Quintana, A. et al. *J Chem Ecol* **2003**, 29, 639–652.
113. Lindstedt, C. et al. *Oecologia* **2006**, 150, 519–526.
114. Nishida, R. *Annu Rev Entomol* **2002**, 47, 57–92.
115. Burse, A. et al. *Insect Biochem Mol Biol* **2007**, 37, 255–265.
116. Paul, V. J. et al. *Nat ProdRep* **2006**, 23, 153–180.
117. Erickson, A. A. et al. *J Chem Ecol* **2006**, 32, 1883–1895.
118. Paul, V. J.; Van Alstyne, K. L. *J Exp Mar Biol Ecol* **1992**, 160, 191–203.
119. Jung, V.; Pohnert, G. *Tetrahedron* **2001**, 57, 7169–7172.
120. Bhadury, P.; Wright, P. C. *Planta* **2004**, 219, 561–578.
121. Culioli, G. et al. *J Nat Prod* **2008**, 71, 1121–1126.
122. Cometto-Muniz, J. et al. *Pharmacol Biochem Behav* **1998**, 60, 765–770.
123. Rapp, A. *Food Nahr.* **1998**, 42, 351–363.
124. Martin, D. M. et al. *BMC Plant Biol* **2010**, 10, 226.
125. Rosenfeld, H. J. et al. *J Sci Food Agric* **2002**, 82, 1384–1390.
126. Horvat, R. et al. *J Food Sci* **1991**, 56, 714–715.
127. Tang, W. Z. et al. *J Nat Prod* **2009**, 72, 1017–1021.
128. Sy, L. K.; Brown, G. D. *Phytochemistry* **1998**, 49, 1715–1717.
129. Huang, J. M. et al. *Chem Pharm Bull Tokyo* **2000**, 48, 657–659.
130. Fukuyama, Y. et al. *Planta Med* **1993**, 59, 181–182.
131. Lai-King, S.; Brown, G. D. *Phytochemistry* **1998**, 47, 301–302.
132. Pino, J. et al. *Food Nahr* **1990**, 34, 555–560.
133. Kjonaas, R.; Croteau, R. *Arch Biochem Biophys* **1983**, 220, 79–89.
134. Hunter, G.; Brogden, W. *J Food Sci* **1965**, 30, 383–387.
135. De Carvalho, C. C.; da Fonseca, M. *Food Chem* **2006**, 95, 413–422.
136. Sekiwa-Iijima, Y. et al. *J Agric Food Chem* **2001**, 49, 5902–5906.
137. Wu, P. et al. *J Agric Food Chem* **1990**, 38, 1553–1555.
138. Bakkali, F. et al. *Food Chem Toxicol* **2008**, 46, 446–475.
139. Maruzella, J. C.; Henry, P. A. *J Am Pharm Assoc Balt* **1958**, 47, 294–296.
140. Das, P. K. et al. *Arch Int Pharmacodyn Ther* **1962**, 135, 167–177.
141. Abraham, G. J.; Agshikar, N. V. *Pharmacology* **1972**, 7, 109–114.

142. Low, D. et al. *Planta Med* **1974**, 26, 184–185.
143. Banerjee, A.; Nigam, S. S. *Indian J Med Res* **1976**, 64, 1318–1321.
144. Sharma, G. P. et al. *Planta Med* **1979**, 36, 185–186.
145. Garg, S. C.; Dengre, S. L. *Hindustan Antibiot Bull* **1986**, 28, 27–29.
146. Jalsenjak, V. et al. *Pharmazie* **1987**, 42, 419–420.
147. Han, J. L. et al. *Sheng Wu Gong Cheng Xue Bao* **2007**, 23, 561–569.
148. Roberts, S. C. *Nat Chem Biol* **2007**, 3, 387–395.
149. Chang, M. C. et al. *Nat Chem Biol* **2007**, 3, 274–277.
150. Aharoni, A. et al. *Trends Plant Sci* **2005**, 10, 594–602.
151. Martin, V. J. et al. *Nat Biotechnol* **2003**, 21, 796–802.
152. Townsend, B. J.; Llewellyn, D. J. *Trends Biotechnol* **2007**, 25, 239–241.
153. Misawa, N. *Curr Opin Biotechnol* **2011**, 22, 627–633.
154. Cox, A. D.; Der, C. J. *Curr Opin Cell Biol* **1992**, 4, 1008–1016.
155. Gibbs, R. et al. *Curr Med Chem*. **2001**, 8, 1437–1465.
156. Gibbs, B. S. et al. *J Med Chem* **1999**, 42, 3800–3808.
157. Bell, I. M. *Expert Opin Ther Patents* **2000**, 10, 1813–1831.
158. Cohen, L. H. et al. *Biochem Pharmacol* **2000**, 60, 1061–1068.
159. Bliznakov, E. G. *FEBS Lett* **2002**, 525, 169–170.
160. Benetka, W. et al. *Monatshfte Chem*. **2006**, 137, 1241–1281.
161. Proteau, P. *J Biol Chem* **1995**, 270, 21793–21799.
162. Cragg, G. M. *Med Res Rev* **1998**, 18, 315–331.
163. Ruiz-Sanchez, J. et al. *J Appl Microbiol* **2010**, 109, 2144–2150.
164. Zhao, K. et al. *Wei Sheng Wu Xue Bao* **2008**, 48, 403–407.
165. Pang, X. et al. *Se Pu* **2004**, 22, 185.
166. Yeung, T. K. et al. *Biochem Biophys Res Commun* **1999**, 263, 398–404.
167. Olah, E. et al. *Anticancer Res* **1996**, 16, 2469–2477.
168. Horwitz, S. B. *Ann Oncol* **1994**, 5 Suppl 6, S3–S6.
169. Horwitz, S. B. et al. *Ann N Acad Sci* **1986**, 466, 733–744.
170. Manfredi, J. J.; Horwitz, S. B. *Pharmacol Ther* **1984**, 25, 83–125.
171. Kumar, N. *J Biol Chem* **1981**, 256, 10435–10441.
172. Russell, R. G. G. *Pediatrics* **2007**, 119, S150–S162.
173. Chappell, J. *Plant Physiol* **1995**, 107, 1–6.
174. Parmryd, I.; Dallner, G. *Biochem Soc Trans* **1996**, 24, 677–682.
175. Wanke, M. et al. *Acta Biochim Pol* **2001**, 48, 663–672.
176. Sapir-Mir, M. et al. *Plant Physiol* **2008**, 148, 1219–1228.
177. Hemmerlin, A. et al. *Prog Lipid Res* **2011**.
178. Laule, O. et al. *Proc Natl Acad Sci USA* **2003**, 100, 6866–6871.
179. Bick, J. A.; Lange, B. M. *Arch Biochem Biophys* **2003**, 415, 146–154.
180. Schuhr, C. A. et al. *Phytochem Rev* **2003**, 2, 3–16.
181. Fujihashi, M. et al. *Proc Natl Acad Sci USA* **2001**, 98, 4337–4342.
182. Kharel, Y. et al. *FEBS J* **2006**, 273, 647–657.
183. Archer, B. L.; Cockbain, E. G. In *Methods in Enzymology*; Elsevier, 1969; Vol. 15, pp. 476–480.
184. McMullen, A. I.; McSweeney, G. P. *Biochem J* **1966**, 101, 42–47.
185. Schmidt, T. et al. *BMC Biochem* **2010**, 11.
186. Hillebrand, A. et al. *PLoS ONE* **2012**, 7, e41874.
187. Miller, D. J. et al. *Org Biomol Chem* **2007**, 5, 3287–3298.
188. Lin, F. Y. et al. *Proc Natl Acad Sci USA* **2010**, 107, 21337–21342.
189. Leeper, F. J. et al. *Biosynthesis : aromatic polyketides, isoprenoids, alkaloids*; Springer: Berlin; London, 2011.
190. Wheeler, C. J.; Croteau, R. *J Biol Chem* **1987**, 262, 8213–8219.

191. Wheeler, C. J.; Croteau, R. B. *Proc Natl Acad Sci USA* **1987**, *84*, 4856–4859.
192. Roy, A. et al. *J Am Chem Soc* **2007**, *129*, 12453–12460.
193. Van der, P. G. et al. *J Bone Miner Res* **1992**, *7*, 981–986.
194. Lyles, K. W. et al. *N Engl J Med* **2007**, *357*, 1799–1809.
195. Gerstenfeld, L. C. et al. *J Bone Miner Res* **2009**, *24*, 196–208.
196. Sonnemann, J. et al. *Anticancer Drugs* **2003**, *14*, 767–771.
197. Sonnemann, J. et al. *Anticancer Drugs* **2001**, *12*, 459–465.
198. Kubo, T. et al. *Cancer Chemother Pharmacol* **2008**, *62*, 111–116.
199. Croteau, R. B.; Shaskus, J. *Arch Biochem Biophys* **1985**, *236*, 535–543.
200. Otomo, K. et al. *Biosci Biotechnol Biochem* **2004**, *68*, 2001–2006.
201. Gutta, P.; Tantillo, D. J. *J Am Chem Soc* **2006**, *128*, 6172–6179.
202. Gutta, P.; Tantillo, D. J. *Org Lett* **2007**, *9*, 1069–1071.
203. Hong, Y. J.; Tantillo, D. J. *Nat Chem* **2009**, *1*, 384–389.
204. Hong, Y. J.; Tantillo, D. J. *J Am Chem Soc* **2009**, *131*, 7999–8015.
205. Tantillo, D. J. *Chem Soc Rev* **2010**, *39*, 2847–2854.
206. Hong, Y. J.; Tantillo, D. J. *J Am Chem Soc* **2010**, *132*, 5375–5386.
207. Hong, Y. J.; Tantillo, D. J. *Chem Commun Camb* **2011**.
208. Hong, Y. J.; Tantillo, D. J. *J Am Chem Soc* **2011**, *133*, 18249–18256.
209. Tantillo, D. J. *Nat Prod Rep* **2011**, *28*, 1035–1053.
210. Schulze, D. Die Diphosphataktivierung in Proteinen : Betrachtungen aus verschiedenen Blickwinkeln, Martin-Luther-Universität Halle-Wittenberg: Halle (Saale), 2011.
211. Christianson, D. W. *Curr Opin Chem Biol* **2008**, *12*, 141–150.
212. Ohara, K. et al. *Biochem J* **2009**, *421*, 231–241.
213. Bräuer, L. Modellierung- und Mutationsstudien an ausgewählten prenylierenden Enzymen, Martin-Luther-Universität Halle-Wittenberg: Halle (Saale), 2006.
214. Takei, K. et al. *J Biol Chem* **2001**, *276*, 26405–26410.
215. Kakimoto, T. *Plant Cell Physiol* **2001**, *42*, 677–685.
216. Kamada-Nobusada, T.; Sakakibara, H. *Phytochemistry* **2009**, *70*, 444–449.
217. Mok, D. W. S.; Mok, M. C. *Cytokinins: chemistry, activity, and function*; CRC Press: Boca Raton, FL, USA, 1994.
218. Mok, D. W.; Mok, M. C. *Annu Rev Plant Physiol Plant Mol Biol* **2001**, *52*, 89–118.
219. Sakakibara, H. In *Plant Hormones: Biosynthesis, Signal Transduction, Action!*; Springer: Dordrecht, 2010; pp. 95–114.
220. Sakakibara, H. *Ann Rev Plant Biol* **2006**, *57*, 431–449.
221. Miyawaki, K. et al. *Proc Natl Acad Sci USA* **2006**, *103*, 16598–16603.
222. Golovko, A. et al. *Plant Mol Biol* **2002**, *49*, 161–169.
223. Sakakibara, H. et al. *Proc Natl Acad Sci USA* **2005**, *102*, 9972–9977.
224. Blackwell, J. R.; Horgan, R. *Phytochemistry* **1993**, *34*, 1477–1481.
225. Morris, R. et al. *Aust J Plant Physiol* **1993**, *20*, 621.
226. Takei, K. et al. *Plant Cell Physiol* **2004**, *45*, 1053–1062.
227. Miyawaki, K. et al. *Plant J* **2004**, *37*, 128–138.
228. Galichet, A. et al. *Plant Physiol* **2008**, *146*, 1155–1164.
229. Sakamoto, T. et al. *Plant Physiol* **2004**, *134*, 1642–1653.
230. Morrone, D. et al. *Arch Biochem Biophys* **2006**, *448*, 133–140.
231. Hayashi, K. et al. *FEBS Lett* **2006**, *580*, 6175–6181.
232. Xu, M. et al. *Phytochemistry* **2007**, *68*, 312–326.
233. Hedden, P.; Kamiya, Y. *Annu Rev Plant Physiol Plant Mol Biol* **1997**, *48*, 431–460.
234. Depuydt, S.; Hardtke, C. *Curr Biol* **2011**, *21*, R365–R373.

235. Srikanth, A.; Schmid, M. *Cell Mol Life Sci* **2011**, *68*, 2013–2037.
236. Kurosawa, E. *Nat Hist Soc Formosa* **1926**, *16*, 213–227.
237. Arana, M. V. et al. *Proc Natl Acad Sci USA* **2011**, *108*, 9292–9297.
238. Marin-de la Rosa, N. et al. *Plant Signal Behav* **2011**, *6*, 1411–1413.
239. Hueso-Falcon, I. et al. *Eur J Med Chem* **2011**, *46*, 1291–1305.
240. Smanski, M. J. et al. *Proc Natl Acad Sci USA* **2011**, *108*, 13498–13503.
241. Yamaguchi, S. et al. *Plant J* **1996**, *10*, 203–213.
242. Yamaguchi, S. et al. *Plant Physiol* **1998**, *116*, 1271–1278.
243. Saito, T. et al. *Plant Physiol* **1995**, *109*, 1239–1245.
244. Gasteiger, E. et al. *Nucleic Acids Res* **2003**, *31*, 3784–3788.
245. The UniProt Consortium *Nucleic Acids Res* **2012**, *40*, D71–D75.
246. Kawaide, H. et al. *J Biol Chem* **1997**, *272*, 21706–21712.
247. Kawaide, H. et al. *J Biol Chem* **2000**, *275*, 2276–2280.
248. Xu, M. et al. *Proc Natl Acad Sci USA* **2007**, *104*, 7397–7401.
249. Wilderman, P. R.; Peters, R. J. *J Am Chem Soc* **2007**, *129*, 15736–15737.
250. Levin, D. A. *Q Rev Biol* **1973**, *48*, 3–15.
251. Besser, K. et al. *Plant Physiol* **2009**, *149*, 499–514.
252. Kang, J. H. et al. *Plant Physiol* **2010**, *154*, 262–272.
253. Van Der Hoeven, R. S. et al. *Plant Cell* **2000**, *12*, 2283–2294.
254. Jones, C. G. et al. *J Biol Chem* **2011**, *286*, 17445–17454.
255. Sallaud, C. et al. *Plant J* **2012**, *72*, 1–17.
256. Kenny, P. W. *J Comput Aid Mol* **2012**, *26*, 69–72.
257. Marshall, G. R. *J Comput Aid Mol* **2012**, *26*, 3–8.
258. Blaney, J. *J Comput Aid Mol* **2012**, *26*, 13–14.
259. Brown, F. K. *J Comput Aid Mol* **2012**, *26*, 27–28.
260. Clark, R. D.; Waldman, M. *J Comput Aid Mol* **2012**, *26*, 29–34.
261. Jain, A. N.; Cleves, A. E. *J Comput Aid Mol* **2012**, *26*, 57–67.
262. Kuntz, I. D. *J Comput Aid Mol* **2012**, *26*, 73–75.
263. Mobley, D. L. *J Comput Aid Mol* **2012**, *26*, 93–95.
264. Murcko, M. A.; Patrick, W. W. *J Comput Aid Mol* **2012**, *26*, 97–102.
265. Segall, M. *J Comput Aid Mol* **2012**, *26*, 121–124.
266. Woltosz, W. S. *J Comput Aid Mol* **2012**, *26*, 159–163.
267. Schneider, G. *J Comput Aid Mol* **2012**, *26*, 115–120.
268. Stouch, T. R. *J Comput Aid Mol* **2012**, *26*, 125–134.
269. Meng, X. Y. et al. *Curr Comput Aided Drug* **2011**, *7*, 146–157.
270. Huang, S. Y.; Zou, X. *Int J Mol Sci* **2010**, *11*, 3016–3034.
271. Yuriev, E. et al. *J Mol Recognit* **2011**, *24*, 149–164.
272. Whalen, K. L. et al. *Mol Inf* **2011**, *30*, 459–471.
273. Kim, R.; Skolnick, J. *J Comp Chem* **2008**, *29*, 1316–1331.
274. Chang, M. W. et al. *J Comp Chem* **2008**, *29*, 1753–1761.
275. Lee, J.; Seok, C. *Proteins* **2008**, *70*, 1074–1083.
276. Ruvinsky, A. M. *J Comp Chem* **2007**, *28*, 1364–1372.
277. Ruvinsky, A. M.; Kozintsev, A. V. *J Comp Chem* **2005**, *26*, 1089–1095.
278. Verdonk, M. L. et al. *J Med Chem* **2005**, *48*, 6504–6515.
279. Korb, O. et al. *J Chem Inf Model* **2009**, *49*, 84–96.
280. Knegt, R. M. et al. *J Mol Biol* **1997**, *266*, 424–440.
281. Zsoldos, Z. et al. *J Mol Graph Model* **2007**, *26*, 198–212.
282. Grosdidier, A. et al. *J Comp Chem* **2011**.
283. Zhang, C.; Lai, L. *J Comp Chem* **2011**, *32*, 2598–2612.
284. Setny, P.; Zacharias, M. *Nucleic Acids Res* **2011**, *39*, 9118–9129.
285. Kitchen, D. B. et al. *Nat Rev Drug Discov* **2004**, *3*, 935–949.

286. *Molecular Operating Environment (MOE)*; Chemical Computing Group Inc., 2011.
287. Sadjad, B. S.; Zsoldos, Z. *IEEEACM Trans Comput Biol Bioinform* **2011**, *8*, 1120–1133.
288. Gehlhaar, D. K. et al. *Chem Biol* **1995**, *2*, 317–324.
289. Jones, G. et al. *J Mol Biol* **1997**, *267*, 727–748.
290. Gorelik, B.; Goldblum, A. *Proteins* **2008**, *71*, 1373–1386.
291. Liu, M.; Wang, S. *J Comput Aid Mol* **1999**, *13*, 435–451.
292. Meier, R. et al. *J Chem Inf Model* **2010**, *50*, 879–889.
293. Axenopoulos, A. et al. *IEEE ACM Trans Comput Biol Bioinform* **2011**, *8*, 1441–1457.
294. Choi, J. et al. *J Mol Graph Model* **2012**, *32*, 82–88.
295. Chung, J. Y. et al. *Arch Pharm Res* **2011**, *34*, 1451–1458.
296. DesJarlais, R. L. et al. *J Med Chem* **1986**, *29*, 2149–2153.
297. DesJarlais, R. L.; Dixon, J. S. *J Comput Aid Mol* **1994**, *8*, 231–242.
298. Hawkins, P. C. et al. *J Med Chem* **2007**, *50*, 74–82.
299. Kim, D. S. et al. *J Biomol Struct Dyn* **2011**, *29*, 219–242.
300. Shoichet, B. K. et al. *J Comp Chem* **1992**, *13*, 380–397.
301. Venkatachalam, C. M. et al. *J Mol Graph Model* **2003**, *21*, 289–307.
302. Billeter, M. et al. *Biopolymers* **1987**, *26*, 777–793.
303. Wu, G. et al. *J Comp Chem* **2003**, *24*, 1549–1562.
304. Pierce, B. et al. *Bioinformatics* **2005**, *21*, 1472–1478.
305. Meng, E. C. et al. *J Comp Chem* **1992**, *13*, 505–524.
306. Krieger, E. et al. *Proteins-Struct Funct Bioinforma* **2004**, *57*, 678–683.
307. Mooij, W. T.; Verdonk, M. L. *Proteins* **2005**, *61*, 272–287.
308. Rykunov, D.; Fiser, A. *BMC Bioinformatics* **2010**, *11*, 128.
309. Muegge, I. et al. *J Med Chem* **1999**, *42*, 2498–2503.
310. Ruvinsky, A. M.; Kozintsev, A. V. *Biophys Chem* **2005**, *115*, 255–261.
311. Tondel, K. et al. *J Comput Aid Mol* **2006**, *20*, 131–144.
312. Dobes, P. et al. *J Comput Aid Mol* **2011**, *25*, 223–235.
313. Fanfrlik, J. et al. *J Phys Chem B* **2010**, *114*, 12666–12678.
314. Cho, A. E. et al. *J Chem Phys* **2009**, *131*, 134108.
315. Raub, S. et al. *J Chem Inf Model* **2008**, *48*, 1492–1510.
316. Fischer, B. et al. *Proteins* **2008**, *70*, 1264–1273.
317. Ferrara, P. et al. *J Chem Inf Model* **2006**, *46*, 254–263.
318. Raha, K.; Merz, K. M. *J Am Chem Soc* **2004**, *126*, 1020–1021.
319. Raha, K.; Merz, K. M. *J Med Chem* **2005**, *48*, 4558–4575.
320. Ackermann, F. et al. *Proc Int Conf Intell Syst Mol Biol* **1995**, *3*, 3–11.
321. Hoffmann, D. et al. *J Med Chem* **1999**, *42*, 4422–4433.
322. Khandelwal, A. et al. *J Med Chem* **2005**, *48*, 5437–5447.
323. Luo, W. et al. *J Mol Mod* **2010**, *16*, 903–913.
324. Shoichet, B. K.; Kuntz, I. D. *Protein Eng* **1993**, *6*, 723–732.
325. Tantillo, D. J.; Houk, K. N. *J Comp Chem* **2002**, *23*, 84–95.
326. Verdonk, M. L. et al. *Proteins* **2003**, *52*, 609–623.
327. Zoete, V. et al. *J Mol Recognit* **2010**, *23*, 457–461.
328. Wang, J. C. et al. *J Chem Inf Model* **2011**, *51*, 2528–2537.
329. Huang, S. Y. et al. *Phys Chem Chem Phys* **2010**, *12*, 12899–12908.
330. Taylor, R. D. et al. *J Comput Aid Mol* **2002**, *16*, 151–166.
331. Levinthal, C. et al. *Proc Natl Acad Sci USA* **1975**, *72*, 1330–1334.
332. Kuntz, I. D. et al. *J Mol Biol* **1982**, *161*, 269–288.
333. Havel, T. F. et al. *J Theor Biol* **1983**, *104*, 359–381.

334. Cramer, R. D. et al. *J Am Chem Soc* **1988**, *110*, 5959–5967.
335. Cramer, R. D. et al. *Prog Clin Biol Res* **1989**, *291*, 161–165.
336. Jackson, R. M. et al. *J Mol Biol* **1998**, *276*, 265–285.
337. Majeux, N. et al. *Proteins* **1999**, *37*, 88–105.
338. Shoichet, B. K. et al. *Proteins* **1999**, *34*, 4–16.
339. Pei, J. et al. *Proteins* **2004**, *57*, 651–664.
340. Muegge, I.; Martin, Y. C. *J Med Chem* **1999**, *42*, 791–804.
341. Anderson, A. C. et al. *Chem Biol* **2001**, *8*, 445–457.
342. Shoichet, B. K. et al. *Curr Opin Chem Biol* **2002**, *6*, 439–446.
343. Hetenyi, C.; van der Spoel, D. *Protein Sci* **2011**, *20*, 880–893.
344. Iorga, B. et al. *J Mol Mod* **2006**, *12*, 366–372.
345. Smith, G. R.; Sternberg, M. J. *Proteins* **2003**, *52*, 74–79.
346. Komatsu, K. et al. *Proteins* **2003**, *52*, 15–18.
347. Yu, Y. B. et al. *Biophys J* **2001**, *81*, 1632–1642.
348. Biot, C. et al. *J Am Chem Soc* **2003**, *125*, 13988–13994.
349. Ruvinsky, A. M. *J Comput Aid Mol* **2007**, *21*, 361–370.
350. Siebert, X.; Amzel, L. M. *Proteins* **2004**, *54*, 104–115.
351. Floriano, W. B. et al. *J Med Chem* **2004**, *47*, 56–71.
352. Kinnings, S. L. et al. *J Chem Inf Model* **2011**, *51*, 408–419.
353. Kinnings, S. L. et al. *J Chem Inf Model* **2011**.
354. O'Boyle, N. M. et al. *J Chem Inf Model* **2009**, *49*, 1871–1878.
355. Cole, J. C. et al. *Proteins* **2005**, *60*, 325–332.
356. Cummings, M. D. et al. *J Med Chem* **2005**, *48*, 962–976.
357. Joy, S. et al. *Silico Biol* **2006**, *6*, 601–605.
358. Onodera, K. et al. *J Chem Inf Model* **2007**, *47*, 1609–1618.
359. Wolf, A. et al. *J Chem Inf Model* **2007**, *47*, 1036–1044.
360. Cross, J. B. et al. *J Chem Inf Model* **2009**, *49*, 1455–1474.
361. Lee, H. S.; Zhang, Y. *Proteins* **2012**, *80*, 93–110.
362. Sousa, S. F. et al. *Proteins* **2006**, *65*, 15–26.
363. Naik, S. et al. *Fluid Phase Equilibria* **1989**, *49*, 115–126.
364. Rude, M. A.; Schirmer, A. *Curr Opin Microbiol* **2009**, *12*, 274–281.
365. Schmidt, T. et al. *Plant Mol Biol Report* **2010**, *28*, 277–284.
366. Falara, V. et al. *Plant Physiol* **2011**, *157*, 770–789.
367. Brandt, W. et al. *Phytochemistry* **2009**, *70*, 1758–1775.
368. Krieger, E. et al. *Proteins-Struct Funct Genet* **2002**, *47*, 393–402.
369. Altschul, S. F. et al. *Nucleic Acids Res* **1997**, *25*, 3389–3402.
370. Hooft, R. W. et al. *Nature* **1996**, *381*, 272.
371. Hooft, R. W. et al. *Comput Appl Biosci* **1996**, *12*, 525–529.
372. King, R. D.; Sternberg, M. J. *Protein Sci* **1996**, *5*, 2298–2310.
373. Canutescu, A. A. et al. *Protein Sci* **2003**, *12*, 2001–2014.
374. Cameron, M. et al. *IEEE ACM Trans Comput Biol Bioinform* **2004**, *1*, 116–129.
375. Altschul, S. F. et al. *FEBS J* **2005**, *272*, 5101–5109.
376. Larkin, M. A. et al. *Bioinformatics* **2007**, *23*, 2947–2948.
377. Goujon, M. et al. *Nucleic Acids Res* **2010**, *38*, W695–W699.
378. Letunic, I.; Bork, P. *Bioinformatics* **2007**, *23*, 127–128.
379. Letunic, I.; Bork, P. *Nucleic Acids Res* **2011**, *39*, W475–W478.
380. Guimaraes, C. R. et al. *J Chem Inf Model* **2011**, *51*, 1199–1204.
381. Wiederstein, M.; Sippl, M. J. *Nucleic Acids Res* **2007**, *35*, W407–W410.
382. Sippl, M. J. *Proteins* **1993**, *17*, 355–362.
383. Sun, J. Q. et al. *Plant Physiol* **2003**, *131*, 167–176.
384. Cho, G. S.; Szostak, J. W. *Chem Biol* **2006**, *13*, 139–147.

385. R Development Core Team *R: A Language and Environment for Statistical Computing*; R Development Core Team: Vienna, Austria, 2011.
386. Korb, O. et al. *Lect Notes Comput Sci* **2006**, 4150, 247–258.
387. Korb, O. et al. *Swarm Intel* **2007**, 1, 115–134.
388. Stützle, T.; Hoos, H. *Future Gener Comput Syst* **2000**, 16, 889–914.
389. Nelder, J.; Mead, R. *Comput J* **1965**, 7, 308–313.
390. Verkhivker, G. M. et al. *Proteins* **2002**, 48, 539–557.
391. Verkhivker, G. M. et al. *Proteins* **2003**, 53, 201–219.
392. Verkhivker, G. M. *J Mol Graph Model* **2004**, 22, 335–348.
393. Eldridge, M. D. et al. *J Comput Aid Mol* **1997**, 11, 425–445.
394. Murray, C. W. et al. *J Comput Aid Mol* **1998**, 12, 503–519.
395. Clark, M. et al. *J Comp Chem* **1989**, 10, 982–1012.
396. Nissink, J. W. M. et al. *Proteins* **2002**, 49, 457–471.
397. Jones, C. G. et al. *Phytochemistry* **2006**, 67, 2463–2468.
398. *Maestro*; Schrodinger LLC: New York, NY, 2011.
399. Kraft, L. et al. *J Mol Biol* **2002**, 318, 1057–1069.
400. Dobritzsch, D. et al. *Arthritis Rheum* **2011**, 63, 3740–3748.
401. Welin, M. et al. *J Mol Biol* **2007**, 366, 1615–1623.
402. Huang, K.-F. et al. *Biochem J* **2008**, 411, 181.
403. *The PyMOL Molecular Graphics System*; Schrödinger, LLC.
404. Beauchamp, K. A. et al. *J Chem Theory Comput* **2012**, 8, 1409–1414.
405. Lindorff-Larsen, K. et al. *PLoS ONE* **2012**, 7, e32131.
406. Krieger, E. et al. *J Mol Graph Model* **2006**, 25, 481–486.
407. Caruthers, J. M. *J Biol Chem* **2000**, 275, 25533–25539.
408. Rynkiewicz, M. J. et al. *Biochemistry (Mosc.)* **2002**, 41, 1732–1741.
409. Garms, S. et al. *Phytochemistry* **2012**, 75, 6–13.
410. Jindal, G.; Sunoj, R. B. *Org Biomol Chem* **2012**.
411. Keeling, C. I. et al. *Proc Natl Acad Sci USA* **2008**, 105, 1085–1090.
412. Greenhagen, B. T. et al. *Proc Natl Acad Sci USA* **2006**, 103, 9826–9831.
413. Cane, D. E. *Acc Chem Res* **1985**, 18, 220–226.
414. Croteau, R. *Chem Rev* **1987**, 87, 929–954.
415. Consoli, F. L. et al. *J Chem Ecol* **2002**, 28, 1675–1689.
416. Landmann, C. et al. *Arch Biochem Biophys* **2007**, 465, 417–429.
417. Krieger, E. et al. In *Structural Bioinformatics*; Methods of biochemical analysis; Wiley-Liss Inc.: Hoboken, NJ, 2003; pp. 507–521.
418. Carman, R.; Dennis, N. *Aust J Chem* **1968**, 21, 823.
419. Gasteiger, J.; Saller, H. *Angew Chem* **1985**, 97, 699–701.
420. Meyer, E. A. et al. *Angew Chem Int Ed* **2003**, 42, 1210–1250.
421. Duan G. et al. *Mol Phys* **2001**, 99, 1689–1699.
422. Quinonero, D. et al. *J Phys Chem* **2005**, 109, 4632–4637.
423. Quinonero, D. et al. *Chem Phys Chem* **2006**, 7, 2487–2491.
424. Dougherty, D. A. *J Nutr* **2007**, 137, 1504S–1508S.
425. Plewczynski, D. et al. *J Comp Chem* **2011**, 32, 742–755.
426. Erickson, J. A. et al. *J Med Chem* **2004**, 47, 45–55.
427. Taha, M. O. et al. *J Chem Inf Model* **2011**, 51, 647–669.
428. Verkhivker, G. M. et al. *J Comput Aid Mol* **2000**, 14, 731–751.
429. Hawkins, P. C. D. et al. *J Comput Aid Mol* **2008**, 22, 179–190.
430. Jain, A. N.; Nicholls, A. *J Comput Aid Mol* **2008**, 22, 133–139.
431. Korb, O. et al. *J Comput Aid Mol* **2012**, 26, 185–197.
432. Cheng, T. et al. *J Chem Inf Model* **2009**, 49, 1079–1093.
433. Paton, R. S.; Goodman, J. M. *J Chem Inf Model* **2009**, 49, 944–955.

434. Bryce, R. A. *Future Med Chem* **2011**, 3, 683–698.
435. Pons, C. et al. *Proteins* **2010**, 78, 95–108.
436. Allemann, R. K. *Pure Appl Chem* **2008**, 80, 1791–1798.

ANHANG

A 1 Aminosäuresequenzen der AtIPT

Im Folgenden sind die Aminosäuresequenzen der neun annotierten Isopentenyltransferasen aus *Arabidopsis thaliana* im fasta-Format aufgelistet:

> ATIPT1

MTELNFHLLPIISDRFTTTTTTSPSFSSHSSSSSSLLSFTKRRRKHQPLVSSIRMEQSRSRNRKDKVVVILGATG
AGKSRLSVDLATRFPSEIINSDKIQVYEGLEITTNQITLQDRRGVPHLLGVINPEHGELTAGEFRSAASNVVKE
ITSRQKVPIIAGGSNSFVHALLAQRFDPKFDPFSSGSLISSDLRYECCFIWVDVSETVLYEYLLRRVDEMMDSG
MFEELSRFYDPVKSGLETRFGIRKAIGVPEFDGYFKEYPPEKKMIKWDALRKAAYDKAVDDIKRNTWTLAKRQVK
KIEMLKDAGWEIERVDATASFKAVMKSSSEKKWRENWEEQVLEPSVKIVKRHLVQN

>ATIPT2

MMMLNPSNGGIEGEKMKKKAKVVVIMGPTGSGKSKLAVDLASHFPVEIINADAMQIYSGLDVLTNKVTVDEQKGV
PHLLGTVSSDMEFTARDFRDFTVPLIEEIVSRNHI PVLVGGTHYYIQAVVSKFLLDDAAEDTECCADVAVSVD
QDMVVESVFRDDL SHGYELLKELDPVAANRIHPNNHRKINQYLSLHASRGLVPSKLYQGKTAENWGCINASRFD
YCLICMDAETAVALDRYVEQRVDAMVDAGLLDEVYDIYKPGADYTRGLRQSIGVREFEDFLKIHLSETCAGHLTSL
SNDDKVMKENLRKILNFPKDDKLRIMLEEAIDRVKLNTRLLRRQKRRVSRLETVFGWNIHYIDATEYILSKSEE
SWNAQVVKPASEIIRCFLLETETESGRDPTSGKSIERDLWTQYVCEACGNKILRGRHEWEHHKQGRTHRKRTRHK
NSQTYKNREVQEAENV

>AT1PT3

MIMKISAMACKQPLPPSPTLDFPPARFGPNMLTLNPYGPDKVVMGATGTGKSRLSVDIATRFRAEIINSDKI
QVHQGLDIVTNKITSEESCGVPHLLGVLPPEADLTAANYCHMANLSIESVLNRGKLP IIVGGSNSYVEALVDDK
ENKFRSRYDCCFLWVDVALPVLHG FVSERVDMVESGMVEEVREFFDFSNSDYSRGIKKAIGFPEFDRFFRNEQF
LNVEDREELLSKVL EIKRNTFELACRQREKIERLRKVKKWSIQQRVDATPVFTKRRSKMDANVAWERLVAGPSTD
TVSRFLLDIASRRPLVEASTAVAAAMERELSRCLVA

>ATIPT4

MKCNDKMVVIMGATGSGKSSLSVDLALHFKAIEIINSDKMQFYDGLKITTNQSTIEDRRGVPHLLGELNPEAGEV
TAAEFRVMAAEAISEITQRKKLPILAGGSNSYIHALLAKSYDPENYPFS DHKGSICSELKYDCCFIWIDVDQSVL
FEYLSRLDLMMKSGMFEEIEAFHRSKKAPKEPLGIWKAIGVQEFDDYLKMYKWDNDMDKWDPMRKEAYEKAVRA
IKENTFQLTKDQITKINKLRNAGWDIKKVDATASFREAIRAAKEGEGVAEMQRKIWNKEVLEPCVKIVRSHLDQP
INYYYYYFYLLKRFLSLN

>ATIPT5

MKPCMTALRQVIQPLSLNFQGNMVDVPPFRKDKVVFVMGATGTGKSRLAIDLATRFPAEIVNSDKIQVYKGLDI
VTNKVTPPEESLGVPHLLGTVHDTYEDFTAEDFQREAIRAVESIVQRDRVPIIAGGSNSYIEALVNDVDFRLRY
NCCFLWVDVSRPVLHSFVSERVDMVDMGLVDEVRRIFDPSSDYSAGIRRAIGVPELDEF LRSEMRNYPAE'TTE
RLLETAIEKIKENTCLLACRQLQKIQRLYKQWKWNMHRVDATEVFLRRGEEADEAWDNSVAHPSALAVEKFLSYS
DDHHLEGANILLPEISAVPPLPAVAVAISR

>ATIPT6

MQQLMTLLSPPLSHSSLLPTVTTKFGSPRLVTTTCMGHAGRKNIKDKVVLITGTTGTGKSRLSVDLATRFFPAEII
NSDKMQIYKGFIVTNLIPLHEQGGVPHLLGQFHPQD GELTPAEFRSLATLSISKLISSKKLP IIVGGSNSFNH
ALLAERFDPDIDPFSGSSLSTICSDLRKCCILWVDVLEPVLFOHLCNRVDQMIESGLVEQLAELYDPVVD SGR
RLGVRKTIGVEEFDRYFRVYPKEMDKGIWDLARKAAYEETVKGMKERTCRLVKKQKEKIMKLIRGGWEIKRLDAT
AAIMAELNQTAKGEGKNGREIWEKHIVDESVEIVKKFLLEV

>ATIPT7

MKFSISSLKQVQPILCFKNKLSKVVNSFLHPKEKVI FVMGATGSGKSR LAIDLATR FQGEIINSDKIQLYKGLD
VLTNKVTPKECRGVPHLLGVDFSEAGNLTATQYSRLASQAI SKLSANNKLP IIVAGGSNSYIEALVNHSSGFLLN
NYDCCFIWVDVSLPVLNSFVSKRVDRMMEAGLLEEVREVFNP KANYSVGIRRAIGVPELHEYL RNESLVD RATS
KMLDVAVKNIKNT EILACRQLKKIQR LHKKWKMSMHRVDATEVFLKRNVEEQDEAWENLVARPSERIVDKFYNN
NNQLKNDDVEHCLAASYGGGSGSRAHNMI

>ATIPT8

MQNLSTFVSPSMIPITSPRLRLPPPRSVPMTTVCMEQSYKQKVVMGATGSGKSCLSIDLATRFSGEIVNSD
KIQFYDGLKVTTNQMSILERC VPHLLGELPPDDSELTTSEFRSLASRSISEITARGNLP IIVAGGSNSFIHALL

VDRFDPKTYPFSSSETSISSGLRYECCFLWVDVSVSVLFEYLSKRVDQMMESGMFEELAGFYDPRYSGSAIRAHGI
 HKTIGIPEFDRYFSLYPPERKQKMSEWDQARKGAYDEAVQEI KENTWRLAKKQIERIMKLKSSGWDIQRLDATPS
 FGRSSREIWDNTVLDES IKVVKRFLVKDKV

>ATIPT9

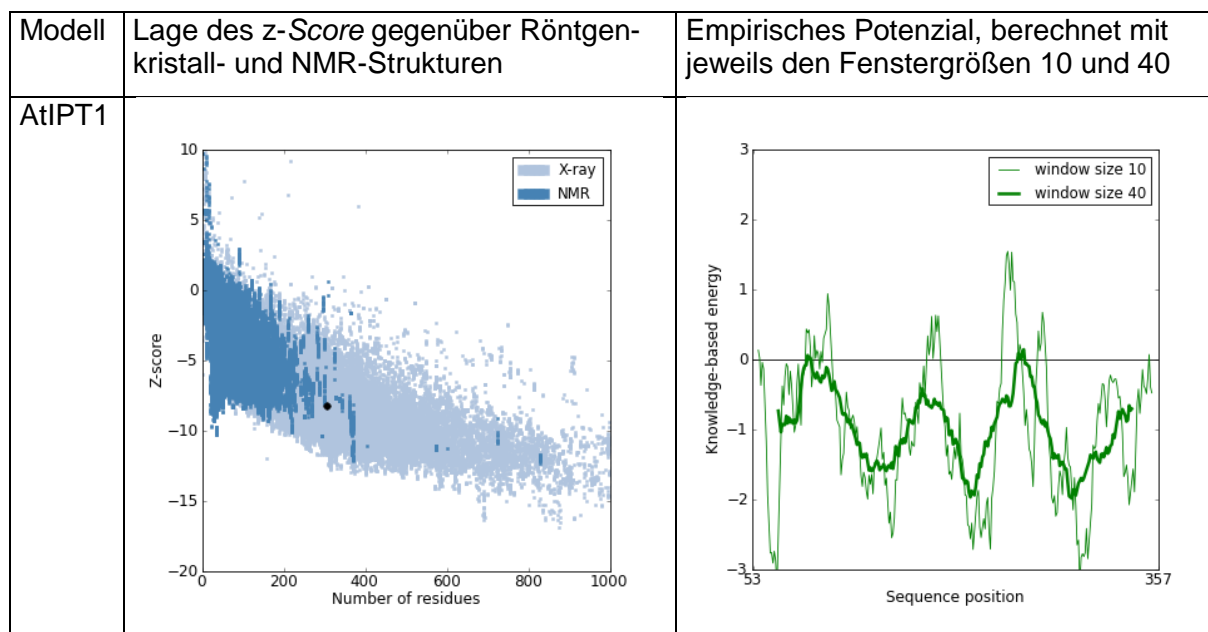
MVIGSGVFLTRTCYLRLQPPSLVLRRRFCAATTACSVPLNGNKKKKSEKEKVIVISGPTGAGKSRLAMELAKRLN
 GEIISADSVQVYKGLDVGSAKPSDSDRKVVPHHLIDILHPSQDYVSGQFYDDGRQATKDI LNRRGRVPIVTGGTGL
 YLRWFMYGKPDVVKPSPEVIAEAHMDLVGFQTEYNWDAAVELVVNAGDPKASSLPRNDWYRLRRSLEILKSTGSP
 PSSFRIPYDSFRVNLVAPDADDFLEDGSSADISI QNIETDLDYDFLCFFLSSPRVALYRSIDFRCEMDLSGPNGV
 LSEARLLDLGLLPNSNPATRAIGYRQAMEYLLQCRRYEGESSPREFYAFLNKFQTASRNFAKRQMTWFRCEPMY
 HVLNASKPLDSILQCIYDAYESEAEEMVEIPESLRMSKDVRDSREASELKGYRSKNRREDCSSVLEWIRSEGCKSE
 ASCVESAI A

A 2 Paarweise Sequenzidentität der AtIPT und sechs potenzieller Template-Strukturen

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1: 1KNQ.A		7.3	6.5	10.6	4.0	6.8	6.7	4.7	5.4	7.9	6.1	7.6	6.4	7.6	5.2
2: 2JAQ.A	8.6		8.7	4.5	4.0	7.9	7.8	3.9	7.1	8.2	6.4	7.3	6.4	7.3	4.6
3: 2QGN.A	12.0	13.7		12.1	6.4	42.1	21.3	20.6	22.0	23.3	23.0	21.9	20.7	21.5	16.8
4: 2YT5.A	4.0	1.5	2.5		1.2	2.1	3.4	2.4	3.0	3.1	3.3	2.9	3.6	3.9	1.5
5: 2ZEG.A	7.4	6.3	6.5	6.1		9.4	10.1	6.2	9.2	8.5	7.6	10.2	8.2	10.6	7.8
6: 3D3Q.A	13.1	13.2	44.4	10.6	9.7		20.7	16.7	20.2	24.5	22.1	17.0	21.6	21.5	16.3
7: AtIPT1	13.7	13.7	23.6	18.2	10.9	21.8		18.9	40.2	51.9	37.6	48.8	38.0	56.4	13.7
8: AtIPT2	12.6	8.8	29.8	16.7	8.8	22.9	24.6		23.5	27.0	30.0	24.0	27.4	26.7	16.8
9: AtIPT3	10.3	11.7	23.0	15.2	9.4	20.0	37.8	17.0		38.1	49.7	37.4	45.0	41.8	12.9
10: AtIPT4	14.3	12.7	23.0	15.2	8.2	22.9	46.2	18.5	36.0		33.3	40.6	35.0	47.9	12.4
11: AtIPT5	11.4	10.2	23.6	16.7	7.6	21.5	34.7	21.2	48.8	34.6		32.5	53.8	36.7	13.1
12: AtIPT6	14.9	12.2	23.3	15.2	10.6	17.1	46.8	17.6	38.1	43.7	33.6		33.7	49.1	12.9
13: AtIPT7	12.0	10.2	21.1	18.2	8.2	20.9	35.0	19.3	44.0	36.2	53.6	32.5		37.0	12.9
14: AtIPT8	14.3	11.7	22.0	19.7	10.6	20.9	52.1	18.9	41.1	49.7	36.7	47.4	37.1		12.9
15: AtIPT9	13.7	10.2	23.9	10.6	10.9	22.1	17.6	16.5	17.6	17.9	18.2	17.3	17.9	17.9	

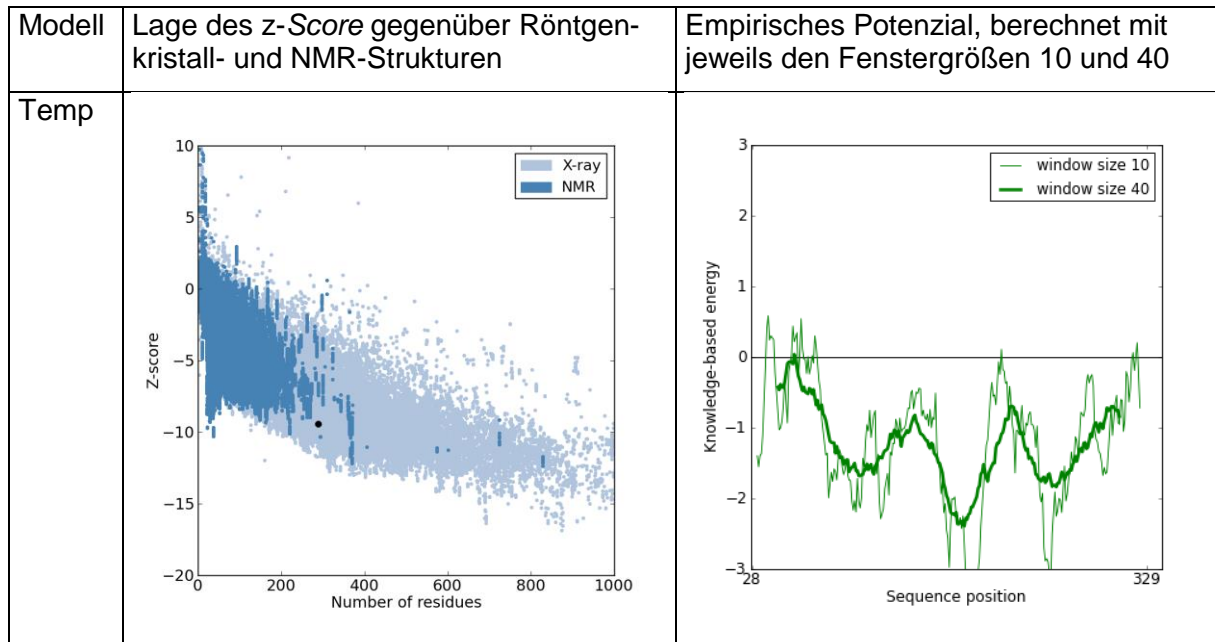
Der Wert in Spalte *i*, Zeile *j* entspricht der Zahl identischen Aminosäuren im *Alignment* der beiden Sequenzen *i* und *j* geteilt durch die Länge der Sequenz *j*; erstellt mit MOE 2011.10, Protein Align 2009.11, Standardoptionen (Substitutionsmatrix: BLOSUM62).

A 3 ProSA-web: z-Score-Darstellungen und empirisches Potenzial für AtIPT



Modell	Lage des z-Score gegenüber Röntgenkristall- und NMR-Strukturen	Empirisches Potenzial, berechnet mit jeweils den Fenstergrößen 10 und 40
AtIPT3		
AtIPT4		
AtIPT5		

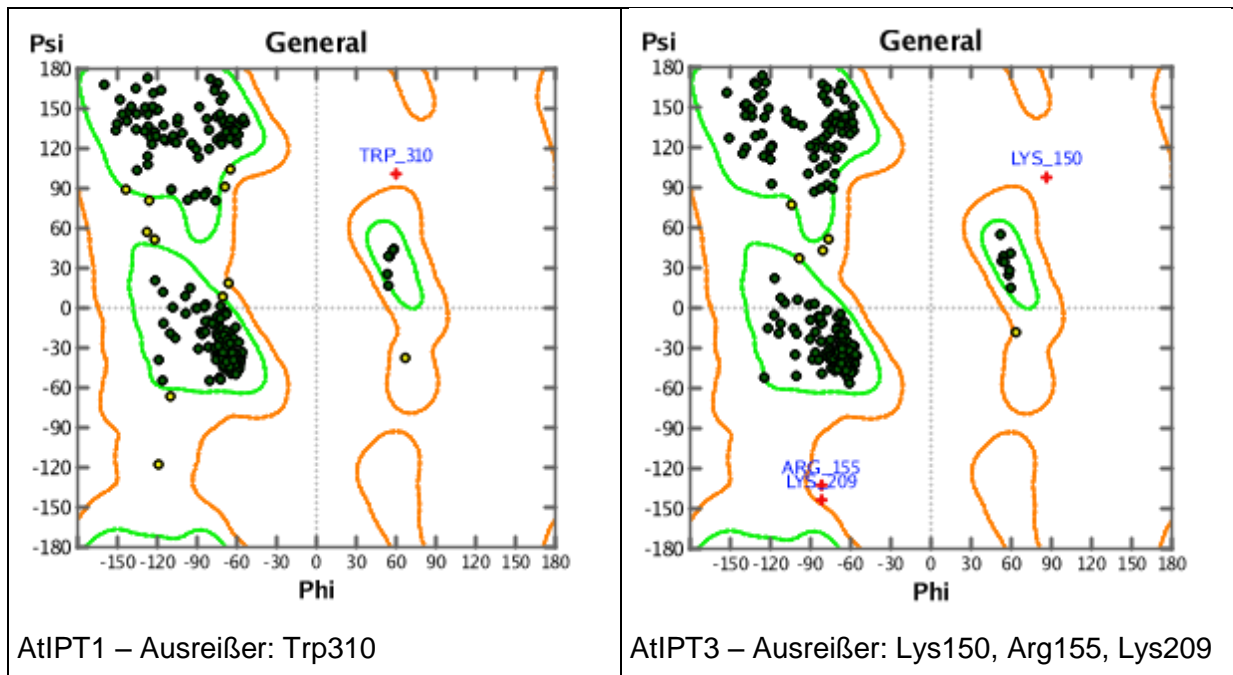
Modell	Lage des z-Score gegenüber Röntgenkristall- und NMR-Strukturen	Empirisches Potenzial, berechnet mit jeweils den Fenstergrößen 10 und 40
AtIPT6	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A black dot is located at approximately (300, -8).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 33 to 342). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>
AtIPT7	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A black dot is located at approximately (300, -8).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 22 to 309). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>
AtIPT8	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A black dot is located at approximately (300, -8).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 31 to 330). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>

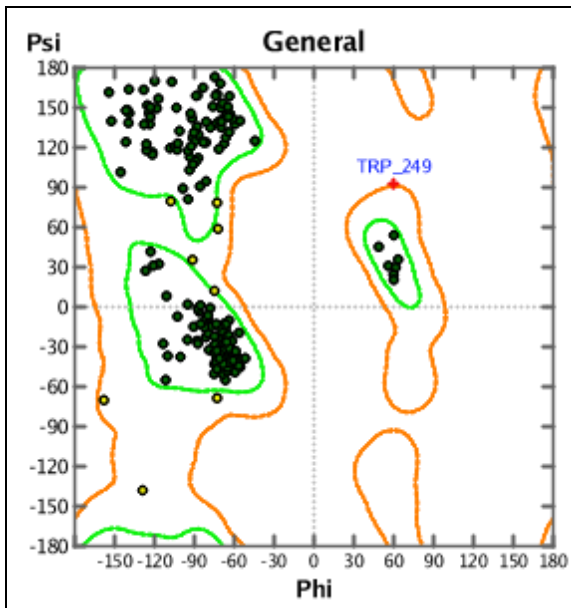


Die Zeile „Temp“ beinhaltet die Diagramme für die *Template*-Struktur, welche zur Modellierung für AtIPT3 bis AtIPT8 verwendet wurde.

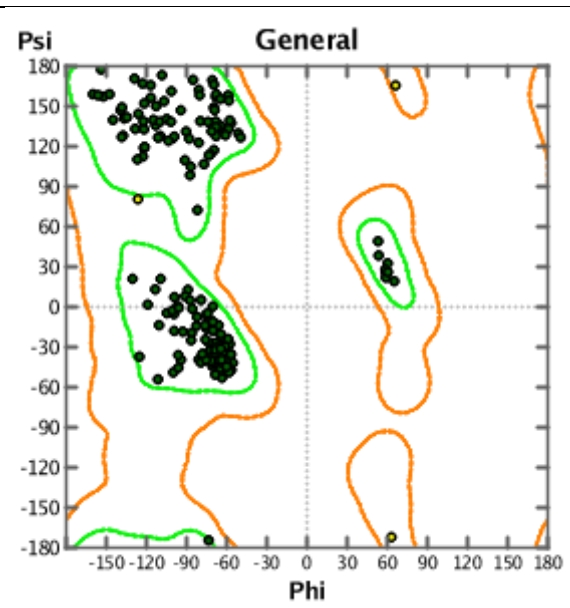
A 4 Ramachandran-Diagramme der AtIPT-Modelle

Die folgende Tabelle zeigt die mit MOE erstellten Ramachandran-Diagramme mit einer Darstellung von Aminosäuren außerhalb der statistisch bevorzugten Bereiche in rot (mit blauer Beschriftung, zur besseren Lesbarkeit extra aufgeführt; die Nummerierung bezieht sich auf die absoluten Werte der Aminosäuren im jeweiligen Modell):

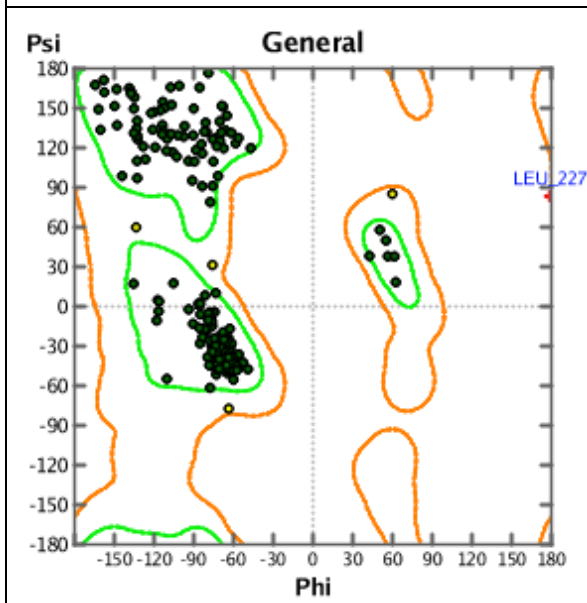




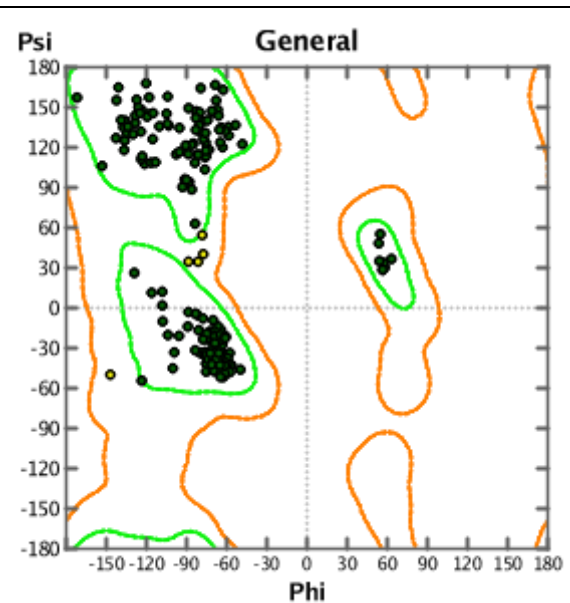
AtIPT4 – Ausreißer: Trp249



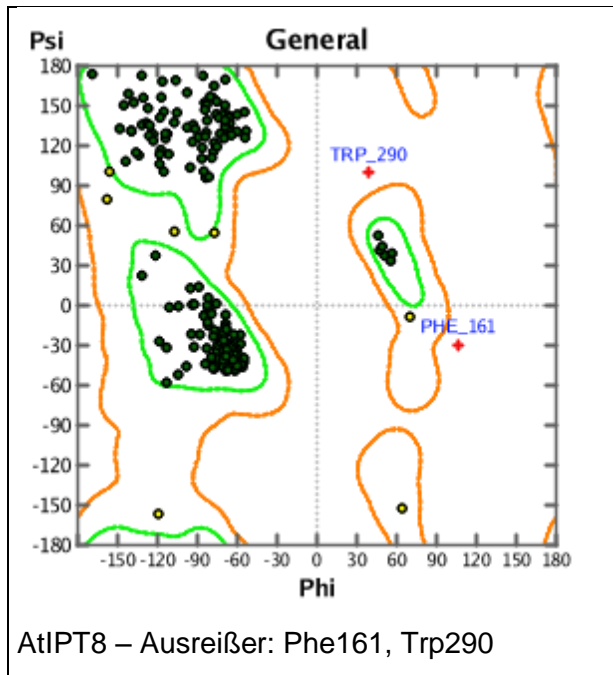
AtIPT5 – Ausreißer: keine



AtIPT6 – Ausreißer: Leu227



AtIPT7 – Ausreißer: keine



A 5 Sequenz-Alignment der AtIPT-Modelle und ihrer beiden *Template*-Sequenzen

001-060

```

IPT1  IRMEQSRSRNRKDKVIVILGATGAGKSRLSVDLATR-FPSEIINSDKIQVYEGLEITTNQ
IPT3  PNMLTLNYPGPKDKVVVIMGATGTGKSRLSVDIATR-FRAEIIINSDKIQVHQGLDIVTNK
IPT4  -----MKCNDKMVVIMGATGSGKSSLSVDLALH-FKAEIINSDKMQFYDGLKITTNQ
IPT5  GNMVDVPPFFRRKDKVVFVMGATGTGKSRLAIDLATR-FPAEIVNSDKIQVYKGLDIVTNK
IPT6  TCMGHAGRNIKDKVVLITGTTGTGKSRLSVDLATRFFPAEIIINSDKMQIYKGFIVTNI
IPT7  SKVNVNSFLHPKEKVI FVMGATGSGKSRLAIDLATR-FQGEIINSDKIQLYKGLDVLTK
IPT8  PMTTVCMEQSYKQKVVVIMGATGSGKSCLSIDLATR-FSGEIVNSDKIQFYDGLKVTTNQ
2ZE7  -----MLLHLIYGPTCSGKTDMAIQIAQE-TGWPVVALDRVQCCPQIATGSGR
3A8T  -----RKEKLLVLMGATGTGKSRLSIDLAH-FPLEVINSDKMQVYKGLDITTNK
    
```

061-120

```

IPT1  ITLQDRRGVPHHLLGVINPEHGELTAGEFRSAASNVVKEITSRQKVP I IAGGSNSFVHAL
IPT3  ITSEESCGVPHHLLGVLPE-ADLTAANYCHMANLSIESVLRGKLP I I VGGNSSYVEAL
IPT4  STIEDRRGVPHHLLGELNPEAGEVTAAEFRVMAAEAI SEITQRKKLP I ILAGGSNSYIHAL
IPT5  VTPEESLGVPHHLLGTVHDTYEDFTAEDFQREAIRAVESIVQRDRVP I IAGGSNSYIEAL
IPT6  IPLHEQGGVPHHLLGQFHPQDGELTPAEFRSLATLSISKLISSKKLP I VGGNSFNHAL
IPT7  VTPKECRGVPHHLLGVFDSEAGNL TATQYSRLASQAI SKLSANNKLP I VAGGSNSYIEAL
IPT8  MSILERC VPHHLLGELPPDDSELTTSEFRSLASRSI SEITARGNLP I IAGGSNSFIHAL
2ZE7  PLESELQSTRRIYLD SRPLTEGILDAESAHRRLIFEVDWRKS-EEGLILEGGSISLLNCM
3A8T  ISVPDRGGVPHHLLGEVDPARGELTPADFRSLAGKAVSEITGRKLPVLVGGNSFIHAL
    
```

121-180

```

IPT1  LAQRFDPKF---DPFSS--GSCLISSDLRYECCFIWVDVS-ETVLYEYLLRRVDEMDS-
IPT3  VDDKEN-----KFRSRYDCCFLWVDVA-LPVLHGFVSRVDMKVES-
IPT4  LAKSYDPEN---YPFSD--HKGSICSELKYDCCFIWIDVD-QSVLFEYLSRLDLMMKS-
IPT5  VNDCV-----DFRLRYNCCFLWVDVS-RPVLHSEFVSRVDMKVM-
IPT6  LAE-RFDPDIDPFSPGS--SLSTICSDLRYKCCILWVDVL-EPVLFQHL CNRVDQMIES-
IPT7  VNHSSG-----FLLNNYDCCFIWVDVS-LPVLNSFVSKRVDRMMEA-
IPT8  LVD-RFDPK---TYPFS--SETSISGLRYECCFLWVDVS-VSVLFEYLSKRVDMMES-
2ZE7  AKSPFW-----RSGFQWHVKRLRLGDSDAFLTRAKQ RVAEMFAIR
3A8T  LVDRFDSSG---PGVFEEGSHSVVSELRYDCCFLWVDVS-VKVLTDYLAKRVDDMLEL-
    
```

181-240

```

IPT1  ----GMFEELSRFYDPVKSGLRETRGIR----KAIIGVPEFDGYFKEYP---PEKKMIKWD
IPT3  ----GMVEEVREFFDFSNSDYSRGIK----KAIIGFPEFDRFFRNEQ-----FLNV
IPT4  ----GMFEEIAEFHRSKKAPKEP-----LGIWKAIIGVQEFDDYLKMYKWDNDM---DKWD
IPT5  ----GLVDEVRRIFDPSSSDYSAGIR----RAIIGVPELDEFRLRSEMR-----NYPA
    
```

```

IPT6 ----GLVEQLAELYDPVVDGRR-----LGVKRTIGVEEFDRYFRVYPK---EMDKGIWD
IPT7 ----GLLEEVREVFNPKANYS-VGIR-----RAIGVPELHEYLRNESL-----VDR
IPT8 ----GMFEELAGFYDPRYSGSAIR---AHGIHKTIIGIPEFDRYFSLYPPE-RKQKMSEWD
2ZE7 EDRPSLLEELAELWNYPAARPILED-----IDGYRCAIRFARKHDLAISQL-----P
3A8T ----GMFDELAEFYSPEDEDHDEDSATRTGLRKAIGVPEFDRYFEKFRPGDVEGEDPGRD
241-300
IPT1 ALRKAAYDKAVDDIKRNTWTLAKRQVKKIEMLKDA-GWEIERVDATASFKAIVMMKSS---
IPT3 EDREELLSKVLEEIKRNTFELACRQREKIERLRKVKKWSIQRVDPVFTKRRSKM----
IPT4 PMRKEAYEKAVRAIKENTFQLTQDQITKINKLRNA-GWDIKKVDATASFRAIRAAGEGE
IPT5 ETTERLLETAIEKIKENTCLLACRQLQKIQRLYQWKWNMHRVDATEVFLR-RGE-----
IPT6 LARKAAYEETVKGMKERTCRLVKKQKEKIMKLIRG-GWEIKRLDATAAIMAELNQSTAK-
IPT7 ATSKMLDVAVKNIKNTEILACRQLKKIQRLLHKKWKMSMHRVDATEVFLKRNVE-----
IPT8 QARKGAYDEAVQEIKENTWRLAKKQIERIMKLKSS-GWDIQRLDATPSFGRS-----
2ZE7 NIDAGRHELIEAIANEYLEHALSQERDFPQWP-----
3A8T RVRRGAFEEAVRAIKENTCHLAKRQIGKILRLKGA-GWDLRRLDATESFRAAMT-SD---
301-338
IPT1 -SEKKWRENWEEQVLEPSVKIVKRHLVQN-----
IPT3 ----DANVAWERLVAGPSTDTVSRFLLDIASRRPLVE-
IPT4 GVAEMQRKIWNKEVLEPCVKIVRSHLDQPINYYYYYFY
IPT5 ----EADAWDNSVAHPSALAVEKFLSY-SDDHHLEGA
IPT6 GEGKNGREIWEKHIVDESVEIVKKFLLEV-----
IPT7 ----EQDEAWENLVARPSEIRIVDKFYNNNNQLKNDDV-
IPT8 ----SREIWDNTVLDESIVVVKRFLVKDKV-----
2ZE7 -----
3A8T -SGEKCTEIVEKQVLEPSVKIVSRFLDE-----

```

A 6 Aminosäuresequenzen der vier KSL-Enzyme

Im Folgenden sind die Aminosäuresequenzen der vier KSL-Enzyme im fasta-Format aufgelistet:

Arabidopsis thaliana ent-Kaurensynthese

>AtKS

```

MSINLRSSGCSSPISATLERRLDSEVQTRANNVSFEQTKKIRKMLEKVELSVSAYDTSWVAMVPSQNAPLF
PQCVKWLLEDNQHDGWSGLDNDHDHQLSKKDVLSSTLASILALKKWGIGERQINKGLQFIELNSALVTDETIQKPT
GFDIIFPGMIKYARDLNLTIPLGSEVVDDMIRKRDLDLKCDEKFSKGREAYLAYVLEGTRNLKDWDLIVKYQRK
NGSLFDSPATTAATAFTQFGNDGCLRYLCSLLQKFEAAVPSVYFPDQYARLSIIVTLESGLIDRDFKTEIKSILDE
TYRYWLRGDEEICLDLATCALAFRLLLAHGYDVSYDPLAFEAESGFSDTLEGYVKNTFVLELFAKAAQSYPHES
ALKKQCCWTKQYLEMELSSWVKTSVRDKYLKKEVEDALAFPSYASLERSDHRRIKLNQSAVENTRVTKTSYRLHN
ICTSDILKLAVDNFQCSIHREEMERLDRWIVENRQLKLFARQKLAYCYFSGAATLFSPELSDARISWAKGGV
LTTVVDDFFDVGGSKEELENLIHLVEKWDLNGVPEYSSEHVEIIFSVLRDTILETGDKAFTYQGRNVTHHIVKIW
LDLLKSMLREAEWSSDKSTPSLEDYMENAYISFALGPVLPATYLI GPPLPEKTVDSHQYNQLYKLVSTMGRLLN
DIQGFKRESAEGKLNVAVSLHMKHERDNRSKEVIIESMKGLAERKREELHKLVLLEEKGSVVPRECKEAFKMSKVL
NLFYRKDDGFTSNDLMSLVKSVIYEPVSLQEEESLT

```

Nicotiana tabacum cis-Abienolsynthese

>NtABS cis-abienol synthase [Nicotiana tabacum]

```

MVLGLRSKIIPLPDHKLGNIKLGSVTNAICHRPCRVRCSHSTASSMEEAKERIRETFGKIELSPSSYDTAWVAMV
PSRYSMNQPCFPQCLDWILENQREDGWSGLNPSHPLLKDSLSSTLASLLALRKWRIGDNQVQRGLGFIETHGWA
VDNKDQISPLGFEIIFPCMINYAELNLDLPLDPNLVNMMLCERELTIERALKNEFEGNMANVEYFAEGLGELCH
WKEMMLRQRHNGSLFDSPATTAALIYHQYDEKCFGYLNSILKLHDNWWPTICPTKIHSNLFVLDALQNLGVDRY
FKTEVVRVLDIYRLWLEKNEEIFSVAHCAMAFRLLRMNNYEVSSEELEGFVDQEHFFTTSSGKLMNHVAILEL
HRASQVAIHERKDHLDKISTWTRNFMEQKLLDKHIPDRSKKEMEFAMRKFGYTFDRVETRRYIESYKMSDFKIL
KAAAYRSSGINNIDLLKFSSEHDFNLQTRHKEELQMKRWFTDCKLEQVGLSQQYLYTSYFIIAAILFEPEYADAR
LAYAKYAIITAVDDFFDFCICKEELQNIIELVREWEGYSTVGFERSERVRIFFLALYKMVEEIAAKAETKQGRCV
KDHLINLWIDMLKMLVELDLWKIKSTTPSIEEYLSVACVTIGVPCFVLTSLYLLGPKLSKDVIESSEVSALCNC
TAAVARLINDIHSYKREQAESSTNMVSIKITQSQGTISEEEAIRQIKEMMESKRRELLGMVLQNKESQLPQVCKD
LFWTTINAAYSITHGDDGYRFPPEEFKNHINDVIYKPLNQYSP

```

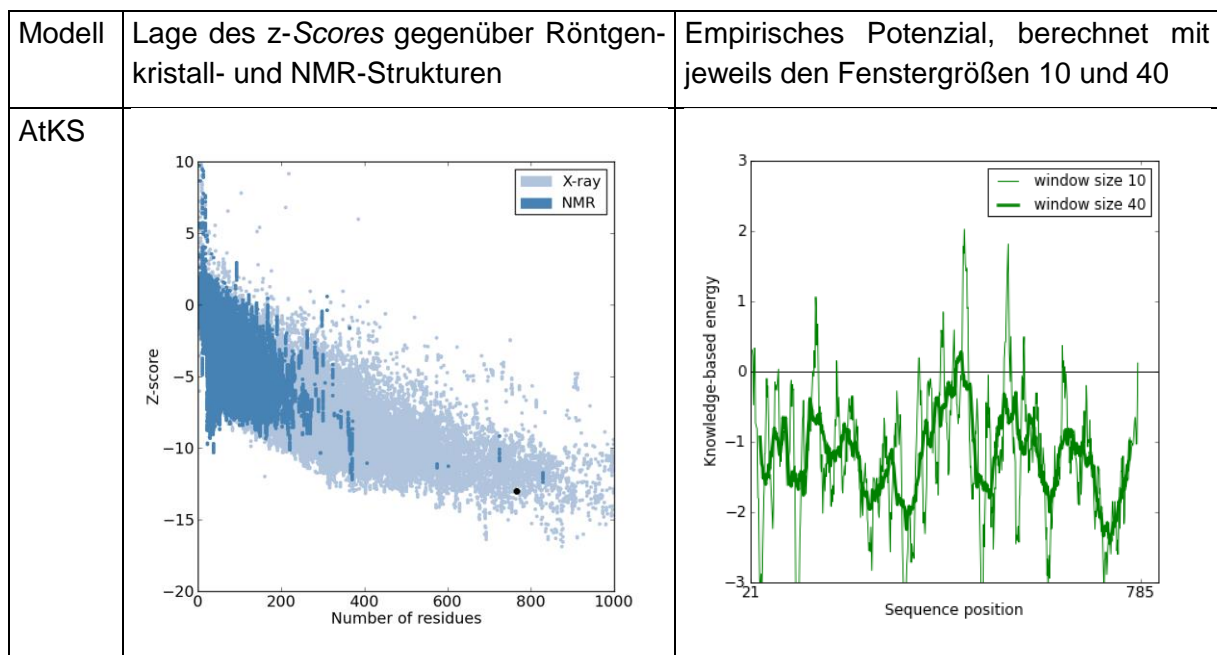
Solanum habrochaites Santalen- und Bergamotensynthese

```
>gi|212727257|gb|ACJ38409.1| santalene and bergamotene synthase [Solanum
habrochaites]
MIVGYRSTIIITLSPKLGNGKTISSNAIFQRSCRVRCSHSTPSSMNGFEDARDRIRESFGKVELSPSSYDTAWVA
MVPSKHSLSNEPCFPQCLDWI IENQREDGWSGLNPSHPLLLKDSLSSTLACLALTKWRVGDEQIK
RGLGFIETQSWAIDNKDQISPLGFEEIFPSSMIKSAEKLNLNLAINKRSTIKRALQNEFTRNIEYMSEGVGELCD
WKEIIKHLHQHQNGSLFDSPATTAALIIYHQHDKKCYEYLNLSILQQHKNWVPTMYPTKIHSLCLLV
DTLQNLGVHRHFKSEIKKALDEIYRLWQQKNEQIFSNVTHCAMAFRLLRMSYYDVSSDELAEFVDEEHFFAISGK
YTSHVEILELHKASQLAIDHEKDDILDKINNWTTRTFMEQKLLNNGFIDRMSKKEVELALRKFYTI
SDLAENRRCIKSYEENNFKILKAAYRSPNIYNKDLFIFSI RN FELCQAQHQEELQQFKRWFEDYRLDQLGIAERY
IHDTYLCVAVIVVPEPELSDARLLYAKYVLLLTIVDDQFDSFASTDECLNIELVERWDDYASVGY
KSEKVKVFFSTLYKSIEELVTIAEIKQGRSVKNHLLNLWLELVKMLMERVEWFGSKTIPSEI EYLYVTSITFGA
RLIPLTTQYFLGIKISEDILESEIYGLCNCTGRVLRILNDLQDSKKEQKEDSVTIVTLLMKSMS
EEEEAIMKIKEILEMNRRELLKMVLVQKKGSQLPQICKDIFWRTSNWADFIYLTQDGYRIAEEMKNHIDEVFKPL
NH
```

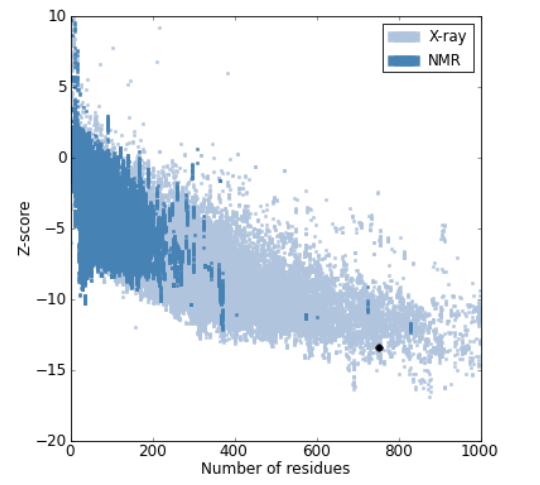
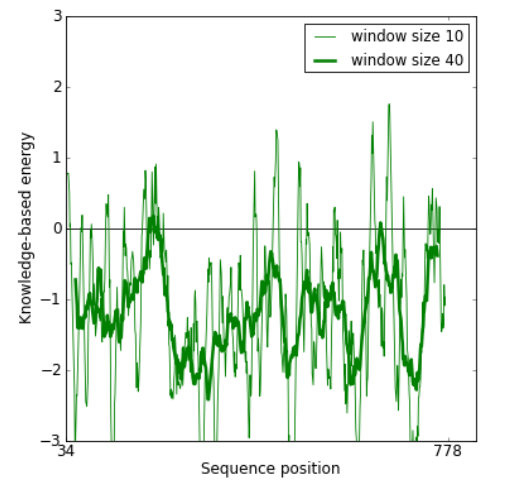
Solanum lycopersicum Phellandrensynthese

```
>gi|226439925|gb|AC056896.1| terpene synthase [Solanum lycopersicum]
MIVGYRSTIIITLSPKLGNGKTISSNAIFQRSCRVRCSHSTTSSMNGFEDARDRIRESFGKLELSPSSYDTAWVA
MVPSRHLSLSNEPCFPQCLDWI IENQREDGWSGLNPTHPLLLKDSLSSTLACLALTKWRVGDEQIK
RGLGFIETQSWAVDNKDQISPLGFEEIFSSMIKSAEKLNLNLPLNLHLVNLVKCKRDSTIKRNVEYMGEGVGE LC
DWKEMIKLHQHQNGSLFDSPATTAALIIYHQHDKKCYEYLNLSIFQQHKNWVPTMYPTKVHSLCLLV
VDTLQNLGVHRHFKSEIKKALDEIYRLWQQKNEQIFSNVTHCAMAFRLLRMSYYDVSSDELAEFVDEEHFFATNG
KYKSHVEILELHKASQLAIDHEKDDILDKINNWTTRTFMEQKLLNNGFIDRMSKKEVELALRKFYTI
TSHLAENRRYIKSYEENNFKILKAAYRSPNINNDLLAFS IHD FELCQAQHREELQQKRWFEYRLDQLGLAER
YIHASYLFGVTVIPEPELSDARLMYAKYVMLLTIVDDHFESFASKDECFNIELVERWDDYASVGY
YKSEKVKVFFSVFYKSIEELATIAEIKQGRSVKNHLINLWLELMKMLMERVEWCSGKTIPSEI EYLYVTSITFC
AKLIPLSTQYFLGIKISKDLLESEICGLWNCGRVMRILNDLQDSKREQKEVSINLV TLLMKSM
SEEEAIMKIKEILEMNRRELLKMVLVQKKGSQLPQLCKDIFWRTSKWAHFTYSQTDGYRIAEEMKNHIDEVFKP
LNH
```

A 7 ProSA-web: z-Score-Darstellungen und empirisches Potenzial für KSL



Modell	Lage des z-Scores gegenüber Röntgenkristall- und NMR-Strukturen	Empirisches Potenzial, berechnet mit jeweils den Fenstergrößen 10 und 40
NtABS	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A single black dot is highlighted at approximately (750, -12).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 29 to 792). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>
ShSBS	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A single black dot is highlighted at approximately (750, -12).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 34 to 777). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>
SIPHS	<p>Scatter plot showing Z-score (y-axis, -20 to 10) versus Number of residues (x-axis, 0 to 1000). Data points are categorized by structure type: X-ray (light blue) and NMR (dark blue). A single black dot is highlighted at approximately (750, -12).</p>	<p>Line plot showing Knowledge-based energy (y-axis, -3 to 3) versus Sequence position (x-axis, 34 to 778). Two lines represent different window sizes: window size 10 (light green) and window size 40 (dark green).</p>

Modell	Lage des z-Scores gegenüber Röntgenkristall- und NMR-Strukturen	Empirisches Potenzial, berechnet mit jeweils den Fenstergrößen 10 und 40
Temp		

Die Zeile „Temp“ beinhaltet die Diagramme für die *Template*-Struktur mit dem PDB-Identifikator 3P5R, welche zur Modellierung der KSL verwendet wurde.

A 8 Pharmakophorbeschreibung für die Suche nach AtIPT-Liganden

```
#moe:ph4que 2009.1
#pharmacophore 5 tag t value *
scheme t Unified matchsize i -1 title t $ comment s $ smask i 476156
#feature 5 expr tt color ix x r y r z r r r ebits ix gbits ix
Ani|Acc df2f2 135.326 41.59 -30.707 1 0 300
Ani|Acc df2f2 133.903 40.968 -32.518 1 0 300
Ani|Acc df2f2 133.038 38.073 -32.527 1 0 300
Cat|Don f2f20d 132.755358816354 40.5562886079135 -28.2726908941773 1 0 300
Acc f2f20d 134.971 36.556 -30.961 1 0 300
#volumesphere 27 x r y r z r r r
136.022895774804 44.0098646914703 -33.2614058002364 2.4
138.052728653187 40.6179231701535 -32.9900883616647 2.4
137.889483476523 37.0050313657266 -33.346874622046 2.4
138.408061398193 33.33545610687 -33.0837025268993 2.4
129.888058058103 41.6209659105516 -28.3504257087916 2.4
132.790261332295 44.1504912520177 -30.5679502869316 2.4
129.90055614175 37.8077150419008 -28.1025153912006 2.4
131.341407012666 34.4571171313178 -29.3239017456172 2.4
132.993828179554 42.801826547005 -27.2218909691101 2.4
136.678518406122 44.0168577914187 -29.3553535231868 2.4
138.054686362163 42.7866857528165 -31.4877980791744 2.4
137.868504662772 39.702251685932 -29.3538733546985 2.4
136.187829090754 42.402306576641 -27.8711246773265 2.4
135.302643783106 39.6325369110887 -26.6423411505194 2.4
130.869626711769 40.3478583853316 -26.1415178583366 2.4
132.932403929247 37.4209233522884 -26.5581350186504 2.4
136.067564070871 36.2404779193926 -27.4985327212735 2.4
138.28546966677 36.494337974163 -30.347394387437 2.4
134.782285199914 35.1107344455554 -34.5907902657592 2.4
134.647953477869 38.9641642985403 -35.1755324886718 2.4
134.826551936748 41.8918279712557 -35.3213530671156 2.4
132.508169863205 44.2371453962292 -33.4079450387463 2.4
129.344966301698 42.1192616001318 -31.8555133016009 2.4
130.668721914252 40.73785416625 -35.4135261668307 2.4
130.035840527581 38.7296227247597 -31.6394772959357 2.4
```

```

132.169241932136 38.0701562204076 -35.5416100088813 2.4
130.413826700248 34.6725109251452 -33.4848490057585 2.4
#volume 27 size i expr tt color ix ebits ix gbits ix
1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $ 606060 1 300 1 $
#endpharmacophore

```

A 9 Formale Definition des ACO-Algorithmus

Jedem Freiheitsgrad x_i ist ein Vektor p_{ij} zugeordnet, der für alle möglichen diskreten Realisierungen j des Freiheitsgrades i einen Wahrscheinlichkeitswert speichert, der die Möglichkeit repräsentiert, dass sich eine Ameise für diese Realisierung entscheidet.

$$p_{ij} = \frac{\left(1 - \frac{1}{1+\gamma \cdot \tau_{ij}}\right) \left(1 - \frac{1}{1+\delta \cdot \eta_{ij}}\right)^\beta}{\sum_{l=1}^{n_i} \left(1 - \frac{1}{1+\gamma \cdot \tau_{il}}\right) \left(1 - \frac{1}{1+\delta \cdot \eta_{il}}\right)^\beta} \quad \text{für Torsionswinkel und}$$

$$p_{kj} = \frac{\tau_{kj}}{\sum_{l=1}^{n_k} \tau_{kl}} \quad \text{für andere Freiheitsgrade}$$

wobei $\gamma = \frac{200}{\tau_{\max_i}}$ und $\delta = 0,3$ experimentell ermittelte Parameter

τ_{\max_i} der maximale Pheromonwert des Freiheitsgrades i

n_i (bzw. n_k) Anzahl von Werten, die der Freiheitsgrad i (bzw. k) annehmen kann

η heuristische Skalierung (Torsionwinkelpotenziale für rotierbare Bindungen)

Gleichung 6: Definition der Wahrscheinlichkeit der Realisierung eines bestimmten Wertes für einen Freiheitsgrad

Mit Hilfe von Gleichung 7 ermittelt PLANTS in jeder Iteration neue Pheromonvektoren, indem die beste Lösung s^{ib} der aktuellen Iteration in die Aufdatierung aller Pheromonvektoren τ_{ij} einfließt:

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + I_{ij}^{ib}(t)\Delta\tau_{ib}(t)$$

wobei $\Delta\tau_{ib}(t) = |f(s^{ib})|$, falls $f(s^{ib}) < 0$ und $\Delta\tau_{ib}(t) = 0$, sonst. Ferner

$$I_{ij}^{ib}(t) = 1, \text{ falls } i \text{ Translationsfreiheitsgrad und für } s^{ib} \text{ gilt } i \in \{j-1, j, j+1\}$$

$$I_{ij}^{ib}(t) = 1, \text{ falls } i \text{ Rotations- oder Torsionsfreiheitsgrad und für } s^{ib} \text{ gilt:}$$

$$i \in \{(j-2) \bmod n_i, (j-1) \bmod n_i, j \bmod n_i, (j+1) \bmod n_i, (j+2) \bmod n_i\}$$

$$I_{ij}^{ib}(t) = 0, \text{ sonst;}$$

ρ ist die Pheromonverdunstungsrate.

Gleichung 7: Aufdatierung der Pheromonvektoren

$$iterations = \sigma \cdot \frac{10}{m} (100 + 50 \cdot lrb + 5 \cdot lha)$$

wobei: m Größe der Ameisenkolonie

σ Skalierungsfaktor für Iterationszahl

lrb Anzahl der frei rotierbaren Bindungen des Liganden

lha Anzahl der Schweratome des Liganden

Gleichung 8: Berechnung der Iterationsanzahl für den PLANTS-Algorithmus

Um einerseits favorisierte Werte für Freiheitsgrade wieder zu benutzen, aber andererseits auch neue Werte für Freiheitsgrade zu versuchen, ist es nötig, ein Gültigkeitsintervall für Pheromonvektoren $\tau_{ij}(t)$ zu definieren, so dass $\tau_{\min} \leq \tau_{ij}(t) \leq \tau_{\max}$ (für das allgemeine MAX-MIN Ant System siehe Stützle und Hoos, ³⁸⁸):

$$\tau_{\min} = \frac{\tau_{\max} \left(1 - \sqrt[n]{p_{best}}\right)}{\left(\frac{n}{2} - 1\right) \sqrt[n]{p_{best}}}, \text{ wobei } p_{best} = 0,9 \text{ für PLANTS verwendet wird.}$$

$$\tau_{\max} = \frac{|f(s^{gb})|}{\rho}, \text{ wobei } f(s^{gb}) \text{ die beste Bewertung darstellt, die}$$

seit Beginn des Algorithmus gefunden wurde.

Gleichung 9: Definition des Gültigkeitsintervalls für Pheromonwerte

A 10 PLANTS-Konfigurationsdateien für Versuche zu *Cluster-Option*

Unabhängiges *Docking*

```
# setup file for PLANTS docking 1 using PLANTS (20110628_17_24)
# search algorithm settings
```



```
aco_ants 100
aco_evap 0.15
aco_sigma 1.0

flip_amide_bonds 1
flip_planar_n 1

# cluster algorithm settings
cluster_structures 1
cluster_rmsd 2.0

##protein
protein_file /raid/home/rklein/DOCK/plants/test/LimoSyn.mol2

# mol2 ligand files or multi-mol2 database
ligand_file /raid/home/rklein/DOCK/plants/test/alphaTerpinylCation.mol2

# binding site definition
bindingsite_center 23 55 -53
bindingsite_radius 10

# scoring parameters
scoring_function chemplp
ligand_intra_score lj

# flexible side-chain

# output directory
output_dir /raid/home/rklein/DOCK/plants/test/20110628_17_24/PLANTS_1
```

Diversitätsorientiertes Docking

```
# setup file for PLANTS docking 1 using PLANTS (20110629_07_26)
# search algorithm settings
aco_ants 100
aco_evap 0.15
aco_sigma 1.0

flip_amide_bonds 1
flip_planar_n 1

# cluster algorithm settings
cluster_structures 30
cluster_rmsd 2.0

##protein
protein_file /raid/home/rklein/DOCK/plants/test/LimoSyn.mol2

# mol2 ligand files or multi-mol2 database
ligand_file /raid/home/rklein/DOCK/plants/test/alphaTerpinylCation.mol2

# binding site definition
bindingsite_center 23 55 -53
bindingsite_radius 10

# scoring parameters
scoring_function chemplp
ligand_intra_score lj

# flexible side-chain
```



```

fi
if [ ! $lig ] ; then
    echo "Ligand file (absolute path) not given - abort!"
    exit 1
fi
if [ ! $rec ] ; then
    echo "Receptor file (absolute path) not given - abort!"
    exit 1
fi
if [ ! $r ] ; then
    echo "Active site radius not given - abort!"
    exit 1
fi
if [ ! $x ] ; then
    echo "x-coordinates of the active site center not given - abort!"
    exit 1
fi
if [ ! $y ] ; then
    echo "y-coordinates of the active site center not given - abort!"
    exit 1
fi
if [ ! $z ] ; then
    echo "z-coordinates of the active site center not given - abort!"
    exit 1
fi

if [ $testMode -eq 0 ] ; then
    if [ ! $anaFile ] ; then
        echo "No analysis request file given - abort!"
        exit 1
    fi
fi

# All files that should exist on start of this script - existence will be checked
mainDir=/raid/home/rklein/DOCK/plants/transform
runPLANTS=$mainDir/runPLANTS
drawPLPs=$mainDir/drawPLPs
evaPLANTS=$mainDir/evaPLANTS
metaPLANTS=$mainDir/metaPLANTS
corDistScores=$mainDir/corDistScores
violinPLANTS=$mainDir/violinPLANTS
violinOverlay=$mainDir/violinOverlay
violinTops=$mainDir/violinTops

scoringBackup=$mainDir/PLANTSproteinLigandScoringFunction_CHEMPLP_BACKUP.cc
originalScoring=/raid/home/rklein/DOCK/plants/trunk/PLANTSproteinLigandScoringFunction_CHEMPLP
.cc

# Instance-related variables
currentDate=`date '+%Y%m%d_%H_%M'`
workDir=$mainDir/transform_${currentDate}
mkdir $workDir
logFile=$workDir/transformer.log
printf "" > $logFile

if [ $testMode -eq 1 ] ; then
    echo "TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE"
    echo "TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE TESTMODE" >>
$logFile
fi

echo "$mainDir/transformer started by $USER on $currentDate"
echo "$mainDir/transformer started by $USER on $currentDate" >> $logFile

# Output of user choices
echo "Potential minima for C.cat-Aro interaction: ${aro[*]}"
echo "Potential minima for C.cat-Aro interaction: ${aro[*]}" >> $logFile
echo "Potential minima for C.cat-MethioninS interaction: ${mets[*]}"
echo "Potential minima for C.cat-MethioninS interaction: ${mets[*]}" >> $logFile
echo "Number of docking poses to be produced: $times"
echo "Number of docking poses to be produced: $times" >> $logFile
echo "Ligand file (absolute path): $lig"
echo "Ligand file (absolute path): $lig" >> $logFile
echo "Receptor file (absolute path): $rec"
echo "Receptor file (absolute path): $rec" >> $logFile
echo "Active site: $r Angstroms around ($x, $y,$z)"

```

```
echo "Active site: $r Angstroms around ($x, $y,$z)" >> $logFile
if [ $cluster ] ; then
    echo "Cluster-RMSD for pose diversity: $cluster"
    echo "Cluster-RMSD for pose diversity: $cluster" >> $logFile
fi
if [ $flex ] ; then
    echo "Residues flexible during docking: ${flex[*]}"
    echo "Residues flexible during docking: ${flex[*]}" >> $logFile
fi
if [ $anaFile ] ; then
    echo "Analysis request file: $anaFile"
    echo "Analysis request file: $anaFile" >> $logFile
fi

# Check for existence
if [ ! -f $rec ] ; then
    echo "Receptor file $rec not found - abort!"
    exit 1
fi
if [ ! -f $lig ] ; then
    echo "Ligand file $lig not found - abort!"
    exit 1
fi
if [ ! -f $runPLANTS ] ; then
    echo "Script to run PLANTS $runPLANTS not found - abort!"
    exit 1
fi
if [ ! -f $evaPLANTS ] ; then
    echo "Script to evaluate PLANTS dockings $evaPLANTS not found - abort!"
    exit 1
fi
if [ ! -f $scoringBackup ] ; then
    echo "Parameter source code file $scoringBackup not found - abort!"
    exit 1
fi
if [ $testMode -eq 0 ] ; then
    if [ ! -f $anaFile ] ; then
        echo "Analysis request file $anaFile not found - abort!"
        exit 1
    fi
fi

printf "\nWill start the different tasks going by ID $currentDate.\n"
printf "\nWill start the different tasks going by ID $currentDate.\n" >> $logFile

# STEP 1: Produce requested PLANTS executables
printf "\nCOMPILING different program versions..."
printf "\nCOMPILING different program versions..." >> $logFile

for aroStep in ${aro[*]}
do
    for metsStep in ${mets[*]}
    do
        # Modify respective numbers in source code
        # ATTENTION: This is specifically tailored for one very special version of
        # the source code and refers to line numbers - make sure this is applicable!

        # Small check on applicability for lines 796 and 801
        if [ `sed -n '796 p' $scoringBackup | sed -e 's/\t//g' -e 's/[ ]*//g'` !=
"en=plpEnergy(paramSteric,-8.0,dist);" ] \
        || [ `sed -n '801 p' $scoringBackup | sed -e 's/\t//g' -e 's/[ ]*//g'` !=
"en=plpEnergy(paramSteric,-32.0,dist);" ] ; then
            echo "The template source code file $scoringBackup has been modified in
a way that this script is not able to work with it anymore - abort!"
            exit 1
        fi

        sed -e "796 s/-8\.0/${aroStep}/" -e "801 s/-32\.0/${metsStep}/" $scoringBackup
> $originalScoring
        ssh rklein@brandt69 "cd ${originalScoring%/*}; make; cp
${originalScoring%/*}/PLANTS $workDir/PLANTS_${aroStep}_${metsStep}" &>> $logFile
        cp $originalScoring $workDir/scoring_${aroStep}_${metsStep}.cc
    done
done
# Restore original source code file from backup
cp $scoringBackup $originalScoring
```

```

echo " - done."
echo " - done." >> $logFile

# STEP 2: Run docking, using a suitable identifier
printf "\nDOCKING with the different program versions..."
printf "\nDOCKING with the different program versions..." >> $logFile

for plantsExec in `ls -lF $workDir | grep "PLANTS_.*\*" | awk '{print $8}' | sed 's/\*$//'\`
do
    if [ ! $cluster ] && [ ! $flex ]; then
        $runPLANTS times=$times r=$r lig=$lig rec=$rec x=$x y=$y z=$z id=$currentDate
    plants=$plantsExec >> $logFile
    elif [ ! $cluster ] ; then
        $runPLANTS times=$times r=$r lig=$lig rec=$rec x=$x y=$y z=$z id=$currentDate
    plants=$plantsExec flex=`echo ${flex[*]} | sed 's/ /,/g'\` >> $logFile
    elif [ ! $flex ] ; then
        $runPLANTS times=$times r=$r lig=$lig rec=$rec x=$x y=$y z=$z id=$currentDate
    plants=$plantsExec cluster=$cluster >> $logFile
    else
        $runPLANTS times=$times r=$r lig=$lig rec=$rec x=$x y=$y z=$z id=$currentDate
    plants=$plantsExec cluster=$cluster flex=`echo ${flex[*]} | sed 's/ /,/g'\` >> $logFile
    fi
done

echo " - done."
echo " - done." >> $logFile

# STEP 3: Create a diagram depiction of the used potentials
printf "\nCREATING DIAGRAMS of used potentials..."
printf "\nCREATING DIAGRAMS of used potentials..." >> $logFile

$drawPLPs $currentDate >> $logFile

printf "\n - done.\n"
printf "\n - done.\n" >> $logFile

# STEP 4: Evaluate dockings folder-wise
printf "\nEVALUATING dockings..."
printf "\nEVALUATING dockings..." >> $logFile

if [ $testMode -eq 1 ] ; then
    printf "\n\nSkipping evaluation due to test mode.\n\n"
    printf "\n\nSkipping evaluation due to test mode.\n\n" >> $logFile
else
    for plantsExec in `ls -lF $workDir | grep "PLANTS_.*\*" | awk '{print $8}' | sed
's/\*$//'\`
    do
        $evaPLANTS id=$currentDate ana=$anaFile ver=`echo "$plantsExec" | sed
's/PLANTS_//'\` >> $logFile
    done
fi

printf "\n - done.\n"
printf "\n - done.\n" >> $logFile

# STEP 5: Evaluate dockings overall
printf "\nMETA-ANALYSIS, CORRELATIONS AND VIOLINS..."
printf "\nMETA-ANALYSIS, CORRELATIONS AND VIOLINS..." >> $logFile

if [ $testMode -eq 1 ] ; then
    printf "\n\nSkipping meta-analysis due to test mode.\n\n"
    printf "\n\nSkipping meta-analysis due to test mode.\n\n" >> $logFile
else
    $metaPLANTS id=$currentDate >> $logFile
    $corDistScores id=$currentDate >> $logFile
    $violinPLANTS id=$currentDate >> $logFile
    $violinOverlay id=$currentDate >> $logFile
    $violinTops $currentDate >> $logFile
fi

printf "\n - done.\n"
printf "\n - done.\n" >> $logFile

```



```

        exit 1
    fi
    if [ ! $y ] ; then
        echo "No y coordinate for binding pocket given - abort!"
        exit 1
    fi
    if [ ! $z ] ; then
        echo "No z coordinate for binding pocket given - abort!"
        exit 1
    fi
    if [ ! $ID ] ; then
        echo "No investigation ID given - abort!"
        exit 1
    fi
    if [ ! $PLANTSexec ] ; then
        echo "No modified PLANTS executable given - abort!"
        exit 1
    fi
    if [ ! -f $ligandFile ] ; then
        echo "Ligand file not found - abort!"
        exit 1
    fi
    if [ ! -f $receptorFile ] ; then
        echo "Receptor file not found - abort!"
        exit 1
    fi

    # Define needed files/folders
    mainDir=/raid/home/rklein/DOCK/plants/transform/transform_{$ID}
    modPLANTS=$mainDir/$PLANTSexec
    version=`echo "$PLANTSexec" | sed 's/PLANTS_//`
    logFile=$mainDir/runPLANTS.log

    if [ ! -d $mainDir ] ; then
        echo "Folder $mainDir not valid - abort!"
        exit 1
    fi
    if [ ! -x $modPLANTS ] ; then
        echo "$modPLANTS cannot be executed - abort!"
        exit 1
    fi

    # Inform user about what will happen, log setup information
    echo "$USER started runPLANTS; ID of this run: $ID, version $version, log file: $logFile."
    echo "$USER started runPLANTS; ID of this run: $ID, version $version, log file: $logFile." >
    $logFile
    echo "Will attempt docking $ligandFile into $receptorFile for $times times (docking center:
    $x, $y, $z , radius $radius)"
    echo "Will attempt docking $ligandFile into $receptorFile for $times times (docking center:
    $x, $y, $z , radius $radius)" >> $logFile
    if [ $flexResidues ] ; then
        echo "The following amino acids will be kept flexible: ${flexResidues[*]}"
        echo "The following amino acids will be kept flexible: ${flexResidues[*]}" >> $logFile
    else
        echo "No flexible side chains."
        echo "No flexible side chains." >> $logFile
    fi
    if [ $clusterRmsd ] ; then
        echo "Will cluster $times docking results into clusters within $clusterRmsd Angstroms
        RMSD."
        echo "Will cluster $times docking results into clusters within $clusterRmsd Angstroms
        RMSD." >> $logFile
    else
        echo "Will run $times independent dockings."
        echo "Will run $times independent dockings." >> $logFile
    fi

    printf "\n$PLANTSexec: docking..."
    printf "\n$PLANTSexec: docking..." >> $logFile

    # The dockings shall be done in one sweep and avoid clustering within an RMSD of $clusterRmsd
    Angstroms
    # This is indicated by the cluster algorithm settings in the PLANTS setup file
    if [ $clusterRmsd ] ; then
        workDir=$mainDir/docked_$version

```

```
        setupFile=${mainDir}/${PLANTSexec}_setup.txt
        plantsCommand="ssh -X $USER@brandt69 $modPLANTS --mode screen $setupFile"

echo "# setup file for PLANTS docking ${i} using ${modPLANTS} ($ID)
# search algorithm settings
aco_ants 100
aco_evap 0.15
aco_sigma 1.0

flip_amide_bonds 1
flip_planar_n 1

# cluster algorithm settings
cluster_structures $times
cluster_rmsd $clusterRmsd

# protein
protein_file $receptorFile

# mol2 ligand files or multi-mol2 database
ligand_file $ligandFile

# binding site definition
bindingsite_center $x $y $z
bindingsite_radius $radius

# scoring parameters
scoring_function chemplp
ligand_intra_score lj

# flexible side-chain " > $setupFile

for flexRes in ${flexResidues[*]}
do
    echo "flexible_protein_side_chain_string $flexRes" >> $setupFile
done

echo "
# output directory
output_dir $workDir" >> $setupFile

    # execute PLANTS docking
    $plantsCommand >> $logFile
    mv $setupFile $workDir
    echo " - done."
    echo " - done." >> $logFile

# The dockings shall be done subsequently and independent from each other
else

    for i in `seq 1 $times`
    do
        workDir=$mainDir/docked_${version}_${i}
        setupFile=${mainDir}/${version}_${i}_setup.txt
        plantsCommand="ssh -X $USER@brandt69 $modPLANTS --mode screen $setupFile"

        printf "%i "

echo "# setup file for PLANTS docking ${i} using ${modPLANTS} ($ID)
# search algorithm settings
aco_ants 100
aco_evap 0.15
aco_sigma 1.0

flip_amide_bonds 1
flip_planar_n 1

# cluster algorithm settings
cluster_structures 1
cluster_rmsd 2.0

# protein
protein_file $receptorFile

# mol2 ligand files or multi-mol2 database
ligand_file $ligandFile
```



```

fi
if [ ! -f $anaFile ] ; then
    echo "Analysis request file $anaFile not found - abort!"
    exit 1
fi

# main folder for these investigations
mainDir=/raid/home/rklein/DOCK/plants/transform/transform_${identifier}
logFile=${mainDir}/evaPLANTS.log
inLogFile=${mainDir}/runPLANTS.log
version=`echo "$PLANTSexec" | sed 's/PLANTS_//`
sumDir=${mainDir}/docked_.$version

if [ ! -f $inLogFile ] ; then
    echo "Docking logfile $inLogFile could not be found - abort!"
    exit 1
fi

# Get data from docking log file
clustered=`sed -n '4 p' $inLogFile | grep -n "cluster" | sed 's/:.*//`
ligandFile=`grep "^Will attempt docking" $inLogFile | awk '{print $4}'`
times=`grep "^Will attempt docking" $inLogFile | awk '{print $8}'`

if [ ! -f $ligandFile ] ; then
    echo "Original ligand file $ligandFile not found - abort!"
    exit 1
fi

# In case of independent dockings, prepare the input for the evaluation
if [ ! $clustered ] ; then

    # Create protein_bindingsite_fixed file
    mkdir $sumDir
    echo $PLANTSexec
    cp ${sumDir}_1/protein_bindingsite_fixed.mol2 $sumDir

    # Create a sorted ranking file
    rankingFile=${sumDir}/ranking
    printf "" > $rankingFile.temp
    for i in `seq 1 $times`
    do
        scoresLine=`tail -1 ${sumDir}_${i}/ranking.csv`
        printf "%02d,${scoresLine#*,}\n" ${i} >> ${rankingFile}.temp
    done
    sort -nk2 -t", " ${rankingFile}.temp > ${rankingFile}.txt
    rm ${rankingFile}.temp
    echo
    "TOTAL SCORE,SCORE_RB_PEN,SCORE_NORM_HEVATOMS,SCORE_NORM_CRT_HEVATOMS,SCORE_NORM_WEIGHT,SCORE_
    NORM_CRT_WEIGHT,SCORE_RB_PEN_NORM_CRT_HEVATOMS,SCORE_NORM_CONTACT,EVAL,TIME" >
    ${rankingFile}.csv
    ligName=`echo ${ligandFile##*/} | sed 's/\.mol2//`
    docked_proteins_file=${sumDir}/docked_proteins.mol2
    docked_ligands_file=${sumDir}/docked_ligands.mol2
    printf "" > $docked_proteins_file
    printf "" > $docked_ligands_file
    for i in `seq 1 $times`
    do
        scoresLine=`sed -n "${i} p" ${rankingFile}.txt`
        oldLigand=${sumDir}_${i}/docked_ligands.mol2
        oldProtein=${sumDir}_${i}/docked_proteins.mol2

        number=`printf "%02d" $i`
        newLigand=${sumDir}/${ligName}_entry_00001_conf_${number}.mol2
        newProtein=${sumDir}/${ligName}_entry_00001_conf_${number}_protein.mol2

        sed "2 s/.*/${ligName}_entry_00001_conf_${number}/" $oldLigand > $newLigand
        sed "2 s/.*/${ligName}_entry_00001_conf_${number}_protein/" $oldProtein >
        $newProtein
        sed "2 s/.*/${ligName}_entry_00001_conf_${number}/" $oldLigand >>
        $docked_ligands_file
        sed "2 s/.*/${ligName}_entry_00001_conf_${number}_protein/" $oldProtein >>
        $docked_proteins_file
        echo "${ligName}_entry_00001_conf_${number},${scoresLine#*,}" >>
        ${rankingFile}.csv
    done
    rm ${rankingFile}.txt
fi

```

```

minCount=$times
# Find actual number of dockings performed (only useful for clustered docking)
number=`grep "@<TRIPOS>MOLECULE" ${sumDir}/docked_ligands.mol2 | wc -l`
if [ $number -lt $minCount ] ; then
    minCount=$number
fi

ligandAtomFile=${sumDir}/ligandAtoms.frag
sed -n '/@<TRIPOS>ATOM/,/@<TRIPOS>BOND/p' $ligandFile > $ligandAtomFile
fixPocketFile=${sumDir}/fixPocket.frag
sed -n '/@<TRIPOS>ATOM/,/@<TRIPOS>BOND/p' ${sumDir}/protein_bindingsite_fixed.mol2 >
$fixPocketFile

# Inform user about what will happen, log setup information
echo "$USER started evaPLANTS for $identifier and the requested analyses in $anaFile."
echo "$USER started evaPLANTS for $identifier and the requested analyses in $anaFile." >
$logFile

echo "Will try to analyze $minCount of requested $times docking poses."
echo "Will try to analyze $minCount of requested $times docking poses." >> $logFile

# Collect requested data for all poses
printf "\n${PLANTSexec}: evaluating docking results...\n"
printf "\n${PLANTSexec}: evaluating docking results...\n" >> $logFile

$splitScript ${sumDir}/docked_ligands.mol2 >> $logFile
$splitScript ${sumDir}/docked_proteins.mol2 >> $logFile

# Collect distance files - needs to be cleared here because the names are the same anyway
distanceFiles=( )

# Parse analysis file
printf "Parsing analysis requests from $anaFile and measuring..."
printf "Parsing analysis requests from $anaFile and measuring..." >> $logFile
NOL=`wc -l < $anaFile`
for lineNo in `seq 1 $NOL`
do
    lineText=`sed -n "$lineNo p" $anaFile`
    # ignore comment lines
    if [ `echo $lineText | grep "^#" | wc -l` -eq 1 ] ; then
        continue
    fi

    NOW=`echo $lineText | wc -w`
    mode="front"
    ligAtoms=( )
    recAtoms=( )
    # If center is indicated, use the mean position of the subsequent atoms for the
evaluation.
    # This is useful for placement of the ligand versus aromats, for example.
    center=0
    # If min is indicated, use only the minimum distance of the previously given ligand
atoms
    # for the evaluation.
    min=0
    for wordNo in `seq 1 $NOW`
    do
        currentWord=`echo $lineText | awk -v w=$wordNo '{print $w}'`
        if [ $mode == "front" ] && [ $currentWord != "FLEX" ] && [ $currentWord !=
"RIGID" ] && [ $currentWord != "MIN" ] ; then
            ligAtoms=( ${ligAtoms[*]} $currentWord )
        elif [ $currentWord == "FLEX" ] ; then
            mode="flex"
        elif [ $currentWord == "RIGID" ] ; then
            mode="rigid"
        elif [ $mode != "front" ] && [ $currentWord != "CENTER" ] && [ $currentWord !=
"MIN" ] ; then
            recAtoms=( ${recAtoms[*]} $currentWord )
        elif [ $currentWord == "CENTER" ] ; then
            center=1
        elif [ $currentWord == "MIN" ] ; then
            min=$lineNo
        else
            echo "Possible syntax error detected in $anaFile - info:"
            echo "lig : ${ligAtoms[*]}"
            echo "rec : ${recAtoms[*]}"
        fi
    done
done

```

```

        echo "center: $center"
        echo "ABORT!"
        exit 1
    fi
done

# Collect distance data from correct files (consider flexible versus rigid!)
cd ${sumDir}
poseFiles=( `find . -type f -regex '.*_entry_00001_conf_[0-9]*.mol2` )
pocketFiles=( `find . -type f -regex '.*_entry_00001_conf_[0-9]*_protein.mol2` )

tail -${minCount} ${sumDir}/ranking.csv | sed 's/,/\t/g' | awk '{print $1"\t"$2}' >
scoreList.txt

# For every docking pose, do the analysis
# if only minimum distances are to be put out
if [ $min -gt 0 ] ; then
    for recAtom in ${recAtoms[*]}
    do
        outFile=lig_min_${lineNo}_rec_${recAtom}_dist.txt
        distanceFiles=( ${distanceFiles[*]} $outFile )
        printf "" > $outFile
        for ligFile in `awk '{print $1}' scoreList.txt`
        do
            miniDist="20.0"
            for ligAtom in ${ligAtoms[*]}
            do
                if [ $mode == "rigid" ] ; then
                    currentDist=`$distanceScript al=$ligAtom
m1=${ligFile}.mol2 a2=$recAtom m2=$fixPocketFile`
                elif [ $mode == "flex" ] ; then
                    currentDist=`$distanceScript al=$ligAtom
m1=${ligFile}.mol2 a2=$recAtom m2=${curDir}/${ligFile}_protein.mol2`
                fi
                if [ `echo "$currentDist < $miniDist" | bc -q` -eq 1 ] ;
then
                    miniDist=$currentDist
                fi
            done
            echo "$miniDist" >> $outFile
        done
    done
else
    for ligAtom in ${ligAtoms[*]}
    do
        if [ $center -eq 1 ] ; then
            outFile=lig_${ligAtom}_center_${lineNo}_dist.txt
            distanceFiles=( ${distanceFiles[*]} $outFile )
            printf "" > $outFile
            for ligFile in `awk '{print $1}' scoreList.txt`
            do
                if [ $mode == "rigid" ] ; then
                    $atomDistAvgPosScript al=$ligAtom
m1=${ligFile}.mol2 m2=$fixPocketFile ${recAtoms[*]} >> $outFile
                elif [ $mode == "flex" ] ; then
                    $atomDistAvgPosScript al=$ligAtom
m1=${ligFile}.mol2 m2=${ligFile}_protein.mol2 ${recAtoms[*]} >> $outFile
                fi
            done
        else
            for recAtom in ${recAtoms[*]}
            do
                outFile=lig_${ligAtom}_rec_${recAtom}_dist.txt
                distanceFiles=( ${distanceFiles[*]} $outFile )
                printf "" > $outFile
                for ligFile in `awk '{print $1}' scoreList.txt`
                do
                    if [ $mode == "rigid" ] ; then
                        $distanceScript al=$ligAtom
m1=${ligFile}.mol2 a2=$recAtom m2=$fixPocketFile >> $outFile
                    elif [ $mode == "flex" ] ; then
                        $distanceScript al=$ligAtom
m1=${ligFile}.mol2 a2=$recAtom m2=${ligFile}_protein.mol2 >> $outFile
                    fi
                done
            done
        fi
    done
done

```

```

        fi
done

echo " - done."
echo " - done." >> $logFile

distanceFiles=( `echo "\${distanceFiles[*]}" | sed 's/ /", "/g'` )

printf "Creating diagrams using R..."
printf "Creating diagrams using R..." >> $logFile

# Create plots about the collected data
echo "# Script to plot interactions from ID $identifier by $anaFile (as of `date`)"

# Function for plotting score-colored distances as pdf and png
plotInteraction <- function (scores, distances, diagTitle, filename, scoreColors) {

    confIntervals <- c(t.test(distances, mu=mean(distances))\$conf.int[1], t.test(distances,
mu=mean(distances))\$conf.int[2])
    confMin <- min(confIntervals)
    confMax <- max(confIntervals)
    meanVal <- mean(distances)
    yBarriers <- c(min(0, distances, confMin), max(0, distances, confMax))

    pdf(paste(filename, ".pdf", sep="\ "))
    barplot(distances, space = 0, ylab = expression(paste("\"Distance (\", 10^-10,\" m)\")),
col=scoreColors, ylim = yBarriers)
    # Add in dotted lines for confidence intervals
    lines(c(0, $minCount), rep(confMin, 2), lwd=2, lty="dotted")
    lines(c(0, $minCount), rep(meanVal, 2), lwd=2)
    lines(c(0, $minCount), rep(confMax,2), lwd=2, lty="dotted")
    title(main=paste(diagTitle, \"(\${identifier})\")
    dev.off()

    png(paste(filename, ".png", sep="\ "))
    barplot(distances, space = 0, ylab = expression(paste("\"Distance (\", 10^-10,\" m)\")),
col=scoreColors, ylim = yBarriers)
    # Add in dotted lines for confidence intervals
    lines(c(0, $minCount), rep(confMin, 2), lwd=2, lty="dotted")
    lines(c(0, $minCount), rep(meanVal, 2), lwd=2)
    lines(c(0, $minCount), rep(confMax,2), lwd=2, lty="dotted")
    title(main=paste(diagTitle, \"(\${identifier})\")
    dev.off()
}

# Handle the docking scores
scores <- read.table(\"${sumDir}/scoreList.txt\")[2]
confIntervals <- c(t.test(scores, mu=mean(scores))\$conf.int[1], t.test(scores,
mu=mean(scores))\$conf.int[2])
confMin <- min(confIntervals)
confMax <- max(confIntervals)
meanVal <- mean(scores[, 1])
yBarriers <- c(min(0, scores[, 1], confMin), max(0, scores[, 1], confMax))

# Plot the scores into pdf
#pdf('${sumDir}/scores.pdf', width=min(10, $minCount), height=6)
pdf('${sumDir}/scores.pdf')
scoreColors <- (scores[, 1] - min(scores[, 1])) / max (scores[, 1] - min(scores[, 1]))
scoreColors <- rgb(scoreColors, max(scoreColors)-scoreColors, 0,
maxColorValue=max(scoreColors))
barplot(scores[, 1], space = 0, xlab = \"Docking poses\", ylab = \"Docking score (arbitrary
units)\", col=scoreColors, ylim = yBarriers)
title(main=\"Scores using $PLANTSexec ($identifier)\")
# Add in dotted lines for confidence intervals
lines(c(0, $minCount), rep(confMin, 2), lwd=2, lty="dotted")
lines(c(0, $minCount), rep(meanVal, 2), lwd=2)
lines(c(0, $minCount), rep(confMax,2), lwd=2, lty="dotted")
dev.off()

# Plot the scores into png
png('${sumDir}/scores.png')
scoreColors <- (scores[, 1] - min(scores[, 1])) / max (scores[, 1] - min(scores[, 1]))
scoreColors <- rgb(scoreColors, max(scoreColors)-scoreColors, 0,
maxColorValue=max(scoreColors))
barplot(scores[,1], space = 0, xlab = \"Docking poses\", ylab = \"Docking score (arbitrary
units)\", col=scoreColors, , ylim = yBarriers)

```



```

# Get common number of docking poses from transformer.log file
nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
for no in ${nos[*]}
do
    if [ $no -lt $loggedNumber ] ; then
        loggedNumber=$no
    fi
done

if [ ! $stopPoses ] ; then
    topPoses=$loggedNumber
else
    if [ $stopPoses -lt 1 ] ; then
        echo "At least one pose needs to be evaluated overall - abort"
        exit 1
    elif [ $stopPoses -gt $loggedNumber ] ; then
        topPoses=$loggedNumber
    fi
fi

if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi

cd $workDir

# Transformations for each varied parameter
pdfFolder=${workDir}/pdf
pngFolder=${workDir}/png
aroSteps=( `sed -n '2 p' $transformerLog | sed 's/.*interaction://g' | sort` )
metSteps=( `sed -n '3 p' $transformerLog | sed 's/.*interaction://g' | sort` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

scoreFile=scoreList.txt
anaFileSuffix=_dist.txt

# Inform user about what will happen, log setup information
echo "$USER started metaPLANTS for identifier $identifier on `date`."
echo "$USER started metaPLANTS for identifier $identifier on `date`." > $logFile

printf "Analyzing results in ${#folders[*]} folders (looking for top $stopPoses poses each)
..."
printf "Analyzing results in ${#folders[*]} folders (looking for top $stopPoses poses each)
..." > $logFile

# Create files to write summary data to
scoreSuperFile=${workDir}/scores_sum.txt
printf "" > $scoreSuperFile
# Assume same distances logged everywhere, take them from first instance
distAnaFiles=( `find ${folders[0]} -name "**${anaFileSuffix}" | sed 's/.*\///g` )
lastIndex=`echo "${#distAnaFiles[*]} - 1" | bc -q`
distSuperFiles=( `echo "${distAnaFiles[*]}" | sed 's/_dist/_sum/g` )

# Extract scores
for folder in ${folders[*]}
do
    cd $folder
    printf "\t$version\n" >> $logFile
    version=`echo "$folder" | sed 's/docked_//`
    echo "$version" > $scoreFile.temp
    awk '{print $2}' $scoreFile >> $scoreFile.temp
    paste $scoreFile.temp $scoreSuperFile > $scoreSuperFile.temp
    mv $scoreSuperFile.temp $scoreSuperFile
    rm $scoreFile.temp
    cd ..
done

```

```

# Extract distances
for i in `seq 0 $lastIndex`
do
    superFile=${workDir}/${distSuperFiles[i]}
    printf "" > $superFile

    for folder in ${folders[*]}
    do
        cd $folder
        version=`echo "$folder" | sed 's/docked_//'\`
        echo "$version" > ${superFile}.temp
        cat ${distAnaFiles[i]} >> ${superFile}.temp
        paste ${superFile}.temp ${superFile} > ${superFile}.temptemp
        mv ${superFile}.temptemp ${superFile}
        rm ${superFile}.temp
        cd ..
    done
done

echo " - done."
echo " - done." >> $logFile

# Do the analysis using R and produce some plots for visualization
printf "Creating diagrams using R..."
printf "Creating diagrams using R..." >> $logFile

# make them an R-compatible enumeration
versions=( `echo "\"${folders[*]}\""" | sed 's/docked_//g; s/ /", "/g'\` )
distanceFiles=( `echo "\"${distSuperFiles[*]}\""" | sed 's/ /", "/g'\` )
aroSteps=( `echo "\"${aroSteps[*]}\""" | sed 's/ /", "/g'\` )
metSteps=( `echo "\"${metSteps[*]}\""" | sed 's/ /", "/g'\` )
mkdir -p $pdfFolder
mkdir -p $pngFolder

# Create plots about the collected data
echo "# Script to plot interactions from ID $identifier and do some statistical analyses (as
of `date`)"
topPoses <- $topPoses

NOaroSteps <- ${#aroSteps[*]}
NometSteps <- ${#metSteps[*]}

# Boxplot scores
scores <- read.table("${scoreSuperFile}", header = TRUE, fill = TRUE)[1:topPoses, ]
if (NOaroSteps > 1 || NometSteps > 1) colnames(scores) <- sub("\"_.\", \"_-\", sub(\"X.\", \"-
\", colnames(scores)))
scores <- as.matrix(cbind(scores))

pdf(paste("${workDir}/scores_sum_top", topPoses, ".pdf", sep="\`")
boxplot(scores ~ col(scores), axes = FALSE)
axis(1, at = 1:length(colnames(scores)), labels = colnames(scores), cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (i in 1:length(colnames(scores))) lines(c(-0.1, 0.1) + i, rep(colMeans(scores)[i], 2), lwd
= 2, col = "\"blue\"")
title(main = paste("\"Docking scores\", \"($identifier)\", ylab = "\"Docking score (arbitrary
units)\")
title(xlab = "\"Modified PLANTS potentials\", line=4.0)
box()
dev.off()

png(paste("${workDir}/scores_sum_top", topPoses, ".png", sep="\`")
boxplot(scores ~ col(scores), axes = FALSE)
axis(1, at = 1:length(colnames(scores)), labels = colnames(scores), cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (i in 1:length(colnames(scores))) lines(c(-0.1, 0.1) + i, rep(colMeans(scores)[i], 2), lwd
= 2, col = "\"blue\"")
title(main = paste("\"Docking scores\", \"($identifier)\", ylab = "\"Docking score (arbitrary
units)\")
title(xlab = "\"Modified PLANTS potentials\", line=4.0)
box()
dev.off()

# Boxplot score changes in one parameter next to each other
if (NOaroSteps > 1 || NometSteps > 1) {
    for (i in 1:NOaroSteps) {
        aroParam <- unlist(strsplit(colnames(scores)[i*NometSteps], "\"_\`"))[1]

```



```

pdf(paste("\${pdfFolder}/scores_aro_", aroParam, "_top", topPoses, ".pdf", sep="\\"))
boxplot(scores[, ((i-1)*NometSteps+1):(i*NometSteps)] ~ col(scores)[, ((i-
1)*NometSteps+1):(i*NometSteps)], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)]), labels =
colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)], cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (j in 1:length(colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)])) lines(c(-0.1,
0.1) + j, rep(colMeans(scores)[((i-1)*NometSteps+1):(i*NometSteps)][j], 2), lwd = 2, col =
"blue")
title(main = paste("Scores with C.cat-aro: ", aroParam, "(${identfier})", ylab =
"Docking score (arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()

png(paste("\${pngFolder}/scores_aro_", aroParam, "_top", topPoses, ".png", sep="\\"))
boxplot(scores[, ((i-1)*NometSteps+1):(i*NometSteps)] ~ col(scores)[, ((i-
1)*NometSteps+1):(i*NometSteps)], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)]), labels =
colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)], cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (j in 1:length(colnames(scores)[((i-1)*NometSteps+1):(i*NometSteps)])) lines(c(-0.1,
0.1) + j, rep(colMeans(scores)[((i-1)*NometSteps+1):(i*NometSteps)][j], 2), lwd = 2, col =
"blue")
title(main = paste("Scores with C.cat-aro: ", aroParam, "(${identfier})", ylab =
"Docking score (arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()
}

# Boxplot score changes in the other parameter next to each other
for (i in 1:NometSteps) {
metParam <- unlist(strsplit(colnames(scores), "_\")[2*(1:NometSteps)][i])

pdf(paste("\${pdfFolder}/scores_met_", metParam, "_top", topPoses, ".pdf", sep="\\"))
boxplot(scores[, i+(0:(NOaroSteps-1))*NometSteps] ~ col(scores)[, i+(0:(NOaroSteps-
1))*NometSteps], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps]), labels =
colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps], cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (j in 1:length(colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps])) lines(c(-0.1, 0.1)
+ j, rep(colMeans(scores)[i+(0:(NOaroSteps-1))*NometSteps][j], 2), lwd = 2, col = "blue")
title(main = paste("Scores with C.cat-MetS: ", metParam, "(${identfier})", ylab =
"Docking score (arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()

png(paste("\${pngFolder}/scores_met_", metParam, "_top", topPoses, ".png", sep="\\"))
boxplot(scores[, i+(0:(NOaroSteps-1))*NometSteps] ~ col(scores)[, i+(0:(NOaroSteps-
1))*NometSteps], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps]), labels =
colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps], cex.axis = 0.7, las = 2)
axis(2, cex.axis=0.7, las=1)
for (j in 1:length(colnames(scores)[i+(0:(NOaroSteps-1))*NometSteps])) lines(c(-0.1, 0.1)
+ j, rep(colMeans(scores)[i+(0:(NOaroSteps-1))*NometSteps][j], 2), lwd = 2, col = "blue")
title(main = paste("Scores with C.cat-MetS: ", metParam, "(${identfier})", ylab =
"Docking score (arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()
}
} else {
onlyParam <- c(${versions[*]})

pdf(paste("\${pdfFolder}/scores_", onlyParam, "_top", topPoses, ".pdf", sep="\\"))
boxplot(scores[, 1] ~ col(scores)[, 1], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[1]), labels = colnames(scores)[1], cex.axis = 0.7,
las = 2)
axis(2, cex.axis=0.7, las=1)
lines(c(0.9, 1.1), rep(colMeans(scores)[1][1], 2), lwd = 2, col = "blue")
title(main = paste("Scores with ", onlyParam, "(${identfier})", ylab = "Docking score
(arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()
}

```

```

png(paste("${pngFolder}/scores_", onlyParam, "_top", topPoses, ".png", sep="\n"))
boxplot(scores[, 1] ~ col(scores)[, 1], axes = FALSE)
axis(1, at = 1:length(colnames(scores)[1]), labels = colnames(scores)[1], cex.axis = 0.7,
las = 2)
axis(2, cex.axis=0.7, las=1)
lines(c(0.9, 1.1), rep(colMeans(scores)[1][1], 2), lwd = 2, col = "blue")
title(main = paste("Scores with ", onlyParam, "($identifier)"), ylab = "Docking score
(arbitrary units)")
title(xlab = "Modified PLANTS potentials", line=4.0)
box()
dev.off()
}

# Some descriptive statistics (in case you need numbers)
sink(paste("${workDir}/Rstats_top", topPoses, ".out", sep = "\n"))
for (i in 1:dim(scores)[2]){
  cat("\n\nStatistical data for scores with potential ", colnames(scores)[i], "\n\n")
  cat("Number of observations:", boxplot.stats(scores[, i])$n, "\n\n")
  cat("Most extreme data interval within +/- 1.5 IQRs from the middle 50:",
boxplot.stats(scores[, i])$stats[1], boxplot.stats(scores[, i])$stats[5], "\n\n")
  cat("25th percentile (1st quartile):", boxplot.stats(scores[, i])$stats[2], "\n\n")
  cat("50th percentile (median):", boxplot.stats(scores[, i])$stats[3], "\n\n")
  cat("75th percentile (3rd quartile):", boxplot.stats(scores[, i])$stats[4], "\n\n")
  cat("Notch interval:", boxplot.stats(scores[, i])$conf, "\n\n")
  cat("Number of outliers (more than +/- 1.5 IQRs from the middle 50):",
length(boxplot.stats(scores[, i])$out), "\n\n")
  cat("IQR = interquartile range (Q3 - Q1), in this case:", abs(boxplot.stats(scores[,
i])$stats[4] - boxplot.stats(scores[, i])$stats[2]), "\n\n")
  cat("Mean: ", mean(scores[, i]), "\n\n")
  cat("95% confidence interval for measured mean:", t.test(scores[, i], mu = mean(scores[,
i]))$conf.int[1:2], "\n\n")
  cat("\n\n\n")
}
sink()

# Boxplot distances
for (interactionFile in c(${distanceFiles[*]})) {
  distFile <- paste("${workDir}/", interactionFile, sep = "\n")
  interaction <- strsplit(interactionFile, "_sum.txt")
  distances <- read.table(distFile, header = TRUE, fill = TRUE)[1:topPoses, ]
  if (NOaroSteps > 1 || NOmetSteps > 1) colnames(distances) <- sub("_.", "_-", sub("X.",
"-", colnames(distances)))
  saveFile <- paste(unlist(strsplit(interactionFile, ".txt"))[1], "_top", topPoses,
sep="\n")
  distances <- as.matrix(cbind(distances))

  # Boxplot distance changes overview for different potentials
  pdf(paste("${workDir}/", saveFile, ".pdf", sep="\n"))
  boxplot(distances ~ col(distances), axes = FALSE)
  axis(1, at = 1:length(colnames(distances)), labels = colnames(distances), cex.axis = 0.7,
las = 2)
  axis(2, cex.axis = 0.7, las = 1)
  for (j in 1:length(colnames(distances))) lines(c(-0.1, 0.1) + j, rep(colMeans(distances)[j],
2), lwd = 2, col = "blue")
  title(main = paste(interaction, "($identifier)"), ylab = expression(paste("Distance
(Å...)")))
  title(xlab = "Modified PLANTS potentials", line=4.0)
  box()
  dev.off()

  png(paste("${workDir}/", saveFile, ".png", sep="\n"))
  boxplot(distances ~ col(distances), axes = FALSE)
  axis(1, at = 1:length(colnames(distances)), labels = colnames(distances), cex.axis = 0.7,
las = 2)
  axis(2, cex.axis=0.7, las=1)
  for (j in 1:length(colnames(distances))) lines(c(-0.1, 0.1) + j, rep(colMeans(distances)[j],
2), lwd = 2, col = "blue")
  title(main = paste(interaction, "($identifier)"), ylab = expression(paste("Distance
(Å...)")))
  title(xlab = "Modified PLANTS potentials", line=4.0)
  box()
  dev.off()

  # Boxplot distance changes in one parameter next to each other

```

```

if (NOaroSteps > 1 || NOMETSteps > 1) {
  for (i in 1:NOaroSteps) {
    aroParam <- unlist(strsplit(colnames(distances)[i*NOMETSteps], "\\_"))[1]

    pdf(paste("${pdfFolder}/dist_\\", interaction, "\\_aro_\\", aroParam, "\\_top\\", topPoses,
    \".pdf\\", sep=\\\"\\\"))
    boxplot(distances[,((i-1)*NOMETSteps+1):(i*NOMETSteps)] ~ col(distances)[,((i-
    1)*NOMETSteps+1):(i*NOMETSteps)], axes = FALSE)
    axis(1, at = 1:length(colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)]), labels
    = colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)], cex.axis = 0.7, las = 2)
    axis(2, cex.axis = 0.7, las = 1)
    for (j in 1:length(colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)])) lines(c(-
    0.1, 0.1) + j, rep(colMeans(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)][j], 2), lwd = 2,
    col = \\\"blue\\\")
    title(main = paste(interaction, \\\"with C.cat-aro: \\\", aroParam, \\\"($identifier)\\\"), ylab
    = expression(paste(\\\"Distance (Å...)\\\")))
    title(xlab = \\\"Modified PLANTS potentials\\\", line=4.0)
    box()
    dev.off()

    png(paste("${pngFolder}/dist_\\", interaction, "\\_aro_\\", aroParam, "\\_top\\", topPoses,
    \".png\\", sep=\\\"\\\"))
    boxplot(distances[,((i-1)*NOMETSteps+1):(i*NOMETSteps)] ~ col(distances)[,((i-
    1)*NOMETSteps+1):(i*NOMETSteps)], axes = FALSE)
    axis(1, at = 1:length(colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)]), labels
    = colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)], cex.axis = 0.7, las = 2)
    axis(2, cex.axis = 0.7, las = 1)
    for (j in 1:length(colnames(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)])) lines(c(-
    0.1, 0.1) + j, rep(colMeans(distances)[((i-1)*NOMETSteps+1):(i*NOMETSteps)][j], 2), lwd = 2,
    col = \\\"blue\\\")
    title(main = paste(interaction, \\\"with C.cat-aro: \\\", aroParam, \\\"($identifier)\\\"), ylab
    = expression(paste(\\\"Distance (Å...)\\\")))
    title(xlab = \\\"Modified PLANTS potentials\\\", line=4.0)
    box()
    dev.off()
  }

  # Boxplot distance changes in the other parameter next to each other
  for (i in 1:NOMETSteps) {
    metParam <- unlist(strsplit(colnames(distances), "\\_"))[2*(1:NOMETSteps)][i]

    pdf(paste("${pdfFolder}/dist_\\", interaction, "\\_met_\\", metParam, "\\_top\\", topPoses,
    \".pdf\\", sep=\\\"\\\"))
    boxplot(distances[,i+(0:(NOaroSteps-1))*NOMETSteps] ~ col(distances)[,i+(0:(NOaroSteps-
    1))*NOMETSteps], axes = FALSE)
    axis(1, at = 1:length(colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps]), labels =
    colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps], cex.axis = 0.7, las = 2)
    axis(2, cex.axis = 0.7, las = 1)
    for (j in 1:length(colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps])) lines(c(-0.1,
    0.1) + j, rep(colMeans(distances)[i+(0:(NOaroSteps-1))*NOMETSteps][j], 2), lwd = 2, col =
    \\\"blue\\\")
    title(main = paste(interaction, \\\"with C.cat-MetS: \\\", metParam, \\\"($identifier)\\\"),
    ylab = expression(paste(\\\"Distance (Å...)\\\")))
    title(xlab = \\\"Modified PLANTS potentials\\\", line=4.0)
    box()
    dev.off()

    png(paste("${pngFolder}/dist_\\", interaction, "\\_met_\\", metParam, "\\_top\\", topPoses,
    \".png\\", sep=\\\"\\\"))
    boxplot(distances[,i+(0:(NOaroSteps-1))*NOMETSteps] ~ col(distances)[,i+(0:(NOaroSteps-
    1))*NOMETSteps], axes = FALSE)
    axis(1, at = 1:length(colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps]), labels =
    colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps], cex.axis = 0.7, las = 2)
    axis(2, cex.axis = 0.7, las = 1)
    for (j in 1:length(colnames(distances)[i+(0:(NOaroSteps-1))*NOMETSteps])) lines(c(-0.1,
    0.1) + j, rep(colMeans(distances)[i+(0:(NOaroSteps-1))*NOMETSteps][j], 2), lwd = 2, col =
    \\\"blue\\\")
    title(main = paste(interaction, \\\"with C.cat-MetS: \\\", metParam, \\\"($identifier)\\\"),
    ylab = expression(paste(\\\"Distance (Å...)\\\")))
    title(xlab = \\\"Modified PLANTS potentials\\\", line=4.0)
    box()
    dev.off()
  }
} else {
  onlyParam <- c(${versions[*]})

  pdf(paste("${pdfFolder}/dist_\\", interaction, "\\_\\", onlyParam, "\\_top\\", topPoses,
  \".pdf\\", sep=\\\"\\\"))

```



```

        id)      identifier=`echo "$1" | sed 's/.*/'`;
        tops)   topPoses=`echo "$1" | sed 's/.*/'`;
        *)      echo "Could not interpret $1 as an entity index - abort."
                exit 1;;
    esac
    shift
done

if [ ! $identifier ] ; then
    echo "No identifier given - abort!"
    exit 1
fi

# Folder and file variables
mainDir=/raid/home/rklein/DOCK/plants/transform
workDir=$mainDir/transform_$identifier
logFile=$workDir/corDistScores.log
evaLogFile=$workDir/evaPLANTS.log
transformerLog=${workDir}/transformer.log

if [ ! -f $evaLogFile ] ; then
    echo "Evaluation log file $evaLogFile was not found - abort!"
    exit 1
fi

# Just a hint from the last of the docking runs
loggedNumber=`sed -n '2 p' $evaLogFile | awk '{printf $5}'`

# Get common number of docking poses from transformer.log file
nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
for no in ${nos[*]}
do
    if [ $no -lt $loggedNumber ] ; then
        loggedNumber=$no
    fi
done

if [ ! $topPoses ] ; then
    topPoses=$loggedNumber
else
    if [ $topPoses -lt 1 ] ; then
        echo "At least one pose needs to be evaluated overall - abort!"
        exit 1
    elif [ $topPoses -gt $loggedNumber ] ; then
        topPoses=$loggedNumber
    fi
fi

if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi

# Inform user about what will happen, log setup information
echo "$USER started corDistScores for identifier $identifier on `date`."
echo "$USER started corDistScores for identifier $identifier on `date`." > $logFile

cd $workDir

# Transformations for each varied parameter
transformerLog=${workDir}/transformer.log
pdfFolder=${workDir}/pdf
pngFolder=${workDir}/png
aroSteps=( `sed -n '2 p' $transformerLog | sed 's/.*/interaction://g' | sort` )
metSteps=( `sed -n '3 p' $transformerLog | sed 's/.*/interaction://g' | sort` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

anaFileSuffix=_dist.txt

# Files from meta analysis

```

```

scoreSuperFile=${workDir}/scores_sum.txt
# Assume same distances logged everywhere, take them from first instance
distAnaFiles=( `find ${folders[0]} -name "*${anaFileSuffix}" | sed 's/.*/g/'` )

# make R-compatible enumerations
distanceFiles=( `echo "\"${distAnaFiles[*]}\""" | sed 's/_dist/_sum/g;s/ /, "/g'` )
versions=( `echo "\"${folders[*]}\""" | sed 's/docked_/g; s/ /, "/g'` )
aroSteps=( `echo "\"${aroSteps[*]}\""" | sed 's/ /, "/g'` )
metSteps=( `echo "\"${metSteps[*]}\""" | sed 's/ /, "/g'` )

# Do the analysis using R and produce some plots for visualization
printf "Creating correlation diagrams using R..."
printf "Creating correlation diagrams using R..." >> $logFile

# Create plots about the collected data
echo "# Script to plot correlations between docking scores and distances from ID $identifier
(as of `date`)"
topPoses <- $topPoses
NOaroSteps <- ${#aroSteps[*]}
NOMETSteps <- ${#metSteps[*]}

# Read in scores
scores <- read.table("${scoreSuperFile}", header = TRUE, fill = TRUE)[1:topPoses, ]
if (NOaroSteps > 1 || NOMETSteps > 1) colnames(scores) <- c(${versions[*]})
scores <- as.matrix(cbind(scores))

# For each considered interaction, for each potential, plot scores vs. distances
for (interactionFile in c(${distanceFiles[*]})) {
  distFile <- paste("${workDir}/", interactionFile, sep = "\")
  currentInteraction <- strsplit(interactionFile, "_sum.txt")
  distances <- read.table(distFile, header = TRUE, fill = TRUE)[1:topPoses, ]
  if (NOaroSteps > 1 || NOMETSteps > 1) colnames(distances) <- c(${versions[*]})
  distances <- as.matrix(cbind(distances))

  # Scatterplot correlations between scores and distances for one parameter series
  if (NOMETSteps > 1) for (i in 1:NOaroSteps) {
    constInteraction <- unlist(strsplit(colnames(scores)[i*NOMETSteps], "\_"))[1]
    saveFile <- paste("cor_", currentInteraction, "_aro_", constInteraction, "_top",
topPoses, sep="\")
    pearsonCoeffs <- diag(cor(scores[, ((i-1)*NOMETSteps+1):(i*NOMETSteps)], distances[, ((i-
1)*NOMETSteps+1):(i*NOMETSteps)]))
    # if (length(pearsonCoeffs) > 1) pearsonCoeffs <- diag(pearsonCoeffs)
    colNum <- 3
    if (NOMETSteps == 1 || NOMETSteps == 2) colNum <- NOMETSteps
    if (NOMETSteps == 4) colNum <- 2

    pdf(paste("${pdfFolder}/", saveFile, ".pdf", sep="\"))
    layout(matrix(c(1,2), byrow=TRUE), heights=c(5, 1))
    par(mar=c(5,4,4,0)+0.1)
    plot(scores[, ((i-1)*NOMETSteps+1):(i*NOMETSteps)], distances[, ((i-
1)*NOMETSteps+1):(i*NOMETSteps)], xlab = "Docking scores (arbitrary units)", ylab =
expression(paste("Distance (Å...)")), main = paste(currentInteraction, "with C.cat-aro:",
constInteraction, "($identifier)", col = rep(rainbow(NOMETSteps), each = topPoses))
    for (j in 1:NOMETSteps) abline(lm(distances[, (i-1)*NOMETSteps+j] ~ scores[, (i-
1)*NOMETSteps+j]), col = rainbow(NOMETSteps)[j])
    par(mar=c(0,0,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))
    legend("topright", paste(colnames(scores)[((i-1)*NOMETSteps+1):(i*NOMETSteps)], "\, r =
", round(pearsonCoeffs, 2), sep = "\"), col = rainbow(NOMETSteps), bg = "white", ncol =
colNum, title = "Depicted potentials", pch = 1)
    dev.off()

    png(paste("${pngFolder}/", saveFile, ".png", sep="\"), width = 1024, height = 768)
    layout(matrix(c(1,2), byrow=TRUE), heights=c(6, 1))
    par(mar=c(5,4.5,4,0)+0.1)
    plot(scores[, ((i-1)*NOMETSteps+1):(i*NOMETSteps)], distances[, ((i-
1)*NOMETSteps+1):(i*NOMETSteps)], xlab = "Docking scores (arbitrary units)", ylab =
expression(paste("Distance (Å...)")), main = paste(currentInteraction, "with C.cat-aro:",
constInteraction, "($identifier)", col = rep(rainbow(NOMETSteps), each = topPoses))
    for (j in 1:NOMETSteps) abline(lm(distances[, (i-1)*NOMETSteps+j] ~ scores[, (i-
1)*NOMETSteps+j]), col = rainbow(NOMETSteps)[j])
    par(mar=c(0,4.5,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))

```

```

    legend("topleft", paste(colnames(scores)[(i-1)*NOmetSteps+1):(i*NOmetSteps)], "\", r =
    "\", round(pearsonCoeffs, 2), sep = "\\\"", col = rainbow(NOmetSteps), bg = \"white\", ncol =
    colNum, title = \"Depicted potentials\", pch = 1)
    dev.off()
  }

  # Scatterplot correlations between scores and distances for the other parameter series
  if (NOaroSteps > 1) for (i in 1:NOmetSteps) {
    constInteraction <- unlist(strsplit(colnames(scores)[i], \"_\")[2])
    saveFile <- paste(\"cor_\", currentInteraction, \"_met_\", constInteraction, \"_top\",
topPoses, sep=\\\")
    pearsonCoeffs <- diag(cor(scores[, (0:(NOaroSteps-1))*NOmetSteps+i], distances[,
(0:(NOaroSteps-1))*NOmetSteps+i]))
    # if (length(pearsonCoeffs) > 1) pearsonCoeffs <- diag(pearsonCoeffs)
    colNum <- 3
    if (NOaroSteps == 1 || NOaroSteps == 2) colNum <- NOaroSteps
    if (NOaroSteps == 4) colNum <- 2

    pdf(paste(\"${pdfFolder}/\", saveFile, \".pdf\", sep=\\\")
    layout(matrix(c(1,2), byrow=TRUE), heights=c(5, 1))
    par(mar=c(5,4,4,0)+0.1)
    plot(scores[, (0:(NOaroSteps-1))*NOmetSteps+i], distances[, (0:(NOaroSteps-
1))*NOmetSteps+i], xlab = \"Docking scores (arbitrary units)\", ylab =
expression(paste(\"Distance (Å...)\")), main = paste(currentInteraction, \"with C.cat-MetS:\",
constInteraction, \"($identifier)\", col = rep(rainbow(NOaroSteps), each = topPoses))
    for (j in 1:NOaroSteps) abline(lm(distances[, i+(j-1)*NOmetSteps] ~ scores[, i+(j-
1)*NOmetSteps]), col = rainbow(NOaroSteps)[j])
    par(mar=c(0,0,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))
    legend(\"topright\", paste(colnames(scores)[(0:(NOaroSteps-1))*NOmetSteps+i], \"\", r = \"\",
round(pearsonCoeffs, 2), sep = "\\\"", col = rainbow(NOaroSteps), bg = \"white\", ncol =
colNum, title = \"Depicted potentials\", pch = 1)
    dev.off()

    png(paste(\"${pngFolder}/\", saveFile, \".png\", sep=\\\"), width = 1024, height = 768)
    layout(matrix(c(1,2), byrow=TRUE), heights=c(6, 1))
    par(mar=c(5,4.5,4,0)+0.1)
    plot(scores[, (0:(NOaroSteps-1))*NOmetSteps+i], distances[, (0:(NOaroSteps-
1))*NOmetSteps+i], xlab = \"Docking scores (arbitrary units)\", ylab =
expression(paste(\"Distance (Å...)\")), main = paste(currentInteraction, \"with C.cat-MetS:\",
constInteraction, \"($identifier)\", col = rep(rainbow(NOaroSteps), each = topPoses))
    for (j in 1:NOaroSteps) abline(lm(distances[, i+(j-1)*NOmetSteps] ~ scores[, i+(j-
1)*NOmetSteps]), col = rainbow(NOaroSteps)[j])
    par(mar=c(0,4.5,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))
    legend(\"topleft\", paste(colnames(scores)[(0:(NOaroSteps-1))*NOmetSteps+i], \"\", r = \"\",
round(pearsonCoeffs, 2), sep = "\\\"", col = rainbow(NOaroSteps), bg = \"white\", ncol =
colNum, title = \"Depicted potentials\", pch = 1)
    dev.off()
  }

  if (NOaroSteps == 1 && NOmetSteps == 1) {
    saveFile <- paste(\"cor_\", currentInteraction, \"_\", ${versions[*]}, \"_top\", topPoses,
sep=\\\")
    pearsonCoeffs <- cor(scores[, 1], distances[, 1])

    pdf(paste(\"${pdfFolder}/\", saveFile, \".pdf\", sep=\\\")
    layout(matrix(c(1,2), byrow=TRUE), heights=c(5, 1))
    par(mar=c(5,4,4,0)+0.1)
    plot(scores[, 1], distances[, 1], xlab = \"Docking scores (arbitrary units)\", ylab =
\"Distance (Å...)\", main = paste(currentInteraction, \"with\", ${versions[*]},
\"($identifier)\", col = \"blue\")
    abline(lm(distances[, 1] ~ scores[, 1]), col = \"blue\")
    par(mar=c(0,0,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))
    legend(\"topright\", paste(${versions[*]}, \"\", r = \"\", round(pearsonCoeffs, 2), sep =
\\\"), col = \"blue\", bg = \"white\", ncol = colNum, title = \"Depicted potentials\", pch =
1)
    dev.off()

    png(paste(\"${pngFolder}/\", saveFile, \".png\", sep=\\\"), width = 1024, height = 768)
    layout(matrix(c(1,2), byrow=TRUE), heights=c(6, 1))
    par(mar=c(5,4,4,0)+0.1)

```

```

    plot(scores[, 1], distances[, 1], xlab = \"Docking scores (arbitrary units)\", ylab =
\"Distance (Å...)\", main = paste(currentInteraction, \"with\", ${versions[*]},
\"(${identfier})\"), col = \"blue\")
    abline(lm(distances[, 1] ~ scores[, 1]), col = \"blue\")
    par(mar=c(0,4.5,0,0)+0.1)
    plot.new()
    plot.window(c(0,1), c(0,1))
    legend(\"topright\", paste(${versions[*]}, \", r = \", round(pearsonCoeffs, 2), sep =
\"\"), col = \"blue\", bg = \"white\", ncol = colNum, title = \"Depicted potentials\", pch =
1)
    dev.off()
}

# Heatmap depiction of correlations for all parameter translations
corCoeffs <- diag(cor(scores, distances))
saveFile <- paste(\"cor_\", currentInteraction, \"_all_top\", topPoses, sep=\"\")

# interpolate colors and extract from complete palette
NOcolors <- 1000
minIndex <- min(round((1+NOcolors)/2 + corCoeffs*((NOcolors+1)/2)-1))
maxIndex <- max(round((1+NOcolors)/2 + corCoeffs*((NOcolors+1)/2)-1))

pdf(paste(\"${workDir}/\", saveFile, \".pdf\", sep=\"\"))
layout(matrix(c(1,2), 1,2 ), widths=c(5, 1))
image(1:NOmetSteps, 1:NOaroSteps, matrix(corCoeffs, NOmetSteps, NOaroSteps),
main=paste(\"Score vs.\", currentInteraction, \"distance (${identfier})\", xlab=\"MetS
potential translation\", ylab=\"Aro potential translation\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
axis(1, at = 1:NOmetSteps, labels = unlist(strsplit(colnames(scores)[1:NOmetSteps],
\"_\")[1:NOmetSteps]*2])
axis(2, at = 1:NOaroSteps, labels =
unlist(strsplit(colnames(scores)[(1:NOaroSteps)*NOmetSteps, \"_\")[1:NOaroSteps]*2-1], las
= 1)

box()
for (i in 1:NOmetSteps) for (j in 1:NOaroSteps) text(i,j, paste(colnames(scores)[i+(j-
1)*(NOmetSteps)], \"\nr =\", round(corCoeffs[i+(j-1)*(NOmetSteps)], 2)), cex = 0.7)
# plot a colorful legend
par(mar=c(5,4,4,0)+0.1)
image(1, 1:100, matrix(1:100, 1, 100), xlab=\", ylab=\"Pearson's r\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE)
axis(2, at = seq(1,100, length.out=9), labels = round(seq(-1,1,length.out=9), 2), las = 1)
dev.off()

png(paste(\"${workDir}/\", saveFile, \".png\", sep=\"\"), width = 1024, height = 768)
layout(matrix(c(1,2), 1,2 ), widths=c(5, 1))
image(1:NOmetSteps, 1:NOaroSteps, matrix(corCoeffs, NOmetSteps, NOaroSteps),
main=paste(\"Correlations: score -\", currentInteraction, \"distance (${identfier})\",
xlab=\"MetS potential translation\", ylab=\"Aro potential translation\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
axis(1, at = 1:NOmetSteps, labels = unlist(strsplit(colnames(scores)[1:NOmetSteps],
\"_\")[1:NOmetSteps]*2])
axis(2, at = 1:NOaroSteps, labels =
unlist(strsplit(colnames(scores)[(1:NOaroSteps)*NOmetSteps, \"_\")[1:NOaroSteps]*2-1], las
= 1)
box()
for (i in 1:NOmetSteps) for (j in 1:NOaroSteps) text(i,j, paste(colnames(scores)[i+(j-
1)*(NOmetSteps)], \"\nr =\", round(corCoeffs[i+(j-1)*(NOmetSteps)], 2)), cex = 1.0)
# plot a colorful legend
par(mar=c(5,4,4,0)+0.1)
image(1, 1:100, matrix(1:100, 1, 100), xlab=\", ylab=\"Pearson's r\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE)
axis(2, at = seq(1,100, length.out=9), labels = round(seq(-1,1,length.out=9), 2), las = 1)
dev.off()
} > ${workDir}/corDistScores.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/corDistScores.R >> $logFile

echo \" - done.\"
printf \"\n - done.\n\" >> $logFile
echo \"corDistScores completely done on `date`.\"
echo \"See $logFile for logged information.\"
echo \"corDistScores completely done on `date`.\" >> $logFile

```

A 16 Bash-Shell-Skript violinPLANTS

```
#!/bin/bash
```



```

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

# Files in /docked_ARO_MET
scoreFile=scoreList.txt

# Files in .
scoreSuperFile=${workDir}/scores_sum.txt
distanceSuperFiles=( `find ${workDir}/lig*_sum.txt` )

interactions=( `find ${workDir}/lig*_sum.txt | sed 's/.*\///; s/_sum\.txt//` )
interactions=( `echo "${interactions[*]}" | sed 's/ /", "/g` )

# Translate to R-readable
distanceFiles=( `echo "${distanceSuperFiles[*]}" | sed 's/ /", "/g` )
aroSteps=( `echo "${aroSteps[*]}" | sed 's/ /", "/g` )
metSteps=( `echo "${metSteps[*]}" | sed 's/ /", "/g` )
versions=( `echo "${folders[*]}" | sed 's/docked_/g; s/ /", "/g` )

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile
printf "\tCreating violin plots for score and distance distributions ...\n"
printf "\tCreating violin plots for score and distance distributions ...\n" >> $logFile

# Create violin plots
echo "# Script to produce violin plots from scores and interactions from ID $identifier (as of
`date`)"

topPoses <- $topPoses
interactions <- c(${interactions[*]})
NOaroSteps <- ${#aroSteps[*]}
NometSteps <- ${#metSteps[*]}

# Function to produce a violin plot for a matrix-variable
# Tailored for score and distance overviews in script violinPLANTS
violinSuperPlot <- function (matrixData, xDim, yDim, mainTitle, subLabels, xLabel, yLabel,
fileName, deviceName) {

    xScale <- if (xDim > 3) xDim * 1.1 else 1
    yScale <- if (yDim > 3) yDim * 1.1 else 1

    if (deviceName == "pdf") {
        saveFile <- paste(fileName, ".pdf", sep="")
        if (xDim > yDim) pdf(saveFile, width = 6 * xDim / yDim, height = 6)
        else pdf(saveFile, width = 6 * xScale, height = 6 * yDim / xDim * yScale)
    } else
    if (deviceName == "png") {
        saveFile <- paste(fileName, ".png", sep="")
        if (xDim > yDim) png(saveFile, width = 720 * xDim / yDim, height = 720 * yScale)
        else png(saveFile, width = 720 * xScale, height = 720 * yDim / xDim * yScale)
    }

    # Have an extra line for the title text
    layout(matrix(c(rep(1,2*xDim), 1+(1:(2 * xDim * yDim))), (1+yDim), 2 * xDim, byrow=TRUE),
height = c(1, rep(5, yDim)))
    par(mar = c(0,0,2.5,0))
    frame()
    mtext(mainTitle, cex = 1.5)

    # Plot the butterfly-like diagrams
    for (i in 1:(xDim * yDim)) {
        # Some statistical descriptors
        firstQuartile <- boxplot.stats(matrixData[, i])$stats[2]
        median <- boxplot.stats(matrixData[, i])$stats[3]
        thirdQuartile <- boxplot.stats(matrixData[, i])$stats[4]
        meanValue <- mean(matrixData[, i])

        # some space around the extreme density values

```

```

    maxValue <- max(density(matrixData[, 1])\$y)
    if (dim(matrixData)[2] > 1) for (j in 2:dim(matrixData)[2]) if (max(density(matrixData[,
j])\$y) > maxValue) maxValue <- max(density(matrixData[, j])\$y)
    minValue <- min(density(matrixData[, 1])\$y)
    if (dim(matrixData)[2] > 1) for (j in 2:dim(matrixData)[2]) if (min(density(matrixData[,
j])\$y) < minValue) minValue <- min(density(matrixData[, j])\$y)
    xAxAdd <- abs(maxValue - minValue) * 0.2

    # Plot density function inversely
    par(mar=c(2, 2, 3, 0))
    plot(-density(matrixData[, i])\$y, density(matrixData[, i])\$x, type="l", xlim = c(-
abs(maxValue) - xAxAdd, 0), ylim = range(density(matrixData)\$x), xlab = "\", ylab = "\",
xaxs = "i", axes = FALSE)
    title(main = subLabels[i])
    title(line = 1, xlab = xLabel)
    title(line = 0, ylab = yLabel)
    box()
    lines(c(-0.2*xAxAdd, 0), c(meanValue, meanValue), lwd = 2, col = "blue")
    lines(c(-0.4*xAxAdd, 0), c(firstQuartile, firstQuartile), lwd = 1, col = "black")
    lines(rep(-0.4*xAxAdd, 2), c(firstQuartile, thirdQuartile), lwd = 1, col = "black")
    lines(c(-0.4*xAxAdd, 0), c(median, median), lwd = 3, col = "black")
    lines(c(-0.4*xAxAdd, 0), c(thirdQuartile, thirdQuartile), lwd = 1, col = "black")

    # plot density function normally
    par(mar=c(2, 0, 3, 1))
    plot(density(matrixData[, i])\$y, density(matrixData[, i])\$x, type="l", xlim = c(0,
abs(maxValue) + xAxAdd), ylim = range(density(matrixData)\$x), xlab = "\", ylab = "\",
xaxs = "i")
    lines(c(0, 0.2*xAxAdd), c(meanValue, meanValue), lwd = 2, col = "blue")
    lines(c(0, 0.4*xAxAdd), c(firstQuartile, firstQuartile), lwd = 1, col = "black")
    lines(rep(0.4*xAxAdd, 2), c(firstQuartile, thirdQuartile), lwd = 1, col = "black")
    lines(c(0, 0.4*xAxAdd), c(median, median), lwd = 3, col = "black")
    lines(c(0, 0.4*xAxAdd), c(thirdQuartile, thirdQuartile), lwd = 1, col = "black")
}

if (deviceName == "pdf" || deviceName == "png") dev.off()
}

# Read in scores
scores <- read.table("\$scoreSuperFile", header = TRUE, fill = TRUE)[1:topPoses, ]
if (NOaroSteps > 1 || NOmetSteps > 1) colnames(scores) <- sub("\_.\", \"_\-", sub("\X.\", \"-
\", colnames(scores))
scores <- as.matrix(cbind(scores))

# Violin-plot scores
violinSuperPlot(scores, max(NOmetSteps, NOaroSteps), min(NOmetSteps, NOaroSteps), "Docking
score densities (\$identifier)\", paste("Potential:\", colnames(scores)), "Data density:\",
"Docking scores (arbitrary units)\", paste("\${workDir}/scores_violin_top", topPoses,
sep="\\"), "pdf")
violinSuperPlot(scores, max(NOmetSteps, NOaroSteps), min(NOmetSteps, NOaroSteps), "Docking
score densities (\$identifier)\", paste("Potential:\", colnames(scores)), "Data density:\",
"Docking scores (arbitrary units)\", paste("\${workDir}/scores_violin_top", topPoses,
sep="\\"), "png")

# Violin-plot score changes
if (NOaroSteps == 1 && NOmetSteps == 1) {
  # Violin-plot score changes in both parameters
  param <- c(\${versions[*]})
  violinSuperPlot(scores, 1, 1, paste("Scores with\"", param, \"(\$identifier)\",
paste("Potential:\", param), "Data density:\", "Docking scores (arbitrary units)\",
paste("\${pdfFolder}/scores_violin_only_\", param, \"_top\", topPoses, sep="\\"), "pdf")
  violinSuperPlot(scores, 1, 1, paste("Scores with\"", param, \"(\$identifier)\",
paste("Potential:\", param), "Data density:\", "Docking scores (arbitrary units)\",
paste("\${pngFolder}/scores_violin_only_\", param, \"_top\", topPoses, sep="\\"), "png")
} else {
  # Violin-plot score changes in one parameter next to each other
  if (NOaroSteps == 1) {
    aroParam <- unlist(strsplit(colnames(scores)[NOmetSteps], \"_\-\"))[1]
    violinSuperPlot(scores, NOmetSteps, 1, paste("Scores with C.cat-aro: \", aroParam,
\"(\$identifier)\", paste("Potential:\", colnames(scores)[1:NOmetSteps]), "Data density:\",
"Docking scores (arbitrary units)\", paste("\${pdfFolder}/scores_violin_aro_\", aroParam,
\"_top\", topPoses, sep="\\"), "pdf")
    violinSuperPlot(scores, NOmetSteps, 1, paste("Scores with C.cat-aro: \", aroParam,
\"(\$identifier)\", paste("Potential:\", colnames(scores)[1:NOmetSteps]), "Data density:\",
"Docking scores (arbitrary units)\", paste("\${pngFolder}/scores_violin_aro_\", aroParam,
\"_top\", topPoses, sep="\\"), "png")
  } else if (NOmetSteps > 1){

```

```

    for (i in 1:NOaroSteps) {
      aroParam <- unlist(strsplit(colnames(scores)[i*NOmetSteps], "\\_")[1])
      violinSuperPlot(scores[,((i-1)*NOmetSteps+1):(i*NOmetSteps)], NOmetSteps, 1,
        paste("Scores with C.cat-aro: \", aroParam, \"($identifier)\", paste(\"Potential:\",
        colnames(scores)[((i-1)*NOmetSteps+1):(i*NOmetSteps)], \"Data density:\", \"Docking scores
        (arbitrary units)\", paste(\"${pdfFolder}/scores_violin_aro_\", aroParam, \"_top\", topPoses,
        sep=\"\"), \"pdf\")
      violinSuperPlot(scores[,((i-1)*NOmetSteps+1):(i*NOmetSteps)], NOmetSteps, 1,
        paste("Scores with C.cat-aro: \", aroParam, \"($identifier)\", paste(\"Potential:\",
        colnames(scores)[((i-1)*NOmetSteps+1):(i*NOmetSteps)], \"Data density:\", \"Docking scores
        (arbitrary units)\", paste(\"${pngFolder}/scores_violin_aro_\", aroParam, \"_top\", topPoses,
        sep=\"\"), \"png\")
    }
  }
  # Violin-plot score changes in the other parameter next to each other
  if (NOmetSteps == 1) {
    metParam <- unlist(strsplit(colnames(scores), "\\_")[2*(1:NOmetSteps)][1])
    violinSuperPlot(scores, NOaroSteps, 1, paste("Scores with C.cat-MetS: \", metParam,
    \"($identifier)\", paste(\"Potential:\", colnames(scores)[1:NOaroSteps]), \"Data density:\",
    \"Docking scores (arbitrary units)\", paste(\"${pdfFolder}/scores_violin_met_\", metParam,
    \"_top\", topPoses, sep=\"\"), \"pdf\")
    violinSuperPlot(scores, NOaroSteps, 1, paste("Scores with C.cat-MetS: \", metParam,
    \"($identifier)\", paste(\"Potential:\", colnames(scores)[1:NOaroSteps]), \"Data density:\",
    \"Docking scores (arbitrary units)\", paste(\"${pngFolder}/scores_violin_met_\", metParam,
    \"_top\", topPoses, sep=\"\"), \"png\")
  } else if (NOaroSteps > 1) {
    for (i in 1:NOmetSteps) {
      metParam <- unlist(strsplit(colnames(scores), "\\_")[2*(1:NOmetSteps)][i])
      violinSuperPlot(scores[,i+(0:(NOaroSteps-1))*NOmetSteps], NOaroSteps, 1, paste("Scores
      with C.cat-MetS: \", metParam, \"($identifier)\", paste(\"Potential:\",
      colnames(scores)[i+(0:(NOaroSteps-1))*NOmetSteps]), \"Data density:\", \"Docking scores
      (arbitrary units)\", paste(\"${pdfFolder}/scores_violin_met_\", metParam, \"_top\", topPoses,
      sep=\"\"), \"pdf\")
      violinSuperPlot(scores[,i+(0:(NOaroSteps-1))*NOmetSteps], NOaroSteps, 1, paste("Scores
      with C.cat-MetS: \", metParam, \"($identifier)\", paste(\"Potential:\",
      colnames(scores)[i+(0:(NOaroSteps-1))*NOmetSteps]), \"Data density:\", \"Docking scores
      (arbitrary units)\", paste(\"${pngFolder}/scores_violin_met_\", metParam, \"_top\", topPoses,
      sep=\"\"), \"png\")
    }
  }
}

# Violin-plot distances
for (interaction in interactions) {
  # Read in distances for currently considered interaction
  distances <- read.table(paste(\"${workDir}/\", interaction, \"_sum.txt\", sep = \"\"),
  header = TRUE, fill = TRUE)[1:topPoses, ]
  if (NOaroSteps > 1 || NOmetSteps > 1) colnames(distances) <- sub(\"_.\", \"_-\", sub(\"X.\",
  \"-\", colnames(distances)))
  distances <- as.matrix(cbind(distances))

  # Violin-plot distance changes overview for different potentials
  violinSuperPlot(distances, max(NOmetSteps, NOaroSteps), min(NOmetSteps, NOaroSteps),
  paste(interaction, \"($identifier)\", paste(\"Potential:\", colnames(scores)), \"Data
  density:\", \"Distance (Å...)\", paste(\"${workDir}/\", interaction, \"_violin_sum_top\",
  topPoses, sep=\"\"), \"pdf\")
  violinSuperPlot(distances, max(NOmetSteps, NOaroSteps), min(NOmetSteps, NOaroSteps),
  paste(interaction, \"($identifier)\", paste(\"Potential:\", colnames(scores)), \"Data
  density:\", \"Distance (Å...)\", paste(\"${workDir}/\", interaction, \"_violin_sum_top\",
  topPoses, sep=\"\"), \"png\")

  if (NOaroSteps == 1 && NOmetSteps == 1) {
    # Violin-plot distance changes in both parameters
    param <- c(${versions[*]})
    violinSuperPlot(distances, 1, 1, paste(interaction, \"with C.cat-aro: \", param,
    \"($identifier)\", paste(\"Potential:\", param), \"Data density:\", \"Distance (Å...)\",
    paste(\"${pdfFolder}/dist_\", interaction, \"_violin_only_\", param, \"_top\", topPoses,
    sep=\"\"), \"pdf\")
    violinSuperPlot(distances, 1, 1, paste(interaction, \"with C.cat-aro: \", param,
    \"($identifier)\", paste(\"Potential:\", param), \"Data density:\", \"Distance (Å...)\",
    paste(\"${pngFolder}/dist_\", interaction, \"_violin_only_\", param, \"_top\", topPoses,
    sep=\"\"), \"png\")
  } else {
    # Violin-plot distance changes in one parameter next to each other
    if (NOaroSteps == 1) {
      aroParam <- unlist(strsplit(colnames(distances)[NOmetSteps], "\\_")[1])
      violinSuperPlot(distances, NOmetSteps, 1, paste(interaction, \"with C.cat-aro: \",
      aroParam, \"($identifier)\", paste(\"Potential:\", colnames(scores)[1:NOmetSteps]), \"Data

```

```

density:\", \"Distance (Å...)\"\", paste(\"${pdfFolder}/dist_\", interaction, \"_violin_aro_\",
aroParam, \"_top\", topPoses, sep=\"\"), \"pdf\")
    violinSuperPlot(distances, NOmetSteps, 1, paste(interaction, \"with C.cat-aro: \",
aroParam, \"($identfier)\", paste(\"Potential:\", colnames(scores)[1:NOmetSteps]), \"Data
density:\", \"Distance (Å...)\"\", paste(\"${pngFolder}/dist_\", interaction, \"_violin_aro_\",
aroParam, \"_top\", topPoses, sep=\"\"), \"png\")
  } else if (NOmetSteps > 1){
    for (i in 1:NOaroSteps) {
      aroParam <- unlist(strsplit(colnames(distances)[i*NOmetSteps], \"_\")[1]
      violinSuperPlot(distances[, ((i-1)*NOmetSteps+1):(i*NOmetSteps)], NOmetSteps, 1,
paste(interaction, \"with C.cat-aro: \", aroParam, \"($identfier)\", paste(\"Potential:\",
colnames(scores)[((i-1)*NOmetSteps+1):(i*NOmetSteps)], \"Data density:\", \"Distance (Å...)\"\",
paste(\"${pdfFolder}/dist_\", interaction, \"_violin_aro_\", aroParam, \"_top\", topPoses,
sep=\"\"), \"pdf\")
      violinSuperPlot(distances[, ((i-1)*NOmetSteps+1):(i*NOmetSteps)], NOmetSteps, 1,
paste(interaction, \"with C.cat-aro: \", aroParam, \"($identfier)\", paste(\"Potential:\",
colnames(scores)[((i-1)*NOmetSteps+1):(i*NOmetSteps)], \"Data density:\", \"Distance (Å...)\"\",
paste(\"${pngFolder}/dist_\", interaction, \"_violin_aro_\", aroParam, \"_top\", topPoses,
sep=\"\"), \"png\")
    }
  }
  # Violin-plot distance changes in the other parameter next to each other
if (NOmetSteps == 1) {
  metParam <- unlist(strsplit(colnames(distances), \"_\")[2*(1:NOmetSteps)][1]
  violinSuperPlot(distances, NOaroSteps, 1, paste(interaction, \"with C.cat-MetS: \",
metParam, \"($identfier)\", paste(\"Potential:\", colnames(scores)[1:NOaroSteps]), \"Data
density:\", \"Distance (Å...)\"\", paste(\"${pdfFolder}/dist_\", interaction, \"_violin_met_\",
metParam, \"_top\", topPoses, sep=\"\"), \"pdf\")
  violinSuperPlot(distances, NOaroSteps, 1, paste(interaction, \"with C.cat-MetS: \",
metParam, \"($identfier)\", paste(\"Potential:\", colnames(scores)[1:NOaroSteps]), \"Data
density:\", \"Distance (Å...)\"\", paste(\"${pngFolder}/dist_\", interaction, \"_violin_met_\",
metParam, \"_top\", topPoses, sep=\"\"), \"png\")
  } else if (NOaroSteps > 1){
    for (i in 1:NOmetSteps) {
      metParam <- unlist(strsplit(colnames(distances), \"_\")[2*(1:NOmetSteps)][i]
      violinSuperPlot(distances[, i+(0:(NOaroSteps-1))*NOmetSteps], NOaroSteps, 1,
paste(interaction, \"with C.cat-MetS: \", metParam, \"($identfier)\", paste(\"Potential:\",
colnames(scores)[i+(0:(NOaroSteps-1))*NOmetSteps]), \"Data density:\", \"Distance (Å...)\"\",
paste(\"${pdfFolder}/dist_\", interaction, \"_violin_met_\", metParam, \"_top\", topPoses,
sep=\"\"), \"pdf\")
      violinSuperPlot(distances[, i+(0:(NOaroSteps-1))*NOmetSteps], NOaroSteps, 1,
paste(interaction, \"with C.cat-MetS: \", metParam, \"($identfier)\", paste(\"Potential:\",
colnames(scores)[i+(0:(NOaroSteps-1))*NOmetSteps]), \"Data density:\", \"Distance (Å...)\"\",
paste(\"${pngFolder}/dist_\", interaction, \"_violin_met_\", metParam, \"_top\", topPoses,
sep=\"\"), \"png\")
    }
  }
}
}
} > ${workDir}/violin.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/violin.R >> $logfile

echo " - done."
printf "\n - done.\n" >> $logfile
echo "violinPLANTS completely done on `date`."
echo "See $logfile for logged information."
echo "violinPLANTS completely done on `date`." >> $logfile

```

A 17 Bash-Shell-Skript violinOverlay

```

#!/bin/bash
#####
#
#   violinOverlay id=ID [tops=TOPS]
#
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
#
# Parameters
#       ID       identifier (date format)
#       TOPS     number of top-scored poses to be analyzed
#               (default: use all poses)
#
# All necessary information is taken from the respective folders and files.
#####
# transfer arguments

```

```

while [ $# -gt 0 ] ; do
    case `echo "$1" | sed 's/=.*//'\` in
        id)    identifier=`echo "$1" | sed 's/.*/='\`;;
        tops) topPoses=`echo "$1" | sed 's/.*/='\`;;
        *)    echo "Could not interpret $1 as an entity index - abort."
              exit 1;;
    esac
    shift
done

if [ ! $identifier ] ; then
    echo "No identifier given - abort!"
    exit 1
fi

# Folder and file variables
mainDir=/raid/home/rklein/DOCK/plants/transform
workDir=$mainDir/transform_$identifier
logFile=$workDir/violinOverlay.log
evaLogFile=$workDir/evaPLANTS.log
transformerLog=${workDir}/transformer.log
pdfFolder=${workDir}/pdf
pngFolder=${workDir}/png

if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi
if [ ! -f $transformerLog ] ; then
    echo "Transformer log file $transformerLog was not found - abort!"
    exit 1
fi

# Just a hint from the last of the docking runs
loggedNumber=`sed -n '2 p' $evaLogFile | awk '{printf $5}'`

# Get common number of docking poses from transformer.log file
nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
for no in ${nos[*]}
do
    if [ $no -lt $loggedNumber ] ; then
        loggedNumber=$no
    fi
done

if [ ! $topPoses ] ; then
    topPoses=$loggedNumber
else
    if [ $topPoses -lt 1 ] ; then
        echo "At least one pose needs to be evaluated overall - abort"
        exit 1
    elif [ $topPoses -gt $loggedNumber ] ; then
        topPoses=$loggedNumber
    fi
fi

# Inform user about what will happen, log setup information
echo "$USER started violinOverlay for identifier $identifier on `date`."
echo "$USER started violinOverlay for identifier $identifier on `date`." > $logFile

aroSteps=( `sed -n '2 p' $transformerLog | sed 's/.*/interaction://g' | sort` )
metSteps=( `sed -n '3 p' $transformerLog | sed 's/.*/interaction://g' | sort` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

# Files in /docked_ARO_MET
scoreFile=scoreList.txt

# Files in .
scoreSuperFile=${workDir}/scores_sum.txt
distanceSuperFiles=( `find ${workDir}/lig*_sum.txt` )

```

```

interactions=( `find ${workDir}/lig*_sum.txt | sed 's/.*/\///; s/_sum\.txt//` )
interactions=( `echo "\"${interactions[*]}\""" | sed 's/ /"/, "/g'` )

# Translate to R-readable
distanceFiles=( `echo "\"${distanceSuperFiles[*]}\""" | sed 's/ /"/, "/g'` )
aroSteps=( `echo "\"${aroSteps[*]}\""" | sed 's/ /"/, "/g'` )
metSteps=( `echo "\"${metSteps[*]}\""" | sed 's/ /"/, "/g'` )
versions=( `echo "\"${folders[*]}\""" | sed 's/docked_/g; s/ /"/, "/g'` )

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile
printf "\tCreating violin plot overlays for score and distance distributions ...\n"
printf "\tCreating violin plot overlays for score and distance distributions ...\n" >>
$logFile

# Create violin plots
echo "# Script to produce violin plot overlays from scores and interactions from ID
$identifier (as of `date`)"

topPoses <- $stopPoses
interactions <- c(${interactions[*]})
NOaroSteps <- ${#aroSteps[*]}
NometSteps <- ${#metSteps[*]}

# Function to overlay up to six violinPlots
# for a multi-color interpolation, use as lineColors e.g.: colorRampPalette(c("green",
"\white", "red"))(length(indices))
# (But be careful about white lines on white background)
multiViolinPlot <- function(matrixData, indices, mainTitle, subLabels, xLabel, yLabel,
lineColors, fileName, deviceName, colNumber = 3) {
  if (deviceName == "pdf") {
    saveFile <- paste(fileName, ".pdf", sep="\")
    pdf(saveFile)
  } else
  if (deviceName == "png") {
    saveFile <- paste(fileName, ".png", sep="\")
    png(saveFile, width = 720, height = 720)
  }
  layout(matrix(c(1,1,2,3,4,4), 3, 2, byrow=TRUE), height = c(1, 10, 3.5))

  # Title text
  par(mar = c(0,0,2.5,0))
  frame()
  mtext(mainTitle, cex = if (deviceName == "png") 1.3 else 1)

  # some space around the extreme density values
  maxValue <- max(density(matrixData[, 1])\$y)
  if (dim(matrixData)[2] > 1) for (j in 2:dim(matrixData)[2]) if (max(density(matrixData[,
j])\$y) > maxValue) maxValue <- max(density(matrixData[, j])\$y)
  minValue <- min(density(matrixData[, 1])\$y)
  if (dim(matrixData)[2] > 1) for (j in 2:dim(matrixData)[2]) if (min(density(matrixData[,
j])\$y) < minValue) minValue <- min(density(matrixData[, j])\$y)
  xAxAdd <- abs(maxValue - minValue) * 0.2

  # Plot density function inversely
  par(mar=c(2, 2, 0, 0))
  plot(1, type="n", xlim = c(-abs(maxValue) - xAxAdd, 0), ylim =
range(density(matrixData)\$x), xlab = "", ylab = "", xaxs = "i", axes = FALSE)
  for (i in indices){
    lines(-density(matrixData[, i])\$y, density(matrixData[, i])\$x, col =
lineColors[which(indices==i)], lwd = 2)
    lines(c(-0.2*xAxAdd, 0), rep(boxplot.stats(matrixData[, i])\$stats[3], 2), lwd = 2, col =
lineColors[which(indices==i)])
  }

  #title(main = mainTitle)
  title(line = 1, xlab = xLabel, cex.lab = 1.5)
  title(line = 0, ylab = yLabel, cex.lab = 1.5)
  box()

  # plot density function normally
  par(mar=c(2, 0, 0, 1))
  plot(1, type="n", xlim = c(0, abs(maxValue) + xAxAdd), ylim =
range(density(matrixData)\$x), xlab = "", ylab = "", xaxs = "i")

```

```

for (i in indices){
  lines(density(matrixData[, i])\$y, density(matrixData[, i])\$x, col =
lineColors[which(indices==i)], lwd = 2)
  lines(c(0, 0.2*xAxAdd), rep(boxplot.stats(matrixData[, i])\$stats[3], 2), lwd = 2, col =
lineColors[which(indices==i)])
}

# add legend
par(mar=c(0, 2, 1, 1))
plot.new()
plot.window(c(0,1), c(0,1))
legend("center", subLabels, col = lineColors, bg = "white", ncol = colNumber, title =
"Depicted potentials", lwd = 2, cex = if (deviceName == "png") 1.5 else 1.2)

if (deviceName == "pdf" || deviceName == "png") dev.off()
}

# Read in scores
scores <- read.table("\$scoreSuperFile", header = TRUE, fill = TRUE)[1:topPoses, ]
if (NOaroSteps > 1 || NOmetSteps > 1) colnames(scores) <- sub("\_.\", \"_\", sub(\"X.\", \"-
\", colnames(scores)))
scores <- as.matrix(cbind(scores))

# Multiple violin-plot scores for changes in one parameter
if (NOmetSteps > 1) for (i in 1:NOaroSteps) {
  indices <- seq(1+(i-1)*NOmetSteps, i*NOmetSteps)
  aroParam <- unlist(strsplit(colnames(scores)[indices], \"_\")[1])
  multiViolinPlot(scores, indices, \"Docking score densities (\$identfier)\",
colnames(scores)[indices], \"Data density:\", \"Docking scores (arbitrary units)\",
colorRampPalette(c(\"green\", \"red\"))(length(indices)),
paste(\"${workDir}/scores_viOverlay_aro_\", aroParam, \"_top\", topPoses, sep=\"\"), \"pdf\")
  multiViolinPlot(scores, indices, \"Docking score densities (\$identfier)\",
colnames(scores)[indices], \"Data density:\", \"Docking scores (arbitrary units)\",
colorRampPalette(c(\"green\", \"red\"))(length(indices)),
paste(\"${workDir}/scores_viOverlay_aro_\", aroParam, \"_top\", topPoses, sep=\"\"), \"png\")
}

# Multiple violin-plot scores for changes in the other parameter
if (NOaroSteps > 1) for (i in 1:NOmetSteps) {
  indices <- seq(i,i+(NOaroSteps-1)*NOmetSteps, NOmetSteps)
  metParam <- unlist(strsplit(colnames(scores)[indices], \"_\")[2])
  multiViolinPlot(scores, indices, \"Docking score densities (\$identfier)\",
colnames(scores)[indices], \"Data density:\", \"Docking scores (arbitrary units)\",
colorRampPalette(c(\"green\", \"red\"))(length(indices)),
paste(\"${workDir}/scores_viOverlay_met_\", metParam, \"_top\", topPoses, sep=\"\"), \"pdf\")
  multiViolinPlot(scores, indices, \"Docking score densities (\$identfier)\",
colnames(scores)[indices], \"Data density:\", \"Docking scores (arbitrary units)\",
colorRampPalette(c(\"green\", \"red\"))(length(indices)),
paste(\"${workDir}/scores_viOverlay_met_\", metParam, \"_top\", topPoses, sep=\"\"), \"png\")
}

# Read in all distances
distances <- list()
length(distances) <- length(interactions)
for (i in 1:length(interactions)) {
  distances[[i]] <- read.table(paste(\"${workDir}/\", interactions[i], \"_sum.txt\", sep =
\"\"), header = TRUE, fill = TRUE)[1:topPoses, ]
  if (NOaroSteps > 1 || NOmetSteps > 1) colnames(distances[[i]]) <- sub(\"_\", \"_\",
sub(\"X.\", \"-\", colnames(distances[[i]]))) else colName <- c({versions[*]})
  distances[[i]] <- as.matrix(cbind(distances[[i]]))
}

# Multiple violin-plot distances
for (i in 1:length(interactions)) {
  # Multiple violin-plot scores for changes in one parameter
  if (NOmetSteps > 1) for (aroStep in 1:NOaroSteps) {
    indices <- seq(1+(aroStep-1)*NOmetSteps, aroStep*NOmetSteps)
    aroParam <- unlist(strsplit(colnames(distances[[i]])[indices], \"_\")[1])
    multiViolinPlot(distances[[i]], indices, paste(interactions[i], \"with C.cat-aro: \",
aroParam, \"(\$identfier)\", colnames(distances[[i]])[indices], \"Data density:\", \"Distance
(Å)\", colorRampPalette(c(\"green\", \"red\"))(length(indices)), paste(\"${workDir}/dist_\",
interactions[i], \"_viOverlay_aro_\", aroParam, \"_top\", topPoses, sep=\"\"), \"pdf\")
    multiViolinPlot(distances[[i]], indices, paste(interactions[i], \"with C.cat-aro: \",
aroParam, \"(\$identfier)\", colnames(distances[[i]])[indices], \"Data density:\", \"Distance

```



```

(Ä...)\"", colorRampPalette(c(\"green\", \"red\"))(length(indices)), paste(\"${workDir}/dist_\",
interactions[i], \"_viOverlay_aro_\", aroParam, \"_top\", topPoses, sep=\", \"), \"png\")
}

# Multiple violin-plot scores for changes in the other parameter
if (NOaroSteps > 1) for (metStep in 1:NOmetSteps) {
  indices <- seq(metStep, metStep+(NOaroSteps-1)*NOmetSteps, NOmetSteps)
  metParam <- unlist(strsplit(colnames(distances[[i]][indices], \"_\")[2]
  multiViolinPlot(distances[[i]], indices, paste(interactions[i], \"with C.cat-MetS: \",
metParam, \"($identifizier)\", colnames(distances[[i]][indices], \"Data density:\", \"Distance
(Ä...)\"), colorRampPalette(c(\"green\", \"red\"))(length(indices)), paste(\"${workDir}/dist_\",
interactions[i], \"_viOverlay_met_\", metParam, \"_top\", topPoses, sep=\", \"), \"pdf\")
  multiViolinPlot(distances[[i]], indices, paste(interactions[i], \"with C.cat-MetS: \",
metParam, \"($identifizier)\", colnames(distances[[i]][indices], \"Data density:\", \"Distance
(Ä...)\"), colorRampPalette(c(\"green\", \"red\"))(length(indices)), paste(\"${workDir}/dist_\",
interactions[i], \"_viOverlay_met_\", metParam, \"_top\", topPoses, sep=\", \"), \"png\")
}
}

# Plot pairwise multiple violin distances without redundancy
# This loop creates about (NOmetSteps + NOaroSteps) * (length(interactions) - 1) *
(length(interactions) / 2) graphics files
if (NOaroSteps > 1 || NOmetSteps > 1) {
  for (i in 1:(length(interactions) - 1)) for (j in (i+1):length(interactions)) {
    iInteraction <- interactions[i]
    jInteraction <- interactions[j]
    for (aroStep in 1:NOaroSteps) {
      indices <- seq(1+(aroStep-1)*NOmetSteps, aroStep*NOmetSteps)
      aroParam <- unlist(strsplit(colnames(distances[[i]][indices], \"_\")[1]
      legLabels <- c(paste(colnames(distances[[i]][indices], \" (\", iInteraction, \")\"), sep
= \" \"), paste(colnames(distances[[j]][indices], \" (\", jInteraction, \")\"), sep = \" \"))
      indices <- c(indices, indices + NOaroSteps * NOmetSteps)
      multiViolinPlot(cbind(distances[[i]], distances[[j]]), indices, paste(iInteraction,
\"vs.\", jInteraction, \"with C.cat-aro: \", aroParam, \"($identifizier)\", legLabels, \"Data
density:\", \"Distance (Ä...)\"), colorRampPalette(c(\"green\", \"#BFFFBF\", \"#FF3F3F\",
\"red\"))(length(indices)), paste(\"${pdfFolder}/comp_\", iInteraction, \"_\", jInteraction,
\"_vio_aro_\", aroParam, \"_top\", topPoses, sep=\", \"), \"pdf\", 2)
      multiViolinPlot(cbind(distances[[i]], distances[[j]]), indices, paste(iInteraction,
\"vs.\", jInteraction, \"with C.cat-aro: \", aroParam, \"($identifizier)\", legLabels, \"Data
density:\", \"Distance (Ä...)\"), colorRampPalette(c(\"green\", \"#BFFFBF\", \"#FF3F3F\",
\"red\"))(length(indices)), paste(\"${pngFolder}/comp_\", iInteraction, \"_\", jInteraction,
\"_vio_aro_\", aroParam, \"_top\", topPoses, sep=\", \"), \"png\", 2)
    }
    for (metStep in 1:NOmetSteps) {
      indices <- seq(metStep, metStep+(NOaroSteps-1)*NOmetSteps, NOmetSteps)
      metParam <- unlist(strsplit(colnames(distances[[j]][indices], \"_\")[2]
      legLabels <- c(paste(colnames(distances[[i]][indices], \" (\", iInteraction, \")\"), sep
= \" \"), paste(colnames(distances[[j]][indices], \" (\", jInteraction, \")\"), sep = \" \"))
      indices <- c(indices, indices + NOaroSteps * NOmetSteps)
      multiViolinPlot(cbind(distances[[i]], distances[[j]]), indices, paste(iInteraction,
\"vs.\", jInteraction, \"with C.cat-MetS: \", metParam, \"($identifizier)\", legLabels, \"Data
density:\", \"Distance (Ä...)\"), colorRampPalette(c(\"green\", \"#BFFFBF\", \"#FF3F3F\",
\"red\"))(length(indices)), paste(\"${pdfFolder}/comp_\", iInteraction, \"_\", jInteraction,
\"_vio_met_\", metParam, \"_top\", topPoses, sep=\", \"), \"pdf\", 2)
      multiViolinPlot(cbind(distances[[i]], distances[[j]]), indices, paste(iInteraction,
\"vs.\", jInteraction, \"with C.cat-MetS: \", metParam, \"($identifizier)\", legLabels, \"Data
density:\", \"Distance (Ä...)\"), colorRampPalette(c(\"green\", \"#BFFFBF\", \"#FF3F3F\",
\"red\"))(length(indices)), paste(\"${pngFolder}/comp_\", iInteraction, \"_\", jInteraction,
\"_vio_met_\", metParam, \"_top\", topPoses, sep=\", \"), \"png\", 2)
    }
  }
} else for (i in 1:(length(interactions) - 1)) for (j in (i+1):length(interactions)) {
  iInteraction <- interactions[i]
  jInteraction <- interactions[j]
  legLabels <- c(iInteraction, jInteraction)
  multiViolinPlot(cbind(distances[[i]], distances[[j]]), 1:2, paste(iInteraction, \"vs.\",
jInteraction, \"with \", colName, \"($identifizier)\", legLabels, \"Data density:\", \"Distance
(Ä...)\"), c(\"green\", \"red\"), paste(\"${pdfFolder}/comp_\", iInteraction, \"_\",
jInteraction, \"_vio_\", colName, \"_top\", topPoses, sep=\", \"), \"pdf\", 2)
  multiViolinPlot(cbind(distances[[i]], distances[[j]]), 1:2, paste(iInteraction, \"vs.\",
jInteraction, \"with \", colName, \"($identifizier)\", legLabels, \"Data density:\", \"Distance
(Ä...)\"), c(\"green\", \"red\"), paste(\"${pngFolder}/comp_\", iInteraction, \"_\",
jInteraction, \"_vio_\", colName, \"_top\", topPoses, sep=\", \"), \"png\", 2)
} > ${workDir}/violinOverlay.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/violinOverlay.R >> $logfile

echo " - done."

```



```

# Files in .
scoreSuperFile=${workDir}/scores_sum.txt
distanceSuperFiles=( `find ${workDir}/lig*_sum.txt` )

interactions=( `find ${workDir}/lig*_sum.txt | sed 's/.*/\//; s/_sum\.txt//` )
interactions=( `echo "\"${interactions[*]}\""" | sed 's/ /", "/g` )

# Translate to R-readable
distanceFiles=( `echo "\"${distanceSuperFiles[*]}\""" | sed 's/ /", "/g` )
aroSteps=( `echo "\"${aroSteps[*]}\""" | sed 's/ /", "/g` )
metSteps=( `echo "\"${metSteps[*]}\""" | sed 's/ /", "/g` )
versions=( `echo "\"${folders[*]}\""" | sed 's/docked_/g; s/ /", "/g` )

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile
printf "\tCreating staggered violin plots for score and distance distributions varying the
number of top poses considered ...\n"
printf "\tCreating staggered violin plots for score and distance distributions varying the
number of top poses considered ...\n" >> $logFile

# Create violin plots
echo "# Script to produce staggered violin plots from scores and interactions from ID
$identifier (as of `date`)"
# Will produce NOmetsteps x NOaroSteps x (length(interactions) + 1) .pdf and .png files

NOposes <- $loggedNumber
NOsteps <- 10
if (NOsteps >= NOposes) stop()
steps <- round(seq(NOposes/NOsteps, NOposes, NOposes/NOsteps))
interactions <- c(${interactions[*]})
NOaroSteps <- ${#aroSteps[*]}
NOmetSteps <- ${#metSteps[*]}

# Function to produce staggered violin plots varying the number of top poses considered
# for a multi-color interpolation, use as lineColors e.g.: colorRampPalette(c("green",
"white", "red"))(length(tops))
# (But be careful about white lines on white background)
topsViolinPlot <- function (dataVector, tops, mainTitle, subLabels, xLabel, yLabel,
lineColors, fileName, deviceName, colNumber = 3) {
  if (deviceName == "pdf") {
    saveFile <- paste(fileName, ".pdf", sep="\")
    pdf(saveFile)
  } else
  if (deviceName == "png") {
    saveFile <- paste(fileName, ".png", sep="\")
    png(saveFile, width = 720, height = 720)
  }
  layout(matrix(c(1,1,2,3,4,4), 3, 2, byrow=TRUE), height = c(1, 10, 3.5))

  # Title text
  par(mar = c(0,0,2.5,0))
  frame()
  mtext(mainTitle, cex = if (deviceName == "png") 1.3 else 1)

  # some space around the extreme density values
  maxValue <- max(density(dataVector[1:tops[1]])\$y)
  for (i in 1:length(tops)) if (max(density(dataVector[1:tops[i]])\$y) > maxValue) maxValue <-
max(density(dataVector[1:tops[i]])\$y)
  minValue <- min(density(dataVector[1:tops[1]])\$y)
  for (i in 1:length(tops)) if (min(density(dataVector[1:tops[i]])\$y) < minValue) minValue <-
min(density(dataVector[1:tops[i]])\$y)
  xAxAdd <- abs(maxValue - minValue) * 0.1

  # Plot density function inversely
  par(mar=c(2, 2, 0, 0))
  plot(1, type="n", xlim = c(-abs(maxValue) - xAxAdd, 0), ylim =
range(density(dataVector)\$x), xlab = "", ylab = "", xaxs = "i", axes = FALSE)
  for (top in tops){
    lines(-density(dataVector[1:top])\$y, density(dataVector[1:top])\$x, col =
lineColors[which(tops==top)], lwd = 2)
    lines(c(-0.2*xAxAdd, 0), rep(boxplot.stats(dataVector[1:top])\$stats[3], 2), lwd = 2, col
= lineColors[which(tops==top)])
  }
  title(line = 1, xlab = xLabel, cex.lab = 1.5)

```

```

title(line = 0, ylab = yLabel, cex.lab = 1.5)
box()

# plot density function normally
par(mar=c(2, 0, 0, 1))
plot(1, type="n", xlim = c(0, abs(maxValue) + xAxAdd), ylim =
range(density(dataVector)\$x), xlab = "\", ylab = "\", xaxs = \"i\")
for (top in tops){
  lines(density(dataVector[1:top])\$y, density(dataVector[1:top])\$x, col =
lineColors[which(tops==top)], lwd = 2)
  lines(c(0, 0.2*xAxAdd), rep(boxplot.stats(dataVector[1:top])\$stats[3], 2), lwd = 2, col =
lineColors[which(tops==top)])
}

# add legend
par(mar=c(0, 2, 1, 1))
plot.new()
plot.window(c(0,1), c(0,1))
legend("center", subLabels, col = lineColors, bg = "white", ncol = colNumber, title =
"Number of top docking poses used", lwd = 2, cex = if (deviceName == "png") 1.5 else 1.2)

if (deviceName == "pdf" || deviceName == "png") dev.off()
}

# Read in scores
scores <- read.table("\$scoreSuperFile", header = TRUE, fill = TRUE)[1:NOposes, ]
if (NOaroSteps > 1 || NOmetSteps > 1) colnames(scores) <- c({versions[*]})
scores <- as.matrix(cbind(scores))

# Multiple violin-plot scores for all parameter translations
for (i in 1:NOaroSteps) for (j in 1:NOmetSteps) {
  if (NOaroSteps > 1 && NOmetSteps > 1) potential <- colnames(scores)[(i-1)*NOmetSteps+j] else
potential <- c({versions[*]})
  topsViolinPlot(scores[, (i-1)*NOmetSteps+j], steps, paste("Docking score densities (\",
potential, \", $identifier)\", sep = "\""), paste("\top \", steps), "Data density:\",
"Docking scores (arbitrary units)\", colorRampPalette(c("red", "green"))(NOsteps),
paste("\${pdfFolder}/scores_tops_\", potential, sep="\\"", "pdf", 5)
  topsViolinPlot(scores[, (i-1)*NOmetSteps+j], steps, paste("Docking score densities (\",
potential, \", $identifier)\", sep = "\""), paste("\top \", steps), "Data density:\",
"Docking scores (arbitrary units)\", colorRampPalette(c("red", "green"))(NOsteps),
paste("\${pngFolder}/scores_tops_\", potential, sep="\\"", "png", 5)
}

# Read in all distances
distances <- list()
length(distances) <- length(interactions)
for (i in 1:length(interactions)) {
  distances[[i]] <- read.table(paste("\${workDir}/\", interactions[i], "_sum.txt", sep =
"\\"", header = TRUE, fill = TRUE)[1:NOposes, ]
  if (NOaroSteps > 1 || NOmetSteps > 1) colnames(distances[[i]]) <- c({versions[*]})
  distances[[i]] <- as.matrix(cbind(distances[[i]]))
}

# Multiple violin-plot distances for all parameter translations
for (i in 1:NOaroSteps) for (j in 1:NOmetSteps) for (k in 1:length(distances)) {
  if (NOaroSteps > 1 && NOmetSteps > 1) potential <- colnames(distances[[k]])[(i-
1)*NOmetSteps+j] else potential <- c({versions[*]})
  topsViolinPlot(distances[[k]][, (i-1)*NOmetSteps+j], steps, paste(interactions[k], \" (\",
potential, \", $identifier)\", sep = "\""), paste("\top \", steps), "Data density:\",
"Distance (Å...)\", colorRampPalette(c("red", "green"))(NOsteps),
paste("\${pdfFolder}/tops_\", interactions[k], "_\", potential, sep="\\"", "pdf", 5)
  topsViolinPlot(distances[[k]][, (i-1)*NOmetSteps+j], steps, paste(interactions[k], \" (\",
potential, \", $identifier)\", sep = "\""), paste("\top \", steps), "Data density:\",
"Distance (Å...)\", colorRampPalette(c("red", "green"))(NOsteps),
paste("\${pngFolder}/tops_\", interactions[k], "_\", potential, sep="\\"", "png", 5)
} > ${workDir}/violinTops.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/violinTops.R >> $logFile

echo " - done."
printf "\n - done.\n" >> $logFile
echo "violinTops completely done on `date`."
echo "See $logFile for logged information."
echo "violinTops completely done on `date`." >> $logFile

```

A 19 Bash-Shell-Skript drawPLPs

```
#!/bin/bash
```



```

for (i in 1:length(minima)) {
  if (minima[i] == -0.4) {
    if (minima[i] %in% both) potLabel <- paste(bothLabel, origLabel, sep = "\", \") else
    if (minima[i] %in% aroOnly) potLabel <- paste(aroLabel, origLabel, sep = "\", \") else
    if (minima[i] %in% metOnly) potLabel <- paste(metLabel, origLabel, sep = "\", \") else
    potLabel <- origLabel
  } else {
    if (minima[i] %in% both) potLabel <- bothLabel else
    if (minima[i] %in% aroOnly) potLabel <- aroLabel else
    if (minima[i] %in% metOnly) potLabel <- metLabel
  }
  potLabels <- c(potLabels, paste(minima[i], "\" (\", potLabel, \")\"", sep = "\"\"))
}

margin <- 0.3;
plpA <- 3.4
plpB <- 3.6
plpC <- 4.5
plpD <- 5.5
plpE <- -0.4
plpF <- 15

saveFile <- \"${workDir}/${saveFile}\"

lineCols <- rainbow(length(minima))
lineCols[which(minima == -0.4)] <- \"black\"

pdf(paste(saveFile, \".pdf\", sep=\"\"))
plot(c(3.0, plpD + margin), y = c(min(minima), 0.8 * abs(min(minima))), xlab = \"Receptor -
ligand atom distance (Å...)\", ylab = \"Score (arbitrary units)\", type = \"n\")
title(main = \"CHEMPLP versions ($identifier)\")
lines(c(0, plpD + margin), c(0, 0), lwd = 1, col = \"gray\")

# Draw (modified) potentials
for (minimum in minima) {
  lines(c(plpA, plpB), c(0, minimum), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(plpB, plpC), c(minimum, minimum), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(plpC, plpD), c(minimum, 0), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(0, plpD + margin), c(minimum, minimum), lty = 2, col = lineCols[which(minima ==
minimum)])
}
# Remaining original part
lines(c(0, plpA), c(plpF, 0), lwd = 4, col = \"black\")
lines(c(plpD, plpD + margin), c(0, 0), lwd = 4, col = \"black\")

legend(plpB, y = 0.8 * abs(min(minima)), potLabels, col = lineCols, lwd = 2)
dev.off()

png(paste(saveFile, \".png\", sep=\"\"), width = 1024, height = 768)
plot(c(3.0, plpD + margin), y = c(min(minima), 0.8 * abs(min(minima))), xlab = \"Receptor -
ligand atom distance (Å...)\", ylab = \"Score (arbitrary units)\", type = \"n\")
title(main = \"CHEMPLP versions ($identifier)\")
lines(c(0, plpD + margin), c(0, 0), lwd = 1, col = \"gray\")

# Draw (modified) potentials
for (minimum in minima) {
  lines(c(plpA, plpB), c(0, minimum), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(plpB, plpC), c(minimum, minimum), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(plpC, plpD), c(minimum, 0), lwd = 2, col = lineCols[which(minima == minimum)])
  lines(c(0, plpD + margin), c(minimum, minimum), lty = 2, col = lineCols[which(minima ==
minimum)])
}
# Remaining original part
lines(c(0, plpA), c(plpF, 0), lwd = 4, col = \"black\")
lines(c(plpD, plpD + margin), c(0, 0), lwd = 4, col = \"black\")

legend(plpB, y = 0.8 * abs(min(minima)), potLabels, col = lineCols, lwd = 2)
dev.off() > ${workDir}/drawPLPs.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/drawPLPs.R >> $logFile

echo \" - done.\"
echo \" - done.\" >> $logFile
echo \"drawPLPs completely done on `date`. See $logFile for logged information.\"
echo \"drawPLPs completely done on `date`.\" >> $logFile

```



```

aroSteps=( `sed -n '2 p' $transformerLog | sed 's/.*interaction://g' | sort` )
metSteps=( `sed -n '3 p' $transformerLog | sed 's/.*interaction://g' | sort` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

# Files in /docked_ARO_MET
scoreFile=scoreList.txt
featureFile=features.csv

# Files in .
scoreSuperFile=${workDir}/scores_sum.txt
distanceSuperFiles=( `find ${workDir}/lig*_sum.txt` )

interactions=( `find ${workDir}/lig*_sum.txt | sed 's/.*\\///; s/_sum\\.txt//'` )
interactions=( `echo "${interactions[*]}" | sed 's/ /", "/g'` )

# Translate to R-readable
distanceFiles=( `echo "${distanceSuperFiles[*]}" | sed 's/ /", "/g'` )
aroSteps=( `echo "${aroSteps[*]}" | sed 's/ /", "/g'` )
metSteps=( `echo "${metSteps[*]}" | sed 's/ /", "/g'` )
versions=( `echo "${folders[*]}" | sed 's/docked_//g; s/ /", "/g'` )

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile
printf "\tCreating partition plots for scores...\n"
printf "\tCreating partition plots for scores...\n" >> $logFile

# Create score partition plots
echo "# Script to produce violin plot overlays from scores and interactions from ID
$identifier (as of `date`)"

topPoses <- $topPoses
interactions <- c(${interactions[*]})
aroSteps <- c(${aroSteps[*]})
metSteps <- c(${metSteps[*]})
NOaroSteps <- length(aroSteps)
NOMETSteps <- length(metSteps)
potentials <- c(${versions[*]})

# Function to plot multiple data columns with lines
multiLinePlot <- function (matrixData, mainTitle, legTitle, xLabel, yLabel, lineColors,
fileName, deviceName, colNumber = 3) {
    if (deviceName == "pdf") {
        saveFile <- paste(fileName, ".pdf", sep="")
        pdf(saveFile)
    } else
    if (deviceName == "png") {
        saveFile <- paste(fileName, ".png", sep="")
        png(saveFile, width = 720, height = 720)
    }

    layout(matrix(c(1,2), 2, 1), height = c(8, 2))
    plot(1, type = "n", xlim = c(1,dim(matrixData)[1]), ylim = range(range(matrixData)), xlab =
xLabel, ylab = yLabel, main = mainTitle)

    for (i in 1:length(matrixData)) lines(1:dim(matrixData)[1],
matrixData[1:dim(matrixData)[1],i], col = lineColors[i])

    # add legend
    par(mar = c(0,1,0,1))
    plot.new()
    corCoeffs <- round(cor(matrixData)[1, ], 2)
    legend("top", paste(colnames(considered), "(r = ", corCoeffs, ")"), sep = "\n", col =
lineColors, ncol = colNumber, title = legTitle, lwd = 2, cex = 0.7)

```



```

    if (deviceName == \"pdf\" || deviceName == \"png\") dev.off()
  }

# Function to plot correlation coefficient of from two variables in heatmap-style
heatCorrelation <- function (corMatrix, mainTitle, xLabel, yLabel, boxLabel, fileName,
deviceName) {
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep=\"\")
    pdf(saveFile, width = 8, height = 6)
  } else
  if (deviceName == \"png\") {
    saveFile <- paste(fileName, \".png\", sep=\"\")
    png(saveFile, width = 1000, height = 720)
  }

  # interpolate colors and extract from complete palette
  NOcolors <- 1000
  minIndex <- min(round((1+NOcolors)/2 + corMatrix * (((NOcolors+1)/2)-1)))
  maxIndex <- max(round((1+NOcolors)/2 + corMatrix * (((NOcolors+1)/2)-1)))

  layout(matrix(c(1,2), 1,2 ), widths=c(5, 1))
  par(mar = c(6.5, 10.5, 4, 1))
  image(1:dim(corMatrix)[1], 1:dim(corMatrix)[2], corMatrix, main = mainTitle, xlab = \"\",
ylab = \"\", col = colorRampPalette(c(\"green\", \"white\",
\"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
  axis(1, at = 1:dim(corMatrix)[1], labels = dimnames(corMatrix)[[1]], las = 2, cex.axis =
0.6)
  axis(2, at = 1:dim(corMatrix)[2], labels = dimnames(corMatrix)[[2]], las = 2, cex.axis =
0.6)
  title(line = 9, ylab = yLabel)
  title(line = 5, xlab = xLabel)
  box()
  if (boxLabel != \"\") for (i in 1:dim(corMatrix)[1]) for (j in 1:dim(corMatrix)[2]) if
(corMatrix[i+(j-1)*dim(corMatrix)[1]] != 1.000001) text(i,j, paste(boxLabel,
round(corMatrix[i+(j-1)*dim(corMatrix)[1]], 2)), cex = 0.6)
  # plot a colorful legend
  par(mar=c(5,4,4,0)+0.1)
  image(1, 1:100, matrix(1:100, 1, 100), xlab=\"\", ylab=\"Pearson correlation coefficient\",
col = colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE)
  axis(2, at = seq(1,100, length.out=9), labels = round(seq(-1,1,length.out=9), 2), las = 1)

  if (deviceName == \"pdf\" || deviceName == \"png\") dev.off()
}

# Use only relevant score parts
# * TOTAL_SCORE: scoring function value obtained during docking
# * PLPparthbond: PLP hbond score
# * PLPpartsteric: PLP steric contact score
# * PLPpartmetal: PLP metal interaction score
# * PLPpartrepulsive: PLP donor/donor and acceptor/acceptor repulsion score
# * PLPpartburpolar: PLP buried polar atoms score (polar atoms occluded by nonpolar ones)
# * TRIPOS_LJ: (deduced: contact potential)
# * TRIPOS_TORS: intra-ligand torsion score
# * CHEMPLP_PLP_PROT: (deduced: intra-protein score, only calculated for i-,ex. side-chains)
# * TRIPOS_TORS_PROT: (deduced: protein torsion potential)
# * CHEMPLP_CLASH2_PROT: (deduced: intra-ligand clash score)
# * ATOMS_OUTSIDE_BINDINGSITE: number of ligand atoms outside binding site

relevantParts <- c(\"TOTAL_SCORE\", \"PLPparthbond\", \"PLPpartsteric\", \"PLPpartmetal\",
\"PLPpartrepulsive\", \"PLPpartburpolar\", \"TRIPOS_LJ\", \"TRIPOS_TORS\",
\"CHEMPLP_PLP_PROT\", \"TRIPOS_TORS_PROT\", \"CHEMPLP_CLASH2_PROT\",
\"ATOMS_OUTSIDE_BINDINGSITE\")

# Read in top scores from all potentials
scores <- list()
length(scores) <- length(potentials)
for (i in 1:length(potentials)) {
  featFile <- paste(\"${workDir}/\", \"docked_\", potentials[i], \"/${featureFile}\",
sep=\"\")
  scores[[i]] <- read.table(featFile, header = TRUE, sep=\",\") [1:topPoses, relevantParts]
}

# Plot score components
for (i in 1:length(potentials)) {
  # Plot only those score parts with non-constant contributions
  considered <- scores[[i]][which(sapply(scores[[i]], sd) != 0)]
}

```

```

    mainTitle <- paste("Score partitioning using potential", potentials[i],
    \"${identifier}\")
    saveFile <- paste("${workDir}/\", \"docked_\", potentials[i], \"/scoreParts_top\",
topPoses, sep=\"\")
    multiLinePlot(considered, mainTitle, \"Depicted potential components (Pearson correlation
coefficient to TOTAL_SCORE)\", \"Docking poses\", \"Docking scores (arbitrary units)\",
rainbow(length(considered)), saveFile, \"pdf\", 2)
    multiLinePlot(considered, mainTitle, \"Depicted potential components (Pearson correlation
coefficient to TOTAL_SCORE)\", \"Docking poses\", \"Docking scores (arbitrary units)\",
rainbow(length(considered)), saveFile, \"png\", 2)
}

# Read in all distances in a (large!) array
distances <- array(dim = c(topPoses, length(interactions), length(potentials)))
dimnames(distances) <- list(NULL, interactions, potentials)
for (i in 1:length(potentials)) {
  for (j in 1:length(interactions)) {
    distFile <- paste("${workDir}/\", \"docked_\", potentials[i], \"/\", interactions[j],
\"_dist.txt\", sep = \"\")
    distances[,j,i] <- read.table(distFile, header = FALSE, sep=\",\") [1:topPoses, ]
  }
}

# Correlate score components and distances for all potentials
for (i in 1:length(potentials)) {
  # Calculate Pearson correlation coefficients (except for constant components)
  corMatrix <- cor(distances[, ,i], scores[[i]][which(apply(scores[[i]], sd) != 0)])
  mainTitle <- paste("Score components vs. distances using", potentials[i],
\"${identifier}\")
  saveFile <- paste("${workDir}/\", \"partCor_\", potentials[i], \"_top\", topPoses, sep =
\"\")
  heatCorrelation(corMatrix, mainTitle, \"Measured distances\", \"Score components\", \"r =\",
saveFile, \"pdf\")
  heatCorrelation(corMatrix, mainTitle, \"Measured distances\", \"Score components\", \"r =\",
saveFile, \"png\")
}

# Correlate score components and potentials for all distances
for (i in 1:length(interactions)) {
  # Calculate Pearson correlation coefficients (except for constant components)
  coeffs <- c()
  for (j in 1:length(potentials)) coeffs <- c(coeffs, cor(distances[, i, j], scores[[j]]))

  # Probably including score components with no standard deviation here:
  corMatrix <- matrix(coeffs, nrow = length(potentials), ncol = length(relevantParts),
dimnames = list(potentials, relevantParts), byrow = TRUE)

  # Remove score components where there is (for at least one potential) no standard deviation
  considered <- c()
  for (part in relevantParts) if (max(is.na(corMatrix[, part])) == 0) considered <-
c(considered, part)
  corMatrix <- corMatrix[, considered]

  mainTitle <- paste(interactions[i], \": score components vs. potential \",
\"${identifier}\", sep = \"\")
  saveFile <- paste("${workDir}/\", \"partCor_\", interactions[i], \"_top\", topPoses, sep =
\"\")
  heatCorrelation(corMatrix, mainTitle, \"Used potential\", \"Score components\", \"\",
saveFile, \"pdf\")
  heatCorrelation(corMatrix, mainTitle, \"Used potential\", \"Score components\", \"\",
saveFile, \"png\")
} > ${workDir}/partPLANTS.R
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/partPLANTS.R >> $logFile

echo " - done."
printf "\n - done.\n" >> $logFile
echo "partPLANTS completely done on `date`."
echo "See $logFile for logged information."
echo "partPLANTS completely done on `date`." >> $logFile

```

A 21 Bash-Shell-Skript collectFeats

```

#!/bin/bash
#####

```

```
#
# collectFeats id
#
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
#
# Parameters
#         ID      identifier (date format)
#         TOPS    number of top-scored poses to be analyzed
#                 (default: use all poses)
#
# All necessary information is taken from the respective folders and files.
#####

# transfer arguments
if [ $# -ne 1 ] ; then
    echo "Accepting exactly one parameter (ID) - abort."
    exit 1
else
    identifier=$1
fi

# Folder and file variables
mainDir=/raid/home/rklein/DOCK/plants/transform
workDir=$mainDir/transform_${identifier}
logFile=$workDir/collectFeats.log
evaLogFile=$workDir/evaPLANTS.log
transformerLog=${workDir}/transformer.log

if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi

if [ ! -f $transformerLog ] ; then
    echo "Transformer log file $transformerLog was not found - abort!"
    exit 1
fi

# Get common number of docking poses from transformer.log file
number=`grep "Number of docking poses to be produced: " $transformerLog | awk '{print $8}`

# Inform user about what will happen, log setup information
echo "$USER started collectFeats for identifier $identifier on `date`."
echo "$USER started collectFeats for identifier $identifier on `date`." > $logFile

aroSteps=( `sed -n '2 p' $transformerLog | sed 's/.*interaction://g' | sort` )
metSteps=( `sed -n '3 p' $transformerLog | sed 's/.*interaction://g' | sort` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aro}_${met} )
    done
done

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"

# Files in docked_ARO_MET_NUMBER/
featureFile=features.csv

printf "\tCollecting feature.csv files into large ones...\n"
printf "\tCollecting feature.csv files into large ones...\n" >> $logFile

for folder in ${folders[*]}
do
    printf "\tWorking on $folder..."
    targetFile=${workDir}/${folder}/${featureFile}
    echo "LIGAND,
TOTAL_SCORE,SCORE_RB_PEN,SCORE_NORM_HEVATOMS,SCORE_NORM_CRT_HEVATOMS,SCORE_NORM_WEIGHT,SCORE_N
ORM_CRT_WEIGHT,SCORE_RB_PEN_NORM_CRT_HEVATOMS,SCORE_NORM_CONTACT,PLPtotal,PLPparthbond,PLPpart
steric,PLPpartmetal,PLPpartrepulsive,PLPpartburpolar,LIG_NUM_CLASH,LIG_NUM_CONTACT,LIG_NUM_NO
CONTACT,CHEMpartmetal,CHEMparthbond,CHEMparthbondCHO,DON,ACC,UNUSED_DON,UNUSED_ACC,TRIPOS_LJ,T
RIPOS_TORS,CHEMPLP_PLP_PROT,TRIPOS_TORS_PROT,CHEMPLP_CLASH2_PROT,ATOMS_OUTSIDE_BINDINGSITE" >
    $targetFile

    for i in `seq 1 $number`
```



```

        ana)    anaFile=`echo "$1" | sed 's/.*/'`;
        *)      echo "Could not interpret $1 as an entity index - abort."
                exit 1;;
    esac
    shift
done

if [ ! $identifier ] ; then
    echo "No identifier given - abort!"
    exit 1
fi
if [ ! $anaFile ] ; then
    echo "No analysis request file given - abort!"
    exit 1
fi
if [ ! -f $anaFile ] ; then
    echo "Analysis request file $anaFile not found - abort!"
    exit 1
fi

# Folder and file variables
mainDir=/raid/home/rklein/DOCK/plants/transform
workDir=$mainDir/transform_$identifier
transformerLog=${workDir}/transformer.log
distanceFolder=${workDir}/distances
logFile=${distanceFolder}/calcDistances.log

if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi
if [ ! -f $transformerLog ] ; then
    echo "Transformer log file $transformerLog was not found - abort!"
    exit 1
fi

if [ ! -d $distanceFolder ] ; then
    mkdir $distanceFolder
fi

# Inform user about what will happen, log setup information
echo "$USER started calcDistances for identifier $identifier and analysis file $anaFile on
`date`."
echo "$USER started calcDistances for identifier $identifier and analysis file $anaFile on
`date`." > $logFile

# Get minimum common number of docking poses
cluster=`grep "cluster" $transformerLog | wc -l`
nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
topPoses=${nos[0]}
if [ $cluster -gt 0 ] ; then
    printf "Docking was performed using the cluster option."
    printf "Docking was performed using the cluster option." >> $logFile
    for no in ${nos[*]}
    do
        if [ $no -lt $topPoses ] ; then
            topPoses=$no
        fi
    done
else
    printf "Independent docking (no clustering) was performed."
    printf "Independent docking (no clustering) was performed." >> $logFile
fi
echo " Minimum common number of poses: $topPoses."
echo " Minimum common number of poses: $topPoses." >> $logFile

aroSteps=( `grep "Potential minima for C.cat-Aro interaction:" $transformerLog | sed
's/.*/'` )
metSteps=( `grep "Potential minima for C.cat-MethioninS interaction:" $transformerLog | sed
's/.*/'` )

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile

for aroStep in ${aroSteps[*]}
do

```

```

    for metStep in ${metSteps[*]}
    do
        folders=( ${folders[*]} docked_${aroStep}_${metStep} )
    done
done

ligandName=`ls ${workDir}/${folders[0]}/*entry_00001_conf_01.mol2 | sed 's/.*\///' | sed
's/_entry_00001_conf_01\.mol2//'`
receptorFile=`grep "Receptor file (absolute path):" $transformerLog | awk '{printf $5}'`

# Parse input analysis file and calculate distances
printf "\tParsing input file $anaFile and calculating distances...\n"
printf "\tParsing input file $anaFile and calculating distances...\n" >> $logFile

NOL=`wc -l < $anaFile`
for lineNumber in `seq 1 $NOL`
do
    currentLine=`sed -n "$lineNumber p" $anaFile`
    # Do nothing with comment lines
    if [ `echo $currentLine | cut -c 1` == "#" ] ; then
        continue
    fi

    # number of words of current line
    NOW=`echo $currentLine | wc -w`
    atomSet1=( )
    atomSet2=( )
    distanceOutFiles=( )
    mode=start
    for wordNumber in `seq 1 $NOW`
    do
        currentWord=`echo $currentLine | awk -v w=$wordNumber '{print $w}'`

        # Switch reading mode accordingly
        if [ $currentWord == "LIG" ] && [ $mode == "start" ] ; then
            mode=setLigAtoms
        elif [ $currentWord == "LIG" ] && [ $mode == "setLigAtoms" ] ; then
            mode=setLigLigAtoms
        elif [ $currentWord == "REC" ] && [ $mode == "start" ] ; then
            mode=setRecAtoms
        elif [ $currentWord == "REC" ] && [ $mode == "setLigAtoms" ] ; then
            mode=setLigRecAtoms
        elif [ $currentWord == "RIGID" ] && [ $mode == "setLigRecAtoms" ] ; then
            mode=setLigRecRigidAtoms
        elif [ $currentWord == "FLEX" ] && [ $mode == "setLigRecAtoms" ] ; then
            mode=setLigRecFlexAtoms
        elif [ $currentWord == "RIGID" ] && [ $mode == "setRecAtoms" ] ; then
            mode=setRecRigidAtoms
        elif [ $currentWord == "FLEX" ] && [ $mode == "setRecAtoms" ] ; then
            mode=setRecFlexAtoms
        elif [ $currentWord == "LIG" ] && [ $mode == "setRecFlexAtoms" ] ; then
            mode=setRecFlexLigAtoms
        elif [ $currentWord == "LIG" ] && [ $mode == "setRecRigidAtoms" ] ; then
            mode=setRecRigidLigAtoms
        elif [ $currentWord == "REC" ] && [ $mode == "setRecFlexAtoms" ] ; then
            mode=setRecFlexRecAtoms
        elif [ $currentWord == "REC" ] && [ $mode == "setRecRigidAtoms" ] ; then
            mode=setRecRigidRecAtoms
        elif [ $currentWord == "RIGID" ] && [ $mode == "setRecRigidRecAtoms" ] ; then
            mode=setRecRigidRecRigidAtoms
        elif [ $currentWord == "FLEX" ] && [ $mode == "setRecRigidRecAtoms" ] ; then
            mode=setRecRigidRecFlexAtoms
        elif [ $currentWord == "FLEX" ] && [ $mode == "setRecFlexRecAtoms" ] ; then
            mode=setRecFlexRecFlexAtoms
        elif [ $currentWord == "RIGID" ] && [ $mode == "setRecFlexRecAtoms" ] ; then
            mode=setRecFlexRecRigidAtoms
        elif [ $currentWord == "DESC" ] && ( [ $mode == "setLigLigAtoms" ] || [ $mode
== "setLigRecRigidAtoms" ] || [ $mode == "setLigRecFlexAtoms" ] || [ $mode ==
"setRecFlexLigAtoms" ] || [ $mode == "setRecRigidLigAtoms" ] || [ $mode ==
"setRecRigidRecRigidAtoms" ] || [ $mode == "setRecRigidRecFlexAtoms" ] || [ $mode ==
"setRecFlexRecFlexAtoms" ] || [ $mode == "setRecFlexRecRigidAtoms" ] ) ; then
            fileMode=$mode
            mode=desc1
        # read accordingly
        elif [ $mode == "setLigAtoms" ] || [ $mode == "setRecRigidAtoms" ] || [ $mode
== "setRecFlexAtoms" ] ; then
            atomSet1=( ${atomSet1[*]} $currentWord )

```



```

        exit
    fi
    if [ ! -d $workDir ] ; then
        echo "The necessary folder $workDir could not be found - abort!"
        exit 1
    fi
    if [ ! -d $distanceFolder ] ; then
        echo "The necessary folder $distanceFolder could not be found - abort!"
        exit 1
    fi
    if [ ! -f $transformerLog ] ; then
        echo "Transformer log file $transformerLog was not found - abort!"
        exit 1
    fi

    # Inform user about what will happen, log setup information
    echo "$USER started evaTransform_c for identifier $identifier on `date`."
    echo "$USER started evaTransform_c for identifier $identifier on `date`." > $logFile

    # Get minimum common number of docking poses
    cluster=`grep "cluster" $transformerLog | wc -l`
    nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
    if [ ! $stopPoses ] ; then
        topPoses=${nos[0]}
    fi
    if [ $cluster -gt 0 ] ; then
        printf "Docking was performed using the cluster option."
        printf "Docking was performed using the cluster option." >> $logFile
        for no in ${nos[*]}
        do
            if [ $no -lt $topPoses ] ; then
                topPoses=$no
            fi
        done
    else
        printf "Independent docking (no clustering) was performed."
        printf "Independent docking (no clustering) was performed." >> $logFile
    fi
    echo " Minimum common number of poses: $topPoses."
    echo " Minimum common number of poses: $topPoses." >> $logFile

    aroSteps=( `grep "Potential minima for C.cat-Aro interaction:" $transformerLog | sed
    's/.*/\n/'` )
    metSteps=( `grep "Potential minima for C.cat-MethioninS interaction:" $transformerLog | sed
    's/.*/\n/'` )

    for aro in ${aroSteps[*]}
    do
        for met in ${metSteps[*]}
        do
            potentials=( ${potentials[*]} ${aro}_${met} )
        done
    done

    printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
    printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
    printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
    printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile

    mkdir -p $graphicsFolder

    superDistanceFiles=( `find ${distanceFolder}/ *.txt | sed 's/.*/\n/g' | grep "^super_dist_"` )
    printf "\tAnalyzing data for ${#superDistanceFiles[*]} measured interactions and
    ${#potentials[*]} different potentials.\n\tThis may take a while...\n"
    printf "\tAnalyzing data for ${#superDistanceFiles[*]} measured interactions and
    ${#potentials[*]} different potentials:\n\tThis may take a while...\n" >> $logFile

    # make suitable for R
    superDistanceFiles=( `echo "\"${superDistanceFiles[*]}\"\" | sed 's/ /", "/g'` )
    potentials=( `echo "\"${potentials[*]}\"\" | sed 's/ /", "/g'` )

    # Create density plots
    echo "# Started by $USER for ID $identifier (as of `date`)"

    # Variables from the preparatory shell script:

    # Take care: The order of scores and distances is not the same!

```

```

# Solution: access data by headers - it's cleaner anyway

identifier <- \"$identifier\"
topPoses <- $topPoses
workDir <- \"$workDir\"
distanceFolder <- \"$distanceFolder\"
scoreFile <- \"$scoreFile\"
potentials <- c(${potentials[*]})
aroSteps <- unique(sub(\"_.*\", \"\", potentials))
metSteps <- unique(sub(\".*_\", \"\", potentials))
superDistanceFiles <- gsub(\".*\\/\", \"\", system(paste(\"ls \", distanceFolder,
\"/super*.txt\", sep=\"\"), intern=T))
interactions <- gsub(\"_\", \" \", gsub(\"_\", \" vs. \", sub(\".txt\", \"\",
sub(\"super_dist_\", \"\", superDistanceFiles))))
relevantScoreParts <- c(\"TOTAL_SCORE\", \"PLPparthbond\", \"PLPpartsteric\",
\"PLPpartmetal\", \"PLPpartrepulsive\", \"PLPpartburpolar\", \"TRIPOS_LJ\", \"TRIPOS_TORS\",
\"CHEMPLP_PLP_PROT\", \"TRIPOS_TORS_PROT\", \"CHEMPLP_CLASH2_PROT\",
\"ATOMS_OUTSIDE_BINDINGSITE\")
featureFileName <- \"$featureFileName\"
graphicsFolder <- \"$graphicsFolder\"

# Use only relevant score parts
# * TOTAL_SCORE: scoring function value obtained during docking
# * PLPparthbond: PLP hbond score
# * PLPpartsteric: PLP steric contact score
# * PLPpartmetal: PLP metal interaction score
# * PLPpartrepulsive: PLP donor/donor and acceptor/acceptor repulsion score
# * PLPpartburpolar: PLP buried polar atoms score (polar atoms occluded by nonpolar ones)
# * TRIPOS_LJ: (deduced: contact potential)
# * TRIPOS_TORS: intra-ligand torsion score
# * CHEMPLP_PLP_PROT: (deduced: intra-protein score, only calculated for i-,ex. side-chains)
# * TRIPOS_TORS_PROT: (deduced: protein torsion potential)
# * CHEMPLP_CLASH2_PROT: (deduced: intra-ligand clash score)
# * ATOMS_OUTSIDE_BINDINGSITE: number of ligand atoms outside binding site

# Read in all distances
distances <- array(NA, c(length(interactions), topPoses, length(potentials)),
list(interactions, 1:topPoses, potentials))
for (distanceFile in paste(distanceFolder, \"/\", superDistanceFiles, sep = \"\")) {
  interaction <- gsub(\"_\", \" \", gsub(\"_\", \" vs. \", sub(\".txt\", \"\",
sub(\".*super_dist_\", \"\", distanceFile))))
  print(paste(\"Reading\", distanceFile))
  readInMatrix <- read.table(distanceFile, header = TRUE)[1:topPoses, ]
  colnames(readInMatrix) <- sub(\"_[.]\", \"_-\", sub(\"^[.]\", \"-\", sub(\"X\", \"\",
colnames(readInMatrix))))
  for (poseNumber in 1:topPoses) for (potential in potentials) distances[interaction,
poseNumber, potential] <- readInMatrix[poseNumber, potential]
}

# Read in all scores (relevant score parts)
scores <- array(NA, c(length(relevantScoreParts), topPoses, length(potentials)),
list(relevantScoreParts, 1:topPoses, potentials))
for (potential in potentials) {
  featureFile <- paste(\"${workDir}/docked_\", potential, \"/\", featureFileName, sep = \"\")
  print(paste(\"Reading\", featureFile))
  readInMatrix <- read.table(featureFile, header = TRUE, sep=\",\") [1:topPoses,
relevantScoreParts]
  for (relevantScorePart in relevantScoreParts) for (poseNumber in 1:topPoses)
scores[relevantScorePart, poseNumber, potential] <- readInMatrix[poseNumber,
relevantScorePart]
}

# Function for producing a barplot depiction of fractions of values in a certain range
intervalCountBarplot <- function(myMatrix, mainTitle, minimum, maximum, yLabel, fileName,
deviceName = \"none\") {
  myCex <- 1.0
  bottomMargin <- 15
  leftMargin <- 5
  yLabelLine <- 3
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep=\",\")
    pdf(saveFile, 10, 10)
    myCex <- 1.2
    bottomMargin <- 20
  } else

```

```

if (deviceName == \"png\") {
  saveFile <- paste(fileName, \".png\", sep=\"\")
  png(saveFile, width = 1000, height = 1000)
  myCex <- 1.2
  bottomMargin <- 20
}
par(mar = c(bottomMargin, leftMargin, 4, 2))
barplot(colSums((myMatrix > minimum) + (myMatrix < maximum) == 2) / dim(myMatrix)[1] * 100,
names.arg = colnames(myMatrix), main = mainTitle, cex.main = myCex, col=\"gray\", border = NA,
las = 2, cex.axis = myCex * 0.8, cex.names = myCex * 0.8)
title(ylab = yLabel, line = yLabelLine, cex.lab = myCex)
if (deviceName != \"none\") dev.off()
}

# Function for producing a heatmap depiction of fractions of values in a certain range
matrixMap <- function(myMatrix, mainTitle, legLabel, fileName, deviceName = \"none\") {

  topMargin <- 4
  leftMargin <- 20
  myCex <- 1.0
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep=\"\")
    pdfWidth <- if (dim(myMatrix)[2] > 20) 15 else 10
    pdf(saveFile, pdfWidth, 10)
    myCex <- 1.5
    leftMargin <- 30
  } else
  if (deviceName == \"png\") {
    saveFile <- paste(fileName, \".png\", sep=\"\")
    png(saveFile, width = 1000, height = 1000)
    myCex <- 1.2
    topMargin <- 3
    leftMargin <- 23
  }

  layout(matrix(c(1,2,1,3), 2, 2), width = c(6,1), height = c(1,10))

  # plot headline
  par(mar = rep(0,4))
  plot(1, type = 'n', axes = FALSE, xlab = \"\", ylab = \"\")
  title(main = mainTitle, line = -2, cex.main = 2.0)

  # plot heatmap
  par(mar=c(1, leftMargin, topMargin, 0.2))
  image(1:dim(myMatrix)[2], 1:dim(myMatrix)[1], t(myMatrix), xlab = \"\", ylab = \"\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors), axes = FALSE)
  axis(2, at = 1:dim(myMatrix)[1], labels = rownames(myMatrix), las = 2, cex.axis = myCex)
  axis(3, at = 1:dim(myMatrix)[2], labels = colnames(myMatrix), las = 2, cex.axis = myCex)
  # title(main = mainTitle, line = 5)
  box()

  # plot legend
  par(mar=c(1, 4, topMargin, 0.2))
  image(1, 1:100, matrix(1:100, 1, 100), xlab = \"\", ylab = \"\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE, cex.lab = myCex * 0.8)
  title(ylab = legLabel, line = 2.5, cex.lab = myCex * 0.8)
  axis(2, at = (0:10) * 10, las = 1, cex.axis = myCex * 0.8)
  box()

  if (deviceName != \"none\") dev.off()
}

minimum <- 3.4
maximum <- 5.5
# For all potentials, plot all distances within range
for (potential in potentials) {
  mainTitle <- paste(\"Interactions using \", potential, \" (\", identifier, \", top \",
topPoses, \")\", sep = \"\")
  yLabel <- paste(\"% of interactions of \", topPoses, \" poses within (\", minimum, \", \",
maximum, \") \", sep = \"\")
  fileName <- paste(graphicsFolder, \"/\", \"perc_\", minimum, \"_\", maximum, \"_\",
potential, \"_top\", topPoses, sep = \"\")
  intervalCountBarplot(t(distances[, , potential]), mainTitle, minimum, maximum, yLabel,
fileName, deviceName=\"pdf\")
  intervalCountBarplot(t(distances[, , potential]), mainTitle, minimum, maximum, yLabel,
fileName, deviceName=\"png\")
}

```



```

scoreFile=${workDir}/"scores_sum.txt"
featureFileName="features.csv"

if [ ! -f $scoreFile ] ; then
    echo "$scoreFile could not be found - abort!"
    echo "$scoreFile could not be found - abort!" >> $logFile
    exit
fi
if [ ! -d $workDir ] ; then
    echo "The necessary folder $workDir could not be found - abort!"
    exit 1
fi
if [ ! -d $distanceFolder ] ; then
    echo "The necessary folder $distanceFolder could not be found - abort!"
    exit 1
fi
if [ ! -f $transformerLog ] ; then
    echo "Transformer log file $transformerLog was not found - abort!"
    exit 1
fi

# Inform user about what will happen, log setup information
echo "$USER started evaTransform_e for identifier $identifier on `date`."
echo "$USER started evaTransform_e for identifier $identifier on `date`." > $logFile

# Get minimum common number of docking poses
cluster=`grep "cluster" $transformerLog | wc -l`
nos=( `grep "Will try to analyze" $transformerLog | awk '{print $5}'` )
if [ ! $topPoses ] ; then
    topPoses=${nos[0]}
fi
if [ $cluster -gt 0 ] ; then
    printf "Docking was performed using the cluster option."
    printf "Docking was performed using the cluster option." >> $logFile
    for no in ${nos[*]}
    do
        if [ $no -lt $topPoses ] ; then
            topPoses=$no
        fi
    done
else
    printf "Independent docking (no clustering) was performed."
    printf "Independent docking (no clustering) was performed." >> $logFile
fi
echo " Minimum common number of poses: $topPoses."
echo " Minimum common number of poses: $topPoses." >> $logFile

aroSteps=( `grep "Potential minima for C.cat-Aro interaction:" $transformerLog | sed
's/.*/'` )
metSteps=( `grep "Potential minima for C.cat-MethioninS interaction:" $transformerLog | sed
's/.*/'` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        potentials=( ${potentials[*]} ${aro}_${met} )
    done
done

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile

mkdir -p $graphicsFolder

superDistanceFiles=( `find ${distanceFolder}/ *.txt | sed 's/.*/' | grep "^super_dist_"` )
printf "\tAnalyzing data for ${#superDistanceFiles[*]} measured interactions and
${#potentials[*]} different potentials.\n\tThis may take a while...\n"
printf "\tAnalyzing data for ${#superDistanceFiles[*]} measured interactions and
${#potentials[*]} different potentials:\n\tThis may take a while...\n" >> $logFile

# make suitable for R
superDistanceFiles=( `echo "\"${superDistanceFiles[*]}\"" | sed 's/ /", "/g'` )
potentials=( `echo "\"${potentials[*]}\"" | sed 's/ /", "/g'` )

```

```

# Create density plots
echo "# Started by $USER for ID $identifier (as of `date`)"

# Variables from the preparatory shell script:

# Take care: The order of scores and distances is not the same!
# Solution: access data by headers - it's cleaner anyway

identifier <- \"$identifier\"
topPoses <- $topPoses
workDir <- \"$workDir\"
distanceFolder <- \"$distanceFolder\"
scoreFile <- \"$scoreFile\"
potentials <- c(${potentials[*]})
aroSteps <- unique(sub(\"_.*\", \"\", potentials))
metSteps <- unique(sub(\".*_\", \"\", potentials))
# superDistanceFiles <- c(${superDistanceFiles[*]})
minimum <- 3.4
maximum <- 5.5

superDistanceFiles <- gsub(\".*\\/\", \"\", system(paste(\"ls \", distanceFolder,
\"/super*.txt\", sep=\"\\\"), intern=T))

interactions <- sub(\".txt\", \"\", sub(\"super_dist_\", \"\", superDistanceFiles))
relevantScoreParts <- c(\"TOTAL_SCORE\", \"PLPparthbond\", \"PLPpartsteric\",
\"PLPpartmetal\", \"PLPpartrepulsive\", \"PLPpartburpolar\", \"TRIPOS_LJ\", \"TRIPOS_TORS\",
\"CHEMPLP_PLP_PROT\", \"TRIPOS_TORS_PROT\", \"CHEMPLP_CLASH2_PROT\",
\"ATOMS_OUTSIDE_BINDINGSITE\")
featureFileName <- \"$featureFileName\"
graphicsFolder <- \"$graphicsFolder\"

# Use only relevant score parts
# * TOTAL_SCORE: scoring function value obtained during docking
# * PLPparthbond: PLP hbond score
# * PLPpartsteric: PLP steric contact score
# * PLPpartmetal: PLP metal interaction score
# * PLPpartrepulsive: PLP donor/donor and acceptor/acceptor repulsion score
# * PLPpartburpolar: PLP buried polar atoms score (polar atoms occluded by nonpolar ones)
# * TRIPOS_LJ: (deduced: contact potential)
# * TRIPOS_TORS: intra-ligand torsion score
# * CHEMPLP_PLP_PROT: (deduced: intra-protein score, only calculated for i-,ex. side-chains)
# * TRIPOS_TORS_PROT: (deduced: protein torsion potential)
# * CHEMPLP_CLASH2_PROT: (deduced: intra-ligand clash score)
# * ATOMS_OUTSIDE_BINDINGSITE: number of ligand atoms outside binding site

print(\"Reading in distances...\")
# Read in all distances
distances <- array(NA, c(length(interactions), topPoses, length(potentials)),
list(interactions, 1:topPoses, potentials))
for (distanceFile in paste(distanceFolder, \"/\", superDistanceFiles, sep = \"\\\")) {
  interaction <- sub(\".txt\", \"\", sub(\".*super_dist_\", \"\", distanceFile))
  print(distanceFile)
  readInMatrix <- read.table(distanceFile, header = TRUE)[1:topPoses, ]
  if (length(potentials) > 1) {
    colnames(readInMatrix) <- sub(\"_.[.]\", \"_-\", sub(\"^[.]\", \"-\", sub(\"X\", \"\",
colnames(readInMatrix))))
    for (poseNumber in 1:topPoses) for (potential in potentials) distances[interaction,
poseNumber, potential] <- readInMatrix[poseNumber, potential]
  } else for (poseNumber in 1:topPoses) distances[interaction, poseNumber, 1] <-
readInMatrix[poseNumber]
}

print(\"Reading in scores...\")
# Read in all scores (relevant score parts)
scores <- array(NA, c(length(relevantScoreParts), topPoses, length(potentials)),
list(relevantScoreParts, 1:topPoses, potentials))
for (potential in potentials) {
  featureFile <- paste(\"${workDir}/docked_\", potential, \"/\", featureFileName, sep = \"\\\"")
  readInMatrix <- read.table(featureFile, header = TRUE, sep=\",\") [1:topPoses,
relevantScoreParts]
  for (relevantScorePart in relevantScoreParts) for (poseNumber in 1:topPoses)
scores[relevantScorePart, poseNumber, potential] <- readInMatrix[poseNumber,
relevantScorePart]
}

```

```

print("\n - done.\n")

# Function for producing a composition of plots to compare data sets:
#   a multiple density plot
#   a multiple boxplot
#   a heatmap depiction of Kolmogorov-Smirnov-Test

comparePlot <- function(myMatrix, mainTitle, xLabel, fileName, deviceName = \"none\") {

  # Resulting picture is composed of five high-level plots

  # Adjust some margins and font sizes for different output
  left3 <- 15.4
  left4 <- 5.2
  line4 <- 3.5
  left5 <- 10
  myCex <- 1.2
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep=\"\")
    pdf(saveFile, 10, 15)
    myCex <- 1.5
    left3 <- 21
    left5 <- 12.6
  }
  else if (deviceName == \"png\") {
    saveFile <- paste(fileName, \".png\", sep=\"\")
    png(saveFile, width = 1000, height = 1500)
    left3 <- 21.2
    left4 <- 7.2
    line4 <- 4.5
    left5 <- 12.7
    myCex <- 2
  }
  else if (deviceName == \"jpg\") {
    saveFile <- paste(fileName, \".jpg\", sep=\"\")
    bitmap(saveFile, type = \"jpeg\", height = 15, width = 10)
    left3 <- 17.6
    left4 <- 6
    line4 <- 4
    left5 <- 10.9
    myCex <- 1.5
  }
}

# PREPARATION: layout
layout(matrix(c(1, 3, 4, 1, 3, 5, 2, 3, 5), 3, 3), heights=c(3, 2, 3), widths=c(1, 1, 7))

# PART 1: multiple density plot

# legend first
par(mar=c(0, 0, 0, 0))
plot.new()
legend(\"center\", rev(colnames(myMatrix)), col = rev(rainbow(dim(myMatrix)[2])), ncol = 1,
lwd = 3, cex = myCex)

# Calculate plot margins
minX <- min(density(myMatrix[, 1])\$x)
if (dim(myMatrix)[2] > 1) for (j in 2:dim(myMatrix)[2]) if (min(density(myMatrix[, j])\$x) <
minX) minX <- min(density(myMatrix[, j])\$x)
maxX <- max(density(myMatrix[, 1])\$x)
if (dim(myMatrix)[2] > 1) for (j in 2:dim(myMatrix)[2]) if (max(density(myMatrix[, j])\$x) >
maxX) maxX <- max(density(myMatrix[, j])\$x)

maxY <- max(density(myMatrix[, 1])\$y)
if (dim(myMatrix)[2] > 1) for (j in 2:dim(myMatrix)[2]) if (max(density(myMatrix[, j])\$y) >
maxY) maxY <- max(density(myMatrix[, j])\$y)

par(mar=c(2.5, 4.2, 4, 1))
plot(1, xlim = c(minX, maxX), ylim = c(0, maxY * 1.05), type = 'n', xlab = \"\", ylab =
\"\", cex.axis = myCex, yaxs = \"i\")
for (i in 1:dim(myMatrix)[2]) lines(density(myMatrix[, i])\$x, density(myMatrix[, i])\$y,
col = rainbow(dim(myMatrix)[2])[i], lwd = 2.5)
title(mainTitle, cex.main = 1.3 * myCex, ylab = paste(\"Data densities (\",
dim(myMatrix)[1], \" obs. each)\", sep = \"\"), cex.lab = myCex)
box()

# PART 2: multiple box plot

```

```

par(mar=c(5, left3, 0, 1))
boxplot(myMatrix ~ col(myMatrix), ylim = c(minX, maxX), horizontal = TRUE, axes = FALSE,
xlab = xLabel, col = rainbow(dim(myMatrix)[2]), cex.lab = myCex)
axis(1, cex.axis = myCex)
axis(2, at = 1:dim(myMatrix)[2], labels = colnames(myMatrix), las = 2, cex.axis = myCex)
box()

# PART 3: Cramer-von Mises test p-value matrix

# Create the Kolmogorov-Smirnov-p-value matrix
corMatrix <- matrix(rep(0, dim(myMatrix)[2]^2), dim(myMatrix)[2], dim(myMatrix)[2])
for (i in 1:dim(myMatrix)[2]) for (j in i:dim(myMatrix)[2]) if (i != j) corMatrix[i, j] <-
ks.test(myMatrix[, i], myMatrix[, j], exact = T)$p.value
colnames(corMatrix) <- colnames(myMatrix)

# plot a colorful legend
par(mar=c(1, left4, 11, 0))
# par(mar=c(1,4,9,0)+0.1)
image(1, 1:100, matrix(1:100, 1, 100), xlab = "\", ylab = \"Kolmogorov-Smirnov p-value\",
col = colorRampPalette(c(\"white\", \"red\"))(100), axes = FALSE, cex.lab = myCex * 0.8, mgp =
c(line4, 1, 0))
axis(2, at = seq(1, 100, length.out = 9), labels = round(seq(0, 1, length.out=9), 2), las =
1, cex.axis = myCex * 0.8)
box()

# interpolate colors and extract from complete palette
NOcolors <- 1000
minIndex <- min(round((1+NOcolors)/2 + corMatrix*((NOcolors+1)/2)-1))
maxIndex <- max(round((1+NOcolors)/2 + corMatrix*((NOcolors+1)/2)-1))

par(mar=c(1, left5, 11, 1))
# par(mar=c(1,6,9,0)+0.1)
image(1:dim(corMatrix)[1], 1:dim(corMatrix)[1], corMatrix, xlab = "\", ylab = "\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
title(\"Comparison of distributions\", line = 9, cex.main = myCex)
axis(2, at = 1:dim(corMatrix)[1], labels = colnames(corMatrix), las = 2, cex.axis = myCex)
axis(3, at = 1:dim(corMatrix)[1], labels = colnames(corMatrix), las = 2, cex.axis = myCex)

for (i in 1:dim(corMatrix)[1]) lines(c(0, i + 0.5), c(i + 0.5, i + 0.5))
for (i in 0:(dim(corMatrix)[1] - 1)) lines(c(i + 0.5, i + 0.5), c(i + 0.5, dim(corMatrix)[1]
+ 0.5))
for (i in 1:dim(corMatrix)[1]) for (j in i:dim(corMatrix)[1]) if (i != j) text(i,j,
round(corMatrix[i, j], 2), cex = myCex * 0.8)

if (deviceName != \"none\") dev.off()
}

# Function for producing plots of two entities over ordinate
# (Plot will have two y axes, x axis is simply ordinal)
# requires my2Cmatrix to be a two column matrix (the number of rows will be the x axis)
rankDistancePlot <- function(my2Cmatrix, mainTitle, xLabel, interval, fileName, deviceName =
\"none\", myColors = c(\"green\", \"red\")) {

# Adjust some margins and font sizes for different output
headlineCex <- 0.9
axisCex <- 0.9
leftMargin <- 4
rightMargin <- 4
rightLine <- 3
if (deviceName == \"pdf\") {
saveFile <- paste(fileName, \".pdf\", sep=\"\")
pdf(saveFile)
}
else if (deviceName == \"png\") {
saveFile <- paste(fileName, \".png\", sep=\"\")
png(saveFile, width = 1000, height = 1000)
headlineCex <- 1.5
axisCex <- 1.5
leftMargin <- 5
rightMargin <- 6
rightLine <- 4
}
else if (deviceName == \"jpg\") {
saveFile <- paste(fileName, \".jpg\", sep=\"\")
bitmap(saveFile, type = \"jpeg\", height = 10, width = 10)
headlineCex <- 1.2
}
}

```



```

axisCex <- 1.2
leftMargin <- 4.5
rightMargin <- 5
rightLine <- 4
}

layout(matrix(1:2), height = c(6,1))

# Need to normalize 2nd column to fit into 1st column space
y2range <- range(my2Cmatrix[, 2])
y2Scaled <- my2Cmatrix[, 2] / diff(y2range) * diff(range(my2Cmatrix[, 1]))
minY2 <- min(y2Scaled)
y2Scaled <- y2Scaled - (minY2 - min(my2Cmatrix[, 1]))

# Plot with two y axes
par(mar = c(5, leftMargin, 2, rightMargin) + 0.1)
plot(my2Cmatrix[, 1], col = myColors[1], pch = 20, xlab = xLabel, ylab =
colnames(my2Cmatrix)[1], axes = FALSE, cex.axis = axisCex, cex.lab = axisCex)
lines(c(1, dim(my2Cmatrix)[1]), rep(interval[1], 2), lwd = 2, lty = 2)
lines(c(1, dim(my2Cmatrix)[1]), rep(interval[2], 2), lwd = 2, lty = 2)
title(main = mainTitle, cex.main = headlineCex)
points(y2Scaled, col = myColors[2], pch = 20)

# axis below
axis(1, cex.axis = axisCex)
# left y axis
axis(2, las = 1, cex.axis = axisCex)
mtext(colnames(my2Cmatrix)[2], 4, line = rightLine, cex = axisCex)
# right y axis
axis(4, las = 1, at = round(seq(min(my2Cmatrix[, 1]), max(my2Cmatrix[, 1]), length.out = 5),
2), labels = signif(seq(y2range[1], y2range[2], length.out = 5), 2), cex.axis = axisCex)

box()

# Plot legend below
par(mar=c(0, 0, 0, 0))
plot.new()
legend("center", c(colnames(my2Cmatrix), paste("\Spearman rho =", round(cor(1:topPoses,
my2Cmatrix[, 1], method = 'spearman'), 2))), paste("\Spearman rho =", round(cor(1:topPoses,
my2Cmatrix[, 2], method = 'spearman'), 2))), col = c(myColors, "\black", "\black"), ncol =
2, cex = headlineCex, pch = c(20, 20, NA, NA), title = "\Spearman rank correlation")

if (deviceName != "none") dev.off()
}

# Function for producing a heatmap depiction of values from [0,1]
# in a matrix as percentages (from 0 = white to 1 = black)
matrixPercMap <- function(myMatrix, mainTitle, legLabel, fileName, deviceName = "none") {

if(sum((myMatrix <= 1) + (myMatrix >= 0)) != dim(myMatrix)[1] * dim(myMatrix)[2] * 2)
stop("\matrixPercMap: Matrix contains elements not in [0, 1] - abort!\")

topMargin <- 4
leftMargin <- 20
leftMargin2 <- 4
leftLine2 <- 2.5

myCex <- 1.0
if (deviceName == "pdf") {
saveFile <- paste(fileName, ".pdf", sep="\")
pdf(saveFile, if (dim(myMatrix)[2] > 20) 15 else 10, 10)
myCex <- 1.5
leftMargin <- 30
leftMargin2 <- 6
leftLine2 <- 4.5
} else
if (deviceName == "png") {
saveFile <- paste(fileName, ".png", sep="\")
png(saveFile, width = 1000, height = 1000)
myCex <- 1.2
topMargin <- 3
leftMargin <- 23
leftMargin2 <- 5
leftLine2 <- 3.5
} else if (deviceName == "jpg") {

```

```

saveFile <- paste(fileName, \".jpg\", sep=\\")
bitmap(saveFile, type = \"jpeg\", height = 10, width = 10)
myCex <- 1.2
leftMargin <- 23
leftMargin2 <- 5
leftLine2 <- 3.5
}

layout(matrix(c(1,2,1,3), 2, 2), width = c(6,1), height = c(1,10))

# plot headline
par(mar = rep(0,4))
plot(1, type = 'n', axes = FALSE, xlab = \\\", ylab = \\")
title(main = mainTitle, line = -2, cex.main = 2.0)
print(myMatrix)
# plot heatmap
par(mar=c(1, leftMargin, topMargin, 0.2))
image(1:dim(myMatrix)[2], 1:dim(myMatrix)[1], t(myMatrix), xlab = \\\", ylab = \\\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE)
axis(2, at = 1:dim(myMatrix)[1], labels = rownames(myMatrix), las = 2, cex.axis = myCex)
axis(3, at = 1:dim(myMatrix)[2], labels = colnames(myMatrix), las = 2, cex.axis = myCex)
# title(main = mainTitle, line = 5)
box()

# plot legend
par(mar=c(1, leftMargin2, topMargin, 0.2))
image(1, 1:100, matrix(1:100, 1, 100), xlab = \\\", ylab = \\\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(100), axes = FALSE, cex.lab = myCex * 0.8)
title(ylab = legLabel, line = leftLine2, cex.lab = myCex * 0.8)
axis(2, at = seq(1, 100, length.out = 9), labels = seq(0, 100, length.out = 9), las = 1,
cex.axis = myCex * 0.8)
box()

if (deviceName != \"none\") dev.off()
}

# Function for producing a heatmap depiction of values in a matrix
# over a given interval (intervalMin = green, mean = white, intervalMax = red),
# colors will be scaled accordingly.
# All matrix values must lie within the given interval.
matrixFullMap <- function(myMatrix, mainTitle, legLabel, fileName, intervalMin, intervalMax,
deviceName = \"none\") {

  if (intervalMin >= intervalMax) stop(\"matrixFullMap: Given interval is not valid -
abort!\")
#   if (sum((myMatrix <= intervalMax) + (myMatrix >= intervalMin)) != dim(myMatrix)[1] *
dim(myMatrix)[2]* 2) stop(\"matrixFullMap: Matrix contains elements not in the given interval
- abort!\")

  topMargin <- 4
  leftMargin <- 20
  leftMargin2 <- 4
  leftLine2 <- 2.5

  myCex <- 1.0
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep=\\")
    pdf(saveFile, if (dim(myMatrix)[2] > 20) 15 else 10, 10)
    myCex <- 1.5
    leftMargin <- 30
    leftMargin2 <- 6
    leftLine2 <- 4.5
  } else
  if (deviceName == \"png\") {
    saveFile <- paste(fileName, \".png\", sep=\\")
    png(saveFile, width = 1000, height = 1000)
    myCex <- 1.2
    topMargin <- 3
    leftMargin <- 23
    leftMargin2 <- 5
    leftLine2 <- 3.5
  } else if (deviceName == \"jpg\") {
    saveFile <- paste(fileName, \".jpg\", sep=\\")
    bitmap(saveFile, type = \"jpeg\", height = 10, width = 10)
    myCex <- 1.2
    leftMargin <- 23

```

```

    leftMargin2 <- 5
    leftLine2 <- 3.5
  }

  layout(matrix(c(1,2,1,3), 2, 2), width = c(6,1), height = c(1,10))

  # plot headline
  par(mar = rep(0, 4))
  plot(1, type = 'n', axes = FALSE, xlab = "\"", ylab = "\"")
  title(main = mainTitle, line = -2, cex.main = 2.0)

  # interpolate colors and extract from complete palette
  NOcolors <- 1000
  minPart <- diff(range(intervalMin, min(myMatrix))) / (intervalMax - intervalMin)
  maxPart <- diff(range(intervalMin, max(myMatrix))) / (intervalMax - intervalMin)
  minIndex <- round(minPart * NOcolors)
  maxIndex <- round(maxPart * NOcolors)

  # plot heatmap
  par(mar=c(1, leftMargin, topMargin, 0.2))
  image(1:dim(myMatrix)[2], 1:dim(myMatrix)[1], t(myMatrix), xlab = "\"", ylab = "\"", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
  axis(2, at = 1:dim(myMatrix)[1], labels = rownames(myMatrix), las = 2, cex.axis = myCex)
  axis(3, at = 1:dim(myMatrix)[2], labels = colnames(myMatrix), las = 2, cex.axis = myCex)
  # title(main = mainTitle, line = 5)
  box()

  # plot legend
  par(mar=c(1, leftMargin2, topMargin, 0.2))
  image(1, 1:100, matrix(1:100, 1, 100), xlab = "\"", ylab = "\"", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors), axes = FALSE, cex.lab = myCex *
0.8)
  title(ylab = legLabel, line = leftLine2, cex.lab = myCex * 0.8)
  axis(2, at = seq(1, 100, length.out = 9), labels = round(seq(intervalMin, intervalMax,
length.out = 9), 2), las = 1, cex.axis = myCex * 0.8)
  box()

  if (deviceName != \"none\") dev.off()
}

# Function for producing a heatmap depiction of values in a matrix
# over a given interval (intervalMin = green, mean = white, intervalMax = red),
# colors will be scaled accordingly.
# Matrix values will not be checked, the numerical values will be depicted as strings
matrixValueMap <- function(myMatrix, mainTitle, legLabel, fileName, intervalMin, intervalMax,
deviceName = \"none\") {

  if (intervalMin >= intervalMax) stop(\"matrixFullMap: Given interval is not valid -
abort!\")

  topMargin <- 4
  leftMargin <- 20
  leftMargin2 <- 4
  leftLine2 <- 2.5

  myCex <- 1.0
  if (deviceName == \"pdf\") {
    saveFile <- paste(fileName, \".pdf\", sep = \"\")
    pdf(saveFile, if (dim(myMatrix)[2] > 20) 15 else 10, 10)
    myCex <- 1.5
    leftMargin <- 30
    leftMargin2 <- 6
    leftLine2 <- 4.5
  } else
  if (deviceName == \"png\") {
    saveFile <- paste(fileName, \".png\", sep = \"\")
    png(saveFile, width = 1000, height = 1000)
    myCex <- 1.2
    topMargin <- 3
    leftMargin <- 23
    leftMargin2 <- 5
    leftLine2 <- 3.5
  } else if (deviceName == \"jpg\") {

```

```

saveFile <- paste(fileName, \".jpg\", sep=\\")
bitmap(saveFile, type = \"jpeg\", height = 10, width = 10)
myCex <- 1.2
leftMargin <- 23
leftMargin2 <- 5
leftLine2 <- 3.5
}

layout(matrix(c(1,2,1,3), 2, 2), width = c(6,1), height = c(1,10))

# plot headline
par(mar = rep(0, 4))
plot(1, type = 'n', axes = FALSE, xlab = \\\", ylab = \\")
title(main = mainTitle, line = -2, cex.main = 2.0)

# interpolate colors and extract from complete palette
NOcolors <- 1000
minPart <- diff(range(intervalMin, min(myMatrix))) / (intervalMax - intervalMin)
maxPart <- diff(range(intervalMin, max(myMatrix))) / (intervalMax - intervalMin)
minIndex <- round(minPart * NOcolors)
maxIndex <- round(maxPart * NOcolors)

# plot heatmap
par(mar=c(1, leftMargin, topMargin, 0.2))
image(1:dim(myMatrix)[2], 1:dim(myMatrix)[1], t(myMatrix), xlab = \\\", ylab = \\\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors)[minIndex:maxIndex], axes = FALSE)
axis(2, at = 1:dim(myMatrix)[1], labels = rownames(myMatrix), las = 2, cex.axis = myCex)
axis(3, at = 1:dim(myMatrix)[2], labels = colnames(myMatrix), las = 2, cex.axis = myCex)
for (i in 1:dim(myMatrix)[1]) for (j in 1:dim(myMatrix)[2]) text(j, i, round(myMatrix[i, j],
2), cex = myCex * 0.8)
box()

# plot legend
par(mar=c(1, leftMargin2, topMargin, 0.2))
image(1, 1:100, matrix(1:100, 1, 100), xlab = \\\", ylab = \\\", col =
colorRampPalette(c(\"green\", \"white\", \"red\"))(NOcolors), axes = FALSE, cex.lab = myCex *
0.8)
title(ylab = legLabel, line = leftLine2, cex.lab = myCex * 0.8)
axis(2, at = seq(1, 100, length.out = 9), labels = round(seq(intervalMin, intervalMax,
length.out = 9), 2), las = 1, cex.axis = myCex * 0.8)
box()

if (deviceName != \"none\") dev.off()
}

#
# # Create a matrix of percentages of score-relevance for all potentials
# myMatrix <- matrix(0, length(interactions), length(potentials), dimnames =
list(interactions, potentials))
# for (interaction in interactions) for (potential in potentials) myMatrix[interaction,
potential] <- sum((distances[interaction, 1:topPoses, potential] > minimum) +
(distances[interaction, 1:topPoses, potential] < maximum) == 2) / topPoses
#
# # assign nice interaction names
# # if (length(potentials) == 1)
# rownames(myMatrix) <- gsub(\"_\", \" \", sub(\"_\", \" vs. \", interactions))
# mainTitle <- paste(\"Score-relevance of interactions (\", identifier, \")\", sep = \\")
# legLabel <- paste(\"% of score-relevant poses within (\", minimum, \", \", maximum, \")
Å...\", sep = \\")
# fileName <- paste(graphicsFolder, \"/e_relevance_top\", topPoses, sep = \\")
#
# # matrixPercMap(myMatrix, mainTitle, legLabel, fileName, deviceName = \"pdf\")
# matrixPercMap(myMatrix, mainTitle, legLabel, fileName, deviceName = \"png\")
# # matrixPercMap(myMatrix, mainTitle, legLabel, fileName, deviceName = \"jpg\")
#

# Create a matrix of rank correlation values for distances and PLPpartsteric for each
potential

```

```

myMatrix <- matrix(0, length(interactions) + 1, length(potentials), dimnames =
list(c(interactions, "PLPpartsteric"), potentials))
for (interaction in interactions) for (potential in potentials) myMatrix[interaction,
potential] <- cor(1:topPoses, distances[interaction, 1:topPoses, potential], method =
'spearman')
# Fill last line with PLPpartsteric rank correlations
for (potential in potentials) myMatrix["PLPpartsteric", potential] <- cor(1:topPoses,
scores["PLPpartsteric", 1:topPoses, potential], method = 'spearman')
# assign nice interaction names
rownames(myMatrix) <- c(gsub("_", " ", sub("__", " vs. ", interactions)),
"PLPpartsteric")
# mainTitle <- paste("Rank correlations (", identifier, ") ", sep = "\ ")
# fileName <- paste(graphicsFolder, "/e_rank_overview_top", topPoses, sep = "\ ")
#
# matrixFullMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "pdf")
# matrixFullMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "png")
# matrixFullMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "jpg")
#
# fileName <- paste(graphicsFolder, "/e_rank_values_top", topPoses, sep = "\ ")
#
# matrixValueMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "pdf")
# matrixValueMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "png")
# matrixValueMap(myMatrix, mainTitle, "Spearman rank correlation rho", fileName, -1, 1,
deviceName = "jpg")

# Create a matrix for correlation of parameter changes to rank correlation changes
corCorMatrix <- matrix(0, length(interactions) + 1, length(aroSteps) + length(metSteps),
dimnames = list(rownames(myMatrix), c(paste(aroSteps, "_", sep = "\ "),
paste("*_", metSteps, sep = "\ "))))

# Take care: if there are only exactly two steps of one parameters, you will get a correlation
of either -1 or +1
# variation in MET
if (length(metSteps) > 1) {
  relevantParameters <- as.numeric(unique(gsub(".*_", "\ ", potentials)))
  for (aroStep in aroSteps) {
    relevantPotentials <- potentials[grep(paste("^", aroStep, "_", sep = "\ "),
potentials)]
    for (interaction in rownames(myMatrix)) corCorMatrix[interaction, paste(aroStep, "_",
sep = "\ ")] <- cor(myMatrix[interaction, relevantPotentials], relevantParameters)
  }
}
# variation in ARO
if (length(aroSteps) > 1) {
  relevantParameters <- as.numeric(unique(gsub(".*_", "\ ", potentials)))
  for (metStep in metSteps) {
    relevantPotentials <- potentials[grep(paste("_", metStep, "$", sep = "\ "),
potentials)]
    for (interaction in rownames(myMatrix)) corCorMatrix[interaction, paste("*_", metStep,
sep = "\ ")] <- cor(myMatrix[interaction, relevantPotentials], relevantParameters)
  }
}

print(corCorMatrix)

if ((length(aroSteps) > 1) || (length(metSteps) > 1)) {
  mainTitle <- paste("Correlation of correlations with parameter changes (", identifier,
"\ ) ", sep = "\ ")
  fileName <- paste(graphicsFolder, "/e_cor_cor_top", topPoses, sep = "\ ")

# matrixFullMap(corCorMatrix, mainTitle, "Pearson correlation r", fileName, -1, 1,
deviceName = "pdf")
# matrixFullMap(corCorMatrix, mainTitle, "Pearson correlation r", fileName, -1, 1,
deviceName = "png")
# matrixFullMap(corCorMatrix, mainTitle, "Pearson correlation r", fileName, -1, 1,
deviceName = "jpg")

fileName <- paste(graphicsFolder, "/e_cor_cor_values_top", topPoses, sep = "\ ")

```

```

# matrixValueMap(corCorMatrix, mainTitle, \"Pearson correlation r\", fileName, -1, 1,
deviceName = \"pdf\")
matrixValueMap(corCorMatrix, mainTitle, \"Pearson correlation r\", fileName, -1, 1,
deviceName = \"png\")
# matrixValueMap(corCorMatrix, mainTitle, \"Pearson correlation r\", fileName, -1, 1,
deviceName = \"jpg\")
}

#
#
# print(\"PLOTTING DISTANCES AND PLPpartsteric VS. RANK\")
# # Plot \"PLPpartsteric\" and distances vs. rank
# for (potential in potentials) for (interaction in interactions) if (length(metSteps) > 1) {
#   my2Cmatrix <- matrix(c(distances[interaction, 1:topPoses, potential],
scores[\"PLPpartsteric\", 1:topPoses, potential]), topPoses, 2, dimnames = list(1:topPoses,
c(\"Distance (Å...)\", \"PLPpartsteric\")))
#   mainTitle <- paste(gsub(\"_\", \" \", sub(\"__\", \" vs. \", interaction)), \" with \",
potential, \" (\", identifier, \")\", sep = \"\")
#   xLabel <- paste(\"Rank by TOTAL SCORE\")
#   fileName <- paste(graphicsFolder, \"/e_cor_\", interaction, \"_\", potential, \"_top\",
topPoses, sep = \"\")
#
#   # rankDistancePlot(my2Cmatrix, mainTitle, xLabel, c(minimum, maximum), fileName,
deviceName=\"pdf\")
#   rankDistancePlot(my2Cmatrix, mainTitle, xLabel, c(minimum, maximum), fileName,
deviceName=\"png\")
#   rankDistancePlot(my2Cmatrix, mainTitle, xLabel, c(minimum, maximum), fileName,
deviceName=\"jpg\")
# }
# print(\"RANK CORRELATIONS DONE.\")
#
#
# print(\"PLOTTING DISTANCES\")
# # A) Plot distances over different potentials
# # comparison of changes in one parameter
# for (aroStep in aroSteps) for (interaction in interactions) if (length(metSteps) > 1) {
#   mainTitle <- paste(\"Distances with aro = \", aroStep, \" (\", identifier, \")\", sep =
\"\")
#   xLabel <- paste(\"Distance:\", gsub(\"_\", \" \", sub(\"__\", \" vs. \", interaction)),
\" (Å...) \")
#   fileName <- paste(graphicsFolder, \"/e_dist_\", interaction, \"_aro\", aroStep, \"_top\",
topPoses, sep = \"\")
#
#   # comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(\"^\", aroStep,
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"jpg\")
#   comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(\"^\", aroStep,
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"png\")
#   comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(\"^\", aroStep,
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"pdf\")
# }
# # comparison of changes in the other parameter
# for (metStep in metSteps) for (interaction in interactions) if (length(aroSteps) > 1) {
#   mainTitle <- paste(\"Distances with met = \", metStep, \" (\", identifier, \")\", sep =
\"\")
#   xLabel <- paste(\"Distance:\", gsub(\"_\", \" \", sub(\"__\", \" vs. \", interaction)),
\" (Å...) \")
#   fileName <- paste(graphicsFolder, \"/e_dist_\", interaction, \"_met\", metStep, \"_top\",
topPoses, sep = \"\")
#
#   # comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(metStep, \"$\",
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"jpg\")
#   comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(metStep, \"$\",
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"png\")
#   comparePlot(distances[interaction, 1:topPoses, potentials[grep(paste(metStep, \"$\",
sep=\"\"), potentials]]], mainTitle, xLabel, fileName, deviceName=\"pdf\")
# }
# print(\"DISTANCES DONE.\")
#
#
#
# print(\"PLOTTING SCORES\")
# # B) Plot scores \"TOTAL_SCORE\" over different potentials
# # comparison of changes in one parameter
# for (aroStep in aroSteps) if (length(metSteps) > 1) {
#   mainTitle <- paste(\"Scores with aro = \", aroStep, \" (\", identifier, \")\", sep = \"\")
#   xLabel <- paste(\"PLANTS TOTAL_SCORE (arbitrary units)\")

```



```

    for no in ${nos[*]}
    do
        if [ $no -lt $stopPoses ] ; then
            topPoses=$no
        fi
    done
    # Need the following line as remedy for an old inconsistency in naming files
    ligand=$ligandName
else
    printf "Independent docking (no clustering) was performed."
    printf "Independent docking (no clustering) was performed." >> $logFile
    # Need the following line as remedy for an old inconsistency in naming files
    ligand=`basename ${ligandFile%%.mol2}`
fi
echo " Minimum common number of poses: $stopPoses."
echo " Minimum common number of poses: $stopPoses." >> $logFile

aroSteps=( `grep "Potential minima for C.cat-Aro interaction:" $transformerLog | sed
's/.*/` )
metSteps=( `grep "Potential minima for C.cat-MethioninS interaction:" $transformerLog | sed
's/.*/` )

for aro in ${aroSteps[*]}
do
    for met in ${metSteps[*]}
    do
        potentials=( ${potentials[*]} ${aro}_${met} )
    done
done

printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n"
printf "\tC.cat - C.ar potential translations: ${aroSteps[*]}\n" >> $logFile
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n"
printf "\tC.cat - MetS potential translations: ${metSteps[*]}\n" >> $logFile

mkdir -p $graphicsFolder

# Collect necessary data into respective files
carboCationIDs=( `grep "C.cat" $ligandFile | awk '{print $1}'` )
flexResidues=( `head $transformerLog | grep "Residues flexible during docking:" | sed
's/Residues flexible during docking: //'` )
carboCationLabel=`echo "cat_${carboCationIDs[*]}" | sed 's/ /_/g'`

for potential in ${potentials[*]}
do
    printf "\t\tCollecting data from $potential..."
    printf "\t\tCollecting data from $potential..." >> $logFile
    outFile=${distanceFolder}/move_${potential}.txt
    printf "${carboCationLabel}_x\t${carboCationLabel}_y\t${carboCationLabel}_z\t" >
$outFile
    for flexResidue in ${flexResidues[*]}
    do
        printf "${flexResidue}_x\t${flexResidue}_y\t${flexResidue}_z\t" >> $outFile
    done
    printf "\n" >> $outFile

    for i in `seq 1 $stopPoses`
    do
        if [ $i -lt 10 ] ; then
            i="0$i"
        fi

        recFile="${workDir}/docked_${potential}/${ligand}_entry_00001_conf_${i}_protein.mol2"
        ligFile="${workDir}/docked_${potential}/${ligand}_entry_00001_conf_${i}.mol2"

        # if more than one carbocation atom is given, use their mean position instead
        if [ ${#carboCationIDs[*]} -gt 1 ] ; then
            $avgAtomPosScript f=${ligFile} ${carboCationIDs[*]} >> $outFile
        else
            sed -n '/@<TRIPOS>ATOM/,/@<TRIPOS>BOND/p' ${ligFile} | grep '^[
]*${carboCationIDs[*]} ' | awk '{printf "%3" "%4" "%5"}' >> $outFile
        fi
        printf "\t" >> $outFile
        # for flexible side chains, use the mean position of the side chains
        for flexResidue in ${flexResidues[*]}
        do

```

```

flexResAtomIDs=( `sed -n '/@<TRIPOS>ATOM/,/@<TRIPOS>BOND/p' $recFile |
grep "$flexResidue" | awk '{print $1}'` )
$avgAtomPosScript f=${recFile} ${flexResAtomIDs[*]} >> $outFile
printf "\t" >> $outFile
done
printf "\n" >> $outFile
done
echo " - done."
echo " - done." >> $logfile
done

printf "\tData collection done for ${#potentials[*]} different potentials.\n\tRunning R
analysis...\n"
printf "\tData collection done for ${#potentials[*]} different potentials.\n\tRunning R
analysis...\n" >> $logfile

potentials=( `echo "\"${potentials[*]}\"" | sed 's/ /, "/g'` )

# Create density plots
echo "# Started by $USER running PPQdepict for ID $identifier (as of `date`)"

# Variables from the preparatory shell script:

identifier <- "\"$identifier\""
topPoses <- $stopPoses
workDir <- "\"$workDir\""
distanceFolder <- "\"$distanceFolder\""
graphicsFolder <- "\"$graphicsFolder\""
potentials <- c(${potentials[*]})
aroSteps <- unique(sub("\".*\\"", "\"", potentials))
metSteps <- unique(sub("\".*\\"", "\"", potentials))
moveFiles <- system(paste("\ls \"", distanceFolder, \"/move*.txt\"", sep="\\"), intern=T)
kValues <- 2:6

# Function for producing an MOE pharmacophore for a given specification
# MOE pharmacophore colors
#          light      dark
# red:     f20d0d     900909
# yellow:  f2f20d     c2a10b
# green:   df20d      66106
# blue:    2b71ff     318ff
# purple:  f20df2     800080
# orange:  ff8000     cc6600

# Use with:
# myCoords:   matrix with coordinates in columns (x,y,z,x2,y2,z2 etc.)
# clusterDist: distance to distinguish between clusters
# fileName:   name of the produced .ph4 and .txt file (if less than seven clusters)
# Data will be clustered by the first three coordinates (columns)
makePharmacophores <- function(myCoords, clusterDist, fileName) {

# Available colors
colorNames <- c("red",    "yellow", "green", "blue",    "purple", "orange")
lightNames <- paste("light", colorNames, sep = "\")
darkNames <- paste("dark", colorNames, sep = "\")
lightColors <- c("f20d0d", "f2f20d", "df20d", "2b71ff", "f20df2", "ff8000")
darkColors <- c("900909", "c2a10b", "66106", "318ff", "800080", "cc6600")

# Calculate distance between cation instances
hc <- hclust(dist(myCoords[, 1:3]))
clustering <- cutree(hc, h = clusterDist)
clusterTable <- table(clustering)
NOclusters <- length(clusterTable)

if (length(clusterTable) > 6) return(NULL)
# stop(paste("Too many clusters (", NOclusters, ")within a given distance of",
clusterDist, "to sensibly visualize.))

# Prepare data structure with normalized frequencies
clusters <- data.frame(matrix(rep(0, dim(myCoords)[2] * NOclusters), NOclusters,
dim(myCoords)[2], dimnames = list(1:NOclusters, colnames(myCoords))), clusterWeight =
as.vector(clusterTable) / dim(myCoords)[1])
for (i in 1:NOclusters) {
clusters[i, seq(1, dim(myCoords)[2], 3)] <- colMeans(myCoords[which(clustering == i),
seq(1, dim(myCoords)[2], 3)])
}
}

```

```

clusters[i, seq(2, dim(myCoords)[2], 3)] <- colMeans(myCoords[which(clustering == i),
seq(2, dim(myCoords)[2], 3)])
clusters[i, seq(3, dim(myCoords)[2], 3)] <- colMeans(myCoords[which(clustering == i),
seq(3, dim(myCoords)[2], 3)])
}

sink(paste(fileName, ".ph4", sep = "\"))
cat("#moe:ph4que 2010.1\n")
cat("#pharmacophore 5 tag t value *\n")
cat("scheme t Unified matchsize i 0 title t \$ comment s \$ smask i 476156\n")
cat("#feature", NOclusters * dim(myCoords)[2] / 3, "expr tt color ix x r y r z r r
ebits ix gbits ix\n")
for (cluster in 1:NOclusters) for (entity in 0:(dim(myCoords)[2] / 3 - 1)) {
  curColor <- if (entity == 0) lightColors[cluster] else darkColors[cluster]
  cat("AtomQ", curColor, clusters[cluster, entity * 3 + 1], clusters[cluster, entity * 3
+ 2], clusters[cluster, entity * 3 + 3], clusters[cluster, dim(myCoords)[2] + 1], "0 300\n")
}
cat("#endpharmacophore\n")
sink(NULL)

# Some remarks on the colors used
sink(paste(fileName, ".txt", sep = "\"))
cat("Colors used:\n=====\n")
cat(
  "Cluster:\t", paste(1:NOclusters,
collapse = "\t"), "\n")
cat(
  "Color:\t", paste(colorNames[1:NOclusters],
collapse = "\t"), "\n")
cat(sub("_x", "", colnames(myCoords)[1]), ":\t", paste(lightNames[1:NOclusters],
collapse = "\t"), "\n")
cat(
  "equals:\t", paste(lightColors[1:NOclusters],
collapse = "\t"), "\n")
cat(
  "Others:\t", paste(darkNames[1:NOclusters],
collapse = "\t"), "\n")
cat(
  "equals:\t", paste(darkColors[1:NOclusters],
collapse = "\t"), "\n")
cat("Flexible residues:\t", paste(sub("_x", "", colnames(myCoords)[seq(4,
dim(myCoords)[2], 3)]), collapse = "\t"), "\n\n\n")
print(clusters)
sink(NULL)
}

# Function for producing an MOE pharmacophore for a given specification
# MOE pharmacophore colors
# light dark
# red: f20d0d 900909
# yellow: f2f20d c2a10b
# green: df20d 66106
# blue: 2b71ff 318ff
# purple: f20df2 800080
# orange: ff8000 cc6600

# Use with:
# myCoords: matrix with coordinates in columns (x,y,z,x2,y2,z2 etc.)
# fileName: name of the produced .ph4 file
# Works only well for less than seven flexible residues.
makeFullPPQ <- function(myCoords, fileName) {

# Available colors
colorNames <- c("red", "yellow", "green", "blue", "purple", "orange")
lightNames <- paste("light", colorNames, sep = "\")
lightColors <- c("f20d0d", "f2f20d", "df20d", "2b71ff", "f20df2", "ff8000")

sink(paste(fileName, ".ph4", sep = "\"))
cat("#moe:ph4que 2010.1\n")
cat("#pharmacophore 5 tag t value *\n")
cat("scheme t Unified matchsize i 0 title t \$ comment s \$ smask i 476156\n")
cat("#feature", dim(myCoords)[1] * dim(myCoords)[2] / 3, "expr tt color ix x r y r z r r
r ebits ix gbits ix\n")
for (entity in 0:(dim(myCoords)[2] / 3 - 1)) for (i in 1:dim(myCoords)[1]) cat("AtomQ",
lightColors[entity + 1], myCoords[i, entity * 3 + 1], myCoords[i, entity * 3 + 2],
myCoords[i, entity * 3 + 3], "\n")
cat("#endpharmacophore\n")
sink(NULL)

# Some remarks on the colors used
sink(paste(fileName, ".txt", sep = "\"))
cat("Color - residues matching:\n=====\n")

```

```

    cat("\Residues:\t", paste(sub(\"_x\", \"\", colnames(myCoords)[seq(1, dim(myCoords)[2],
3)]), collapse = \"\t\", \"\n\")
    cat("\Colors:\t", paste(lightNames[1:(dim(myCoords)[2] / 3)],
collapse = \"\t\", \"\n\")
    cat("\Codes:\t", paste(lightColors[1:(dim(myCoords)[2] / 3)],
collapse = \"\t\", \"\n\n\")
    cat("\Documentation for: \", filename, \".ph4\", sep = \"\")
    sink(NULL)
}

# Make pharmacophores containing all docking poses flexible residues as centers
for (moveFile in moveFiles) {
  myCoords <- read.table(moveFile, header = TRUE)[1:topPoses, ]
  potential <- sub(\".txt\", \"\", sub(\".*\/move_\", \"\", moveFile))
  saveFile <- paste(graphicsFolder, \"/ppq_\", potential, \"_full\", sep = \"\")
  makeFullPPQ(myCoords, saveFile)
}

# Make pharmacophores clustered by C.cat positioning
# for each potential and some different cluster distances (1 to 15), essentially
for (moveFile in moveFiles) for (clusterDist in 1:15) {
  myCoords <- read.table(moveFile, header = TRUE)[1:topPoses, ]
  potential <- sub(\".txt\", \"\", sub(\".*\/move_\", \"\", moveFile))
  saveFile <- paste(graphicsFolder, \"/ppq_\", potential, \"_\", clusterDist, sep = \"\")
  makePharmacophores(myCoords, clusterDist, saveFile)
}

# library(lattice)

# # An onion example
# require(onion)
# myMatrix <-
read.table(\"/raid/home/rklein/DOCK/plants/transform/transform_20110818_08_28/distances/move_-
0.4_-0.4.txt\", header = T)
# columnLabels <- rep(unique(gsub(\"_.*\", \"\", colnames(myMatrix))), each =
dim(myMatrix)[1])
# myDataFrame <- data.frame(cbind(X = as.vector(unlist(c(myMatrix[, seq(1, dim(myMatrix)[2],
3])))), Y = as.vector(unlist(c(myMatrix[, seq(2, dim(myMatrix)[2], 3])))), Z =
as.vector(unlist(c(myMatrix[, seq(3, dim(myMatrix)[2], 3])))), groupLabel = columnLabels))
# myMatrix <- myDataFrame[,1:3]
# matrix(as.numeric(as.matrix(myMatrix)), 500,3)

# # a lattice example
# myMatrix <-
read.table(\"/raid/home/rklein/DOCK/plants/transform/transform_20110818_08_28/distances/move_-
0.4_-0.4.txt\", header = T)
# columnLabels <- rep(unique(gsub(\"_.*\", \"\", colnames(myMatrix))), each =
dim(myMatrix)[1])
# myDataFrame <- data.frame(cbind(X = as.vector(unlist(c(myMatrix[, seq(1, dim(myMatrix)[2],
3])))), Y = as.vector(unlist(c(myMatrix[, seq(2, dim(myMatrix)[2], 3])))), Z =
as.vector(unlist(c(myMatrix[, seq(3, dim(myMatrix)[2], 3])))), groupLabel = columnLabels))
# cloud(Z~X*Y, data = myDataFrame, groups = groupLabel, scales = list(arrows=FALSE), xlab =
\"X (Å...)\", ylab = \"Y (Å...)\", zlab = \"Z (Å...)\", col = rainbow(dim(myCoords)[2] / 3))

# # This would use lattice's cloud method
# drawPoints3d <- function(myMatrix, fileName, deviceName = \"none\") {
#   if (deviceName == \"pdf\") {
#     saveFile <- paste(fileName, \".pdf\", sep=\"\")
#     pdf(saveFile, 10, 10)
#   } else if (deviceName == \"png\") {
#     saveFile <- paste(fileName, \".png\", sep=\"\")
#     png(saveFile, width = 1000, height = 1000)
#   }
# }
#
# columnLabels <- rep(unique(gsub(\"_.*\", \"\", colnames(myMatrix))), each =
dim(myMatrix)[1])
# myDataFrame <- data.frame(cbind(X = as.vector(unlist(c(myMatrix[, seq(1, dim(myMatrix)[2],
3])))), Y = as.vector(unlist(c(myMatrix[, seq(2, dim(myMatrix)[2], 3])))), Z =
as.vector(unlist(c(myMatrix[, seq(3, dim(myMatrix)[2], 3])))), groupLabel = columnLabels))
# cloud(Z~X*Y, data = myDataFrame, groups = groupLabel, scales = list(arrows=FALSE), xlab =
\"X (Å...)\", ylab = \"Y (Å...)\", zlab = \"Z (Å...)\", col = rainbow(dim(myMatrix)[2] / 3))
# }

" > ${workDir}/PPQdepict.R

```

```
ssh -X rklein@brandt61 R CMD BATCH ${workDir}/PPQdepict.R >> $logFile
#ssh -X rklein@ccc08d4 /vol/R/R-2.10.0/bin/R CMD BATCH ${workDir}/PPQdepict.R >> $logFile

echo " - done."
printf "\n - done.\n" >> $logFile
echo "PPQdepict completely done on `date`."
echo "See $logFile for logged information and $graphicsFolder for the produced graphics."
echo "PQdepict completely done on `date`." >> $logFile
```

A 26 Liganddefinition α -Terpinylkation (.mol2-Datei)

```
@<TRIPOS>MOLECULE
alphaTerpinylCation
27 27 1 0 0
SMALL
USER_CHARGES
```

```
@<TRIPOS>ATOM
```

1	C1	21.7210	57.1300	-53.1500	C.3	1	F3P	-0.0455
2	H11	21.0274	56.9529	-52.3002	H	1	F3P	0.0980
3	H12	21.6603	58.2151	-53.3701	H	1	F3P	0.0681
4	C2	23.0880	56.7994	-52.6756	C.2	1	F3P	-0.1873
5	H2	23.5057	57.4909	-51.9307	H	1	F3P	0.1359
6	C3	23.7721	55.7090	-53.0463	C.2	1	F3P	-0.1012
7	C4	23.2454	54.7589	-54.0688	C.3	1	F3P	-0.0755
8	H41	23.5277	53.7180	-53.8069	H	1	F3P	0.1032
9	H42	23.7658	54.9730	-55.0348	H	1	F3P	0.0505
10	C5	21.7402	54.8462	-54.2319	C.3	1	F3P	-0.0747
11	H51	21.2413	54.3955	-53.3489	H	1	F3P	0.1019
12	H52	21.4055	54.2545	-55.1069	H	1	F3P	0.0668
13	C6	21.2289	56.3017	-54.3428	C.3	1	F3P	-0.1653
14	H6	20.1084	56.2628	-54.2918	H	1	F3P	0.1161
15	C7	21.6096	56.8478	-55.6666	C.cat	1	F3P	0.4140
16	C8	20.6197	56.8241	-56.7326	C.3	1	F3P	-0.2124
17	H81	21.0355	56.9549	-57.7442	H	1	F3P	0.1147
18	H82	19.9029	57.6566	-56.5742	H	1	F3P	0.1507
19	H83	20.0061	55.9048	-56.7222	H	1	F3P	0.1286
20	C9	22.9355	57.3566	-55.9275	C.3	1	F3P	-0.2325
21	H91	23.6346	56.4782	-55.9110	H	1	F3P	0.1693
22	H92	23.3140	58.0052	-55.1118	H	1	F3P	0.1500
23	H93	23.0627	57.8724	-56.8909	H	1	F3P	0.1174
24	C10	25.1029	55.3786	-52.4767	C.3	1	F3P	-0.0777
25	H101	25.8293	55.1592	-53.2707	H	1	F3P	0.0551
26	H102	25.0297	54.4860	-51.8394	H	1	F3P	0.0688
27	H103	25.5128	56.1909	-51.8619	H	1	F3P	0.0631

```
@<TRIPOS>BOND
```

1	1	2	1
2	1	3	1
3	1	4	1
4	1	13	1
5	4	5	1
6	4	6	2
7	6	7	1
8	6	24	1
9	7	8	1
10	7	9	1
11	7	10	1
12	10	11	1
13	10	12	1
14	10	13	1
15	13	14	1
16	13	15	1
17	15	16	1
18	15	20	1
19	16	17	1
20	16	18	1
21	16	19	1
22	20	21	1
23	20	22	1
24	20	23	1
25	24	25	1
26	24	26	1
27	24	27	1

```
@<TRIPOS>SUBSTRUCTURE
```

```
1 F3P 13 GROUP 4 **** ** 0
```

```
# MOE 2010.10 (io_trps.svl 2010.06)
```

A 27 Liganddefinition (Z,Z)-Farnesylkation (.mol2-Datei)

```
@<TRIPOS>MOLECULE
zzFarnCationPLANTS
40 39 1 0 0
SMALL
USER_CHARGES
```

```
@<TRIPOS>ATOM
```

1	C	118.5309	-22.5914	113.5513	C.2	1	<1>	-0.2941
2	H	117.9999	-23.4863	113.8404	H	1	<1>	0.0000
3	C	118.2040	-21.4296	114.1739	C.cat	1	<1>	0.2163
4	H1	118.7135	-20.5143	113.9112	H	1	<1>	0.0000
5	H2	117.4346	-21.4209	114.9317	H	1	<1>	0.0000
6	C	119.5226	-22.7049	112.5481	C.cat	1	<1>	0.3731
7	C	120.3054	-21.5540	112.0691	C.3	1	<1>	-0.2105
8	H1	120.7306	-21.7418	111.0490	H	1	<1>	0.1314
9	C	121.4825	-21.2637	113.0244	C.3	1	<1>	-0.0177
10	H1	122.1552	-22.1522	113.0698	H	1	<1>	0.0762
11	H2	121.1076	-21.0853	114.0505	H	1	<1>	0.0535
12	C	122.2314	-20.0800	112.5346	C.2	1	<1>	-0.2326
13	H	121.8659	-19.0992	112.8647	H	1	<1>	0.1216
14	C	123.3164	-20.1735	111.7490	C.2	1	<1>	-0.0317
15	C	124.0624	-18.9659	111.3138	C.3	1	<1>	-0.0888
16	H1	124.1781	-18.9564	110.2206	H	1	<1>	0.0621
17	H2	125.0704	-18.9633	111.7522	H	1	<1>	0.0681
18	H3	123.5693	-18.0298	111.6079	H	1	<1>	0.0553
19	C	123.8458	-21.4883	111.2704	C.3	1	<1>	-0.0839
20	H1	124.9493	-21.4449	111.1800	H	1	<1>	0.0841
21	C	123.2224	-21.8686	109.9314	C.3	1	<1>	-0.0510
22	H1	123.4488	-21.0881	109.1688	H	1	<1>	0.0769
23	H2	122.1106	-21.8905	110.0218	H	1	<1>	0.0264
24	C	123.6971	-23.2037	109.4843	C.2	1	<1>	-0.2150
25	H	123.5631	-24.0203	110.2057	H	1	<1>	0.0969
26	C	124.2703	-23.4480	108.2972	C.2	1	<1>	-0.0635
27	C	124.7200	-24.8181	107.9314	C.3	1	<1>	-0.0828
28	H1	125.8049	-24.8333	107.7573	H	1	<1>	0.0598
29	H2	124.2329	-25.1516	107.0050	H	1	<1>	0.0550
30	H3	124.4994	-25.5610	108.7093	H	1	<1>	0.0436
31	C	124.5053	-22.4059	107.2640	C.3	1	<1>	-0.0881
32	H1	123.9514	-22.6407	106.3448	H	1	<1>	0.0576
33	H2	125.5712	-22.3488	107.0041	H	1	<1>	0.0616
34	H3	124.1884	-21.4055	107.6004	H	1	<1>	0.0472
35	H3	123.6317	-22.2768	112.0287	H	1	<1>	0.0526
36	H3	119.6788	-20.6419	111.9744	H	1	<1>	0.1089
37	C	119.8026	-24.0152	111.9666	C.3	1	<1>	-0.1892
38	H1	120.8741	-24.1437	111.7300	H	1	<1>	0.1191
39	H2	119.2672	-24.1014	111.0002	H	1	<1>	0.1364
40	H3	119.4759	-24.8637	112.5870	H	1	<1>	0.1059

```
@<TRIPOS>BOND
```

1	1	2	1
2	1	3	1
3	1	6	2
4	3	4	1
5	3	5	1
6	6	7	1
7	6	37	1
8	7	8	1
9	7	9	1
10	7	36	1
11	9	10	1
12	9	11	1
13	9	12	1
14	12	13	1
15	12	14	2
16	14	15	1
17	14	19	1
18	15	16	1
19	15	17	1
20	15	18	1
21	19	20	1
22	19	21	1
23	19	35	1

```

24 21 22 1
25 21 23 1
26 21 24 1
27 24 25 1
28 24 26 2
29 26 27 1
30 26 31 1
31 27 28 1
32 27 29 1
33 27 30 1
34 31 32 1
35 31 33 1
36 31 34 1
37 37 38 1
38 37 39 1
39 37 40 1
@<TRIPOS>SUBSTRUCTURE
  1 **** 21 GROUP 4 **** **** 0
# MOE 2010.10 (io_trps.svl 2010.06)

```

A 28 Analysedateien für *Docking*-Komplexe

Docking von (Z,Z)-Farnesylkation in Santalen- und Bergamotensynthase-Modell

```

# Analysis of zzFPP into ShSBSman, clustered dockings
# with TRP746,MET763,TYR495,TYR519 flexible:
# 3, 6 = C.cat
# 4 5 9 13 11 7 = TYR495 aromat
# 21 22 26 30 28 24 = TYR519 aromat
# 43 47 51 53 49 46 = TRP746 aromat
# 36 = MET763 sulfur
#
#
# Carbocations versus TYR495 aromat
3 FLEX CENTER 4 5 9 13 11 7
6 FLEX CENTER 4 5 9 13 11 7
# Carbocations versus TYR519 aromat
3 FLEX CENTER 21 22 26 30 28 24
6 FLEX CENTER 21 22 26 30 28 24
# Carbocations versus TRP746 aromat
3 FLEX CENTER 43 47 51 53 49 46
6 FLEX CENTER 43 47 51 53 49 46
# Carbocations versus MET sulfur
3 FLEX 36
6 FLEX 36

```

Docking von α -Terpinylkation in Limonensynthase-Modell

```

# Analysis of aTPC into LimoSyn, clustered dockings
# with TRP324 MET458 flexible:
# 15 = C.cat
# 9 13 17 15 19 12 = TRP324 aromat
# 35 = MET458 sulfur
#
#
# Carbocation versus TRP carbon aromat
15 FLEX CENTER 9 13 17 15 19 12
# Carbocation versus MET sulfur
15 FLEX 35

```

A 29 Beispiel für Ergebnisse der *Cluster*-Analysen

MOE-Pharmakophor-Repräsentation des *Clustering* der *Docking*-Ergebnisse aus dem Experiment R mit der Potenzialfunktion -0.6_-0.6 bei gefordertem *Cluster*-Abstand von 5 Å

```

#moe:ph4que 2010.1
#pharmacophore 5 tag t value *
scheme t Unified matchsize i 0 title t $ comment s $ smask i 476156

```

```
#feature 10 expr tt color ix x r y r z r r r ebits ix gbits ix
AtomQ f20d0d 127.7037 -19.99518 111.8186 0.94 0 300
AtomQ 900909 128.9348 -25.22432 110.6616 0.94 0 300
AtomQ 900909 131.4919 -21.46905 115.1262 0.94 0 300
AtomQ 900909 127.0789 -16.14381 111.5836 0.94 0 300
AtomQ 900909 128.4289 -21.91582 102.3785 0.94 0 300
AtomQ f2f20d 126.8383 -18.4281 115.9305 0.06 0 300
AtomQ c2a10b 129.8366 -25.35913 110.5567 0.06 0 300
AtomQ c2a10b 131.2317 -21.46705 115.3070 0.06 0 300
AtomQ c2a10b 127.2920 -14.86215 116.1010 0.06 0 300
AtomQ c2a10b 128.4651 -21.94253 102.3753 0.06 0 300
#endpharmacophore
```

Erzeugte Zusatzinformationen zur obigen Pharmakophor-Repräsentation

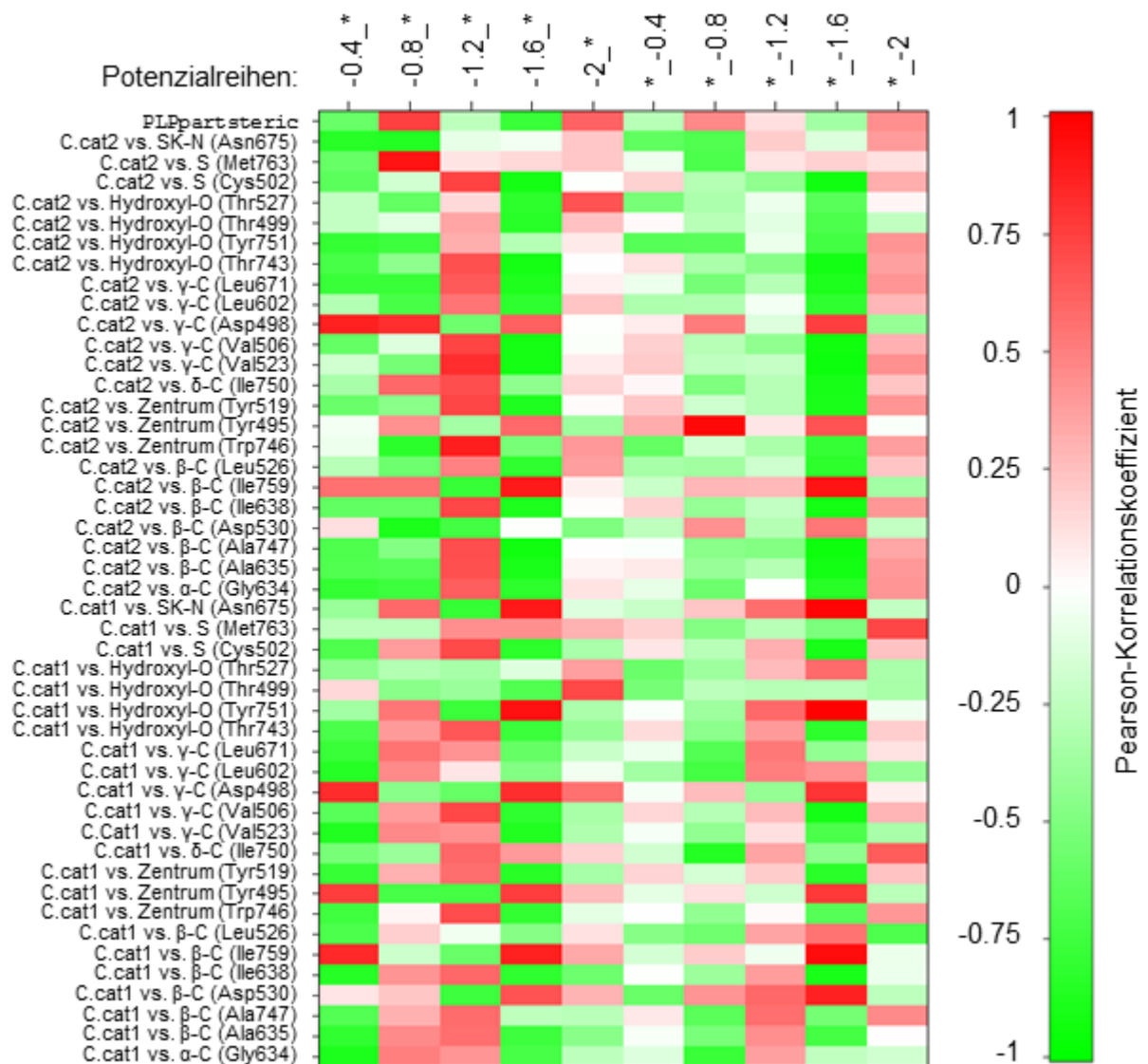
Colors used:

=====

```
Cluster:      1      2
Color: red    yellow
cat_3_6 :     lightred  lightyellow
equals:      f20d0d     f2f20d
Others:      darkred   darkyellow
equals:      900909     c2a10b
Flexible residues: TRP746      MET763 TYR495 TYR519
```

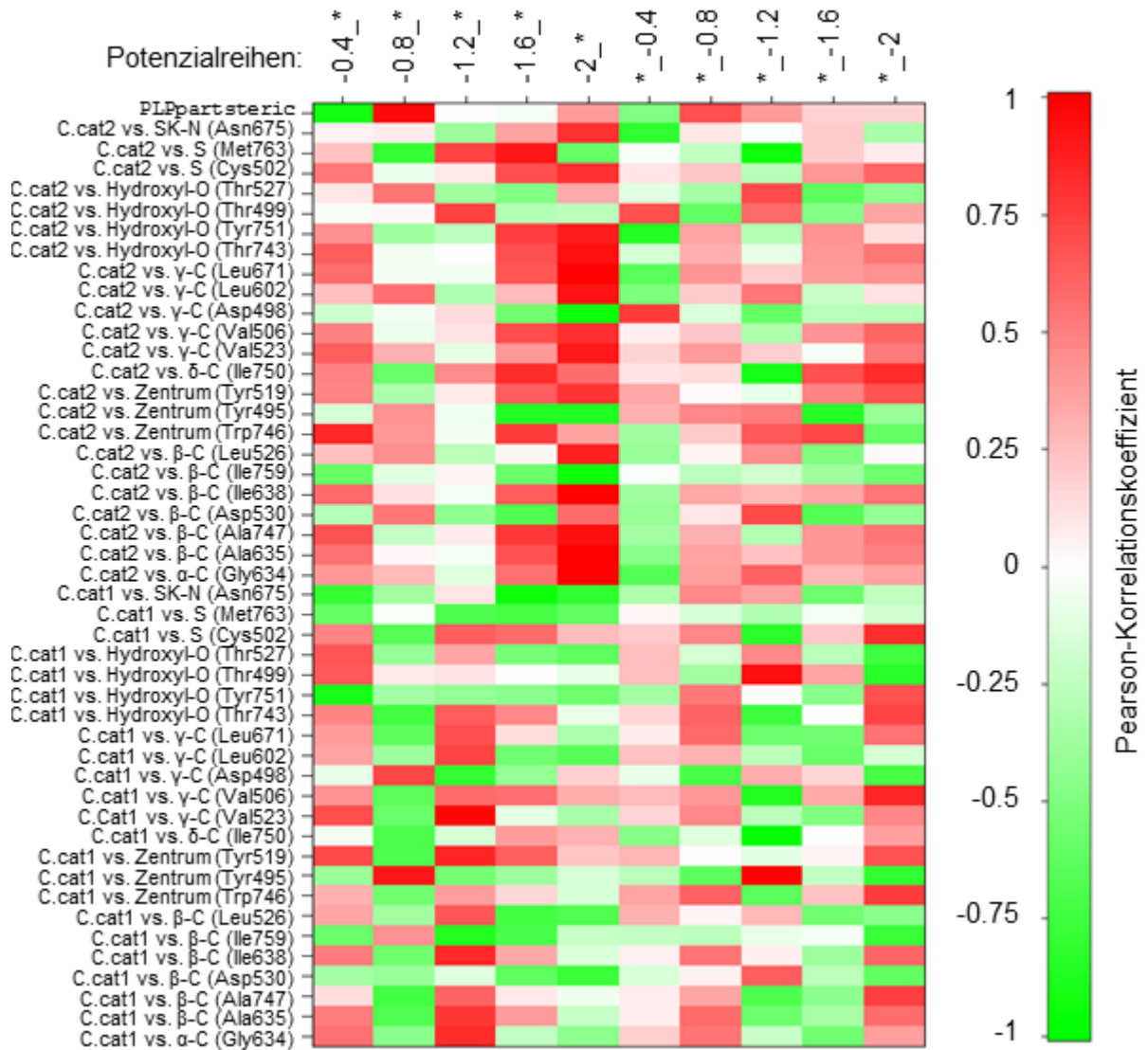
```
cat_3_6_x cat_3_6_y cat_3_6_z TRP746_x TRP746_y TRP746_z MET763_x MET763_y
1 127.7037 -19.99518 111.8186 128.9348 -25.22432 110.6616 131.4919 -21.46905
2 126.8383 -18.42810 115.9305 129.8366 -25.35913 110.5567 131.2317 -21.46705
MET763_z TYR495_x TYR495_y TYR495_z TYR519_x TYR519_y TYR519_z
1 115.1262 127.0789 -16.14381 111.5836 128.4289 -21.91582 102.3785
2 115.3070 127.2920 -14.86215 116.1010 128.4651 -21.94253 102.3753
clusterWeight
1 0.94
2 0.06
```


A 30 Korrelationen der Parametervariation (Experiment R)



Korrelation der Parametervariation im Experiment R mit Rangkorrelationskoeffizienten von PLPpartstERIC und gemessenen Distanzen ausgewählter Wechselwirkungen aus den entsprechenden Potenzialvarianten (Bezeichnungen siehe Abbildung 69, Seite 112)

A 31 Korrelationen der Parametervariation (Experiment *s*)



Korrelation der Parametervariation im Experiment *s* mit Rangkorrelationskoeffizienten von PLPpartstERIC und gemessenen Distanzen ausgewählter Wechselwirkungen aus den entsprechenden Potenzialvarianten (Bezeichnungen siehe Abbildung 69, Seite 112)

CURRICULUM VITAE

Diplom-Bioinformatiker Robert Klein, geboren am 02. Juli 1981 in Schkeuditz

- | | |
|-------------------|--|
| 01/2008 – 01/2012 | Promotion am Leibniz-Institut für Pflanzenbiochemie Halle (Saale) (IPB) in der Arbeitsgruppe Computerchemie |
| 11/2007 – 12/2007 | Gastwissenschaftler an der Friedrich-Alexander-Universität Erlangen-Nürnberg

Quantenchemische Berechnungen zur Polarisierung chemischer Bindungen (mit einem Stipendium der „Molecular Graphics and Modeling Society - Deutsche Sektion“), Arbeitskreis von Prof. Dr. Tim Clark |
| 04/2007 – 10/2007 | Wissenschaftliche Hilfskraft am IPB Halle

Entwicklung einer empirischen Methode zur Berechnung chemischer Reaktionen und Standardbildungsenthalpien, Arbeitsgruppe Computerchemie |
| 10/2001-03/2007 | Studium der Bioinformatik an der Martin-Luther-Universität Halle-Wittenberg

Diplomarbeit am IPB Halle, Thema: „Entwicklung eines Programms zur schnellen Berechnung von Bildungsenthalpien von Molekülen“, Arbeitsgruppe Computerchemie unter Betreuung von PD Dr. W. Brandt |
| 07/2000 – 04/2001 | Grundwehrdienst |
| 07/2000 | Abitur am Wilhelm-Ostwald-Gymnasium Leipzig |

VERÖFFENTLICHUNGEN

ARTIKEL IN FACHZEITSCHRIFTEN

1. "Fast access to total energies"
Robert Klein and Wolfgang Brandt
Chemistry Central Journal 2008, 2(Suppl 1):P35 (Meeting abstracts 3rd German Conference on Chemoinformatics: 21. CIC-Workshop)
2. „Prenyl- und Methyltransferasen in Natur und Synthese“
Ludger Wessjohann, Thomas Vogt, Julia Kufka, Robert Klein
Biospektrum 1/2012

POSTERBEITRÄGE

1. "Fast access to total energies"
Robert Klein, Wolfgang Brandt, Ludger Wessjohann
3rd German Conference on Chemoinformatics – 21. CIC-Workshop 2007, 11.-13. November 2007 in Goslar und Molecular Modeling Workshop 2008, 29.-30. April 2008 in Erlangen
2. "Fishing in the proteome: terpene synthases"
Robert Klein, Dimitar Vasilev, Wolfgang Brandt
Plant Science Student Conference, 23.-26.06.2009 in Halle (Saale)
3. "New relationships between bond order and bond length"
Robert Klein, Marko Wehle, Wolfgang Brandt
Model(l)ing '09 - Annual Molecular Modelling Workshop & Annual International Meeting of the MGMS, 07.-11.09.2009 in Erlangen
4. "Modelling of *Arabidopsis thaliana* Isopentenyl Transferases"
Robert Klein, Vaclav Mik, Wolfgang Brandt
3rd EuCheMS Congress, 29.08.-02.09.2010 in Nürnberg
5. "Insights into the substrate specificity of terpene synthases from tomato and tobacco trichomes"
Romy Töpfer, Robert Klein, Wolfgang Brandt and Alain Tissier
25. Tagung Molekularbiologie der Pflanzen, 28.02.- 02.03.2012 in Dabringhausen

VORTRÄGE AUF TAGUNGEN

- „Cytokinin-Biosynthese: Modellierung von DMAPP:ATP/ADP-Transferasen aus *Arabidopsis thaliana*“
41. Doktorandenworkshop Naturstoffe: Chemie, Biologie, Ökologie, 13.05.2011 in Bonn

EIDESSTATTLICHE ERKLÄRUNG

Hiermit erkläre ich, die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst zu haben. Ich verwendete keine als die von mir angegebenen Quellen und Hilfsmittel und habe die den verwendeten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht.

Diese Dissertationsschrift wurde ausschließlich der Naturwissenschaftlichen Fakultät II (Chemie, Physik und Mathematik) der Martin-Luther-Universität Halle-Wittenberg vorgelegt. Eine Bewerbung um den Doktorgrad zu einem früheren Zeitpunkt ist nicht erfolgt.

Halle (Saale), am 08. September 2014

gez. Robert Klein