New Tabu-Search Algorithms for the Exploration of Energy Landscapes of Molecular Systems

Dissertation zur Erlangung des
naturwissenschaftlichen Doktorgrades
der Julius-Maximilians-Universität Würzburg

vorgelegt von

Christoph Grebner

geboren in Nürnberg

Julius-Maximilians-Universität Würzburg
Fakultät für Chemie und Pharmazie

Würzburg 2012

Eingereicht bei der Fakultät für Chemie und Pharmazie am: ____________________

Gutachter der schriftlichen Arbeit

1. Gutachter: Prof. Dr. Bernd Engels

2. Gutachter: Jun.-Prof. Dr. Johannes Kästner

Prüfer des öffentlichen Promotionskolloquiums

1. Prüfer: ____________________

2. Prüfer: ____________________

3: Prüfer: ____________________

Datum des öffentlichen Promotionskolloquiums: ____________________

Doktorurkunde ausgehändigt am: ____________________

List of abbreviations I

ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
API	Application Programming Interface
ASA	Accessible Surface Area
BFGS	Broyden-Fletcher-Goldfarb-Shanno
BH	Basin Hopping
CASP	Critical Assessment of protein Structure Prediction
CAST	Conformational Analysis and Search Tool
CBS	Carboxylate Binding Site
CG	Conjugate Gradient
CI	Climbing Image
CoV	corona-virus
CPU	Central Processing Unit
CS	Conformational Search
CSA	Conformational Space Annealing
CT	C-terminal
DEM	Diffusion Equation Method
DFP	Davidon-Fletcher-Powell
DPS	Discrete Path Sampling
DS	Diversification Search
GM	Global minimum
GOTS	Gradient Only Tabu Search
GPU	Graphical Processing unit
GTS	Gradient Tabu Search
GUI	Graphical User Interface
HDD	Hard Disk Drive
I/O	input/output
L-BFGS	Limited memory Broyden-Fletcher-Goldfarb-Shanno
LJ	Lennard-Jones
MC	Monte Carlo
MCM	Monte Carlo with Minimization
MCMM/LM	Multiple Minima Monte Carlo/Low Mode Sampling
MD	Molecular Dynamics
MDM	Molecular Dynamics with Minimization
MEP	Minimum Energy Path
MM	Molecular Mechanics
MO	Molecular Orbital
MPI	Message Parsing Interface
NEB	Nudged Elastic Band
NMR	Nuclear Magnetic Resonance
NR	Newton Raphson

List of abbreviations II

PDB	Protein Data Base
PES	Potential Energy Surface
PID	proportional-integral-derivative
PSO	Particle Swarm Optimization
PSS	Potential Energy Smoothing and Search
QM	Quantum Mechanics
QM/MM	Quantum Mechanics/Molecular Mechanics
QSAR	quantitative structure-activity relationships
RMSD	Root Mean Square Deviation
SA	Simulated Annealing
SAR	structure-activity relationships
SARS	Severe acute respiratory syndrome
SARS-CoV M^{pro}	SARS-CoV main protease
SC	side-chain
SCF	self consistent field
SD	Steepest Descent
TL	Tabu-List
TPS	Transition Path Sampling
TS	Tabu-Search
TSP	Travelling Salesman Problem
TSPA	Tabu Search with Powell's Algorithm
vdW	van der Waals

List of molecules

Gly-Ala-Ser	(**1**)
Ring-opened EPNP	(**2**)
Ring-opened E64c	(**3**)
Met^5-enkephalin	(**4**)
Glu-Lys-Ser-Cys-Pro	(**5**)
N-Ac-(D-Glu)$_3$-D-Ala, e$_3$a	(**6**)
N-Ac-(D-Glu)$_3$-L-Ala ,e$_3$A	(**7**)
N-Ac-L-Ala-(D-Glu)$_3$,Ae$_3$	(**8**)
N-Ac-D-Ala-L-Ala-(D-Glu)$_2$,aAe$_2$	(**9**)
inhibitor TS174	(**12**)
CBS-amide	(**10**)
CBS-KKF	(**11**)

Contents

1 Motivation and aim of the work

The aim of the present work is the development of new algorithms for the exploration of energy landscapes of various systems. The work is in part based on previous results obtained in the PhD thesis of Svetlana Stepanenko. She developed three Tabu-Search based algorithms, the Gradient Tabu Search (GTS), the Gradient Only Tabu Search (GOTS), and the Tabu Search with Powell's Algorithm (TSPA), whereas GOTS was the most successful one. The performance of GOTS in optimizing various mathematical test functions in comparison to other well-known global optimization algorithms like genetic algorithms and a first application to conformational search problems, motivated for the development of new Tabu-Search based algorithms. However, a detailed comparison of Tabu-Search to other conformational search algorithms was missing. Therefore, the first aim of the present work is a comparison of the GOTS algorithm to other well-known conformational search approaches. In the course of this comparison, weak points of GOTS should be revealed also and the influence of different starting structures provided by the StartOpt algorithm should be investigated. The two main bottlenecks of GOTS are the diversification part and the modest ascent strategy. For the diversification part it was realized that the Tabu-Search represents a very local approach since the closer phase space is searched very accurately. The Basin Hopping approach (see section 2.1.4), however, searches the phase space in a non-local fashion. Therefore, it will be tested if a Basin Hopping approach with large step sizes (i.e. a very diverse search) can improve the diversification strategy of Tabu-Search. In GOTS, the modest ascent is calculated using energy values only. However, the use of gradient information usually improves the performance of optimization algorithms. Therefore, an alternative approach based on a transition state search algorithm will be implemented and tested for its applicability.

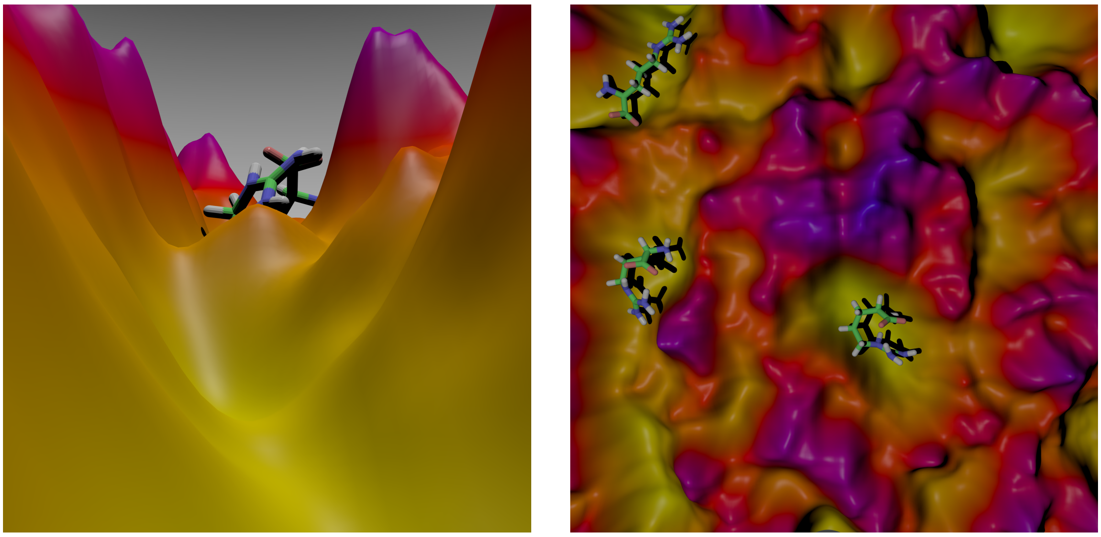
The new algorithms will be implemented in a new conformational search program (Conformational Analysis and Search Tool, CAST). The development of this program is also part of this work providing a platform for a variety of global optimization, conformational search, and analysis tools. As conformational search studies are usually done using force field approaches, the development of CAST starts with the implementation of a common force field. To perform local and global optimization tasks, different local optimization algorithms are needed. In this respect, the performance of different optimization algorithms and libraries should be investigated. Besides local optimization, also transition state search is important for investigations of energy landscapes. Therefore, different algorithms are tested and implemented into the new program package. The investigation of reaction mechanisms of complex systems like large clusters or enzymes is still a very complex task. Hence, algorithms are needed which are able to locate various transition states and reaction pathways with many intermediate minima. An overview of existing reaction path sampling methods will be given in the introduction. As part of this work, a new approach is developed which directly delivers transition states lying between two initial states. The new approach will be tested at hand of rearrangement reactions of different Lennard-Jones clusters as these are usual test candidates in this field.

To investigate the broad applicability of the Tabu-Search based algorithms, different application fields will be investigated. The classical application field of global optimization is the conformational search. Here, the performance of the new Tabu-Search algorithm will be shown by means of some case studies with a comparison to other well-known approaches. Conformational search studies often include a refinement process based on *ab initio* methods, which is often a crucial step in the proper description of the investigated system. A complete conformational search based on *ab initio* methods is usually computationally too demanding. However, recent advances in GPU-accelerated *ab initio* programs resulted in speedups which now allows a global optimization entirely based on quantum mechanical methods. A combination of Tabu-Search and TeraChem is used to investigate the performance of such approaches. Besides the conformational search of molecules, the optimization of cluster systems like argon or water clusters, is very important. These clusters are on the one hand very complex, as very diverse structures may exist, and on the other hand are very important for the description of aggregates or solvent effects. The effect of micro-solvation by a globally optimized water shell is investigated taking a small protein system as an example. Another application area of global optimization algorithms is the optimization of a ligand orientation in the active site of a protein, which is also known as docking. The performance of the new Tabu-Search algorithm is tested with the help of an example problem, where the uncertain orientation of the ligand results in an unclear electron density in the x-ray experiment.

2 Introduction

Optimization problems are inherent in nearly every research area, ranging from engineering[1,2] to natural sciences like biology,[3,4] or chemistry.[5] Sometimes, the optimization problems are very easy and can be solved by simple minimization or maximization. However, in most cases the underlying problems are more complex and the best solution, the global optimum, is often located beside several other local optima. Obviously, a maximization can easily be turned into a minimization by a simple overall sign change. Therefore, global optimization techniques are tend to search for the global minimum solution.[6–9] A very well known global optimization problem is the Traveling Salesman Problem (TSP), where a salesman has to visit several different cities looking for the shortest possible travel distance.[10,11] Further examples are reaction design in chemical engineering,[8] protein folding in biology, as the native structure of a protein is often related to the global minimum,[6,7,12,13] or the optimization of chemical reactions.[14]

The main problem in global optimization is the so-called multiple-minima problem which was proposed by Gibson and Scheraga in 1988.[15] The topic is especially discussed in terms of conformational search and protein folding.[16] The term describes the problem that any local minimization procedure will only lead to the closest minimum instead of leading to the desired global minimum. Furthermore, looking from one specific point of the hypersurface, it is not known in which direction the global minimum lies. This problem is illustrated in Figure 2.1. Figure 2.1-a shows the view from a specific starting structure. A close lying minimum can already be seen behind a small barrier following a smooth reaction pathway. However, looking at the surface more globally (Figure 2.1-b) it can be seen that many other, maybe more stable minima can be reached when branching the reaction path and follow-



(a) Close look at the starting structure and one close minimum.

(b) Global look at the hypersurface.

Figure 2.1: Illustration of the multiple-minima problem.

ing a different pathway. According to their method of operation, global optimization algorithms can be classified into deterministic, heuristic, and meta-heuristic methods (see Figure 2.2).[7,8,17,18] Deterministic methods follow a systematic approach. Therefore, they are predictable and always deliver the same results. When the complete search space is investigated (systematic search) the global optimum can be located exactly. However, these algorithms cannot be applied to more complex problems as the search space becomes too large and the computational effort increases exponentially which is also known as combinatorial explosion.[19] Heuristic algorithms use the information of the current solution to decide on the next step. They approximately solve the global optimization problem in an iterative manner. By reduction of a complex problem to a simpler one, they can deliver a solution which is in close agreement with the global optimal solution with relatively low computational effort. However, they are usually only specific to the problem they are designed for and are rarely transferable to other problems. Meta-heuristics, first mentioned by Glover,[20] are a generalization of heuristic algorithms. Meta-heuristics combine heuristic approaches in an abstract way for a hopefully more efficient exploration of the search space. Mostly, the combination is done stochastically. Therefore, most meta-heuristic algorithms can be summarized as stochastic global optimization algorithms. A further distinction criteria of global optimization algorithms is the necessary amount of information of the underlying functional. Some algorithms only require information about function values. More complex but also most often more accurate methods also require information about gradients (first derivatives) or even the Hessian matrix (second derivatives).

Due to the many different possible approaches, a huge variety of algorithms are known, each with its own strengths and weaknesses.[6,9,21] Most of the commonly used global optimization algorithms belong to meta-heuristic stochastic global optimization algorithms. In the following, important representatives of this class are described together with recent application examples.

2.1 Global optimization algorithms

In a recent review, Bernd Hartke described the essential four basic ingredients of stochastic global optimization algorithms which decide about the efficiency of a search method.[9] The key points include:

1. A fast finding and leaving of local minima.
2. A focus on the overall structure of the objective function instead of getting lost in irrelevant details.
3. An efficient jumping into promising regions by exploiting accumulated information instead of jumping blindly
4. An avoidance of enumeration of all minima.

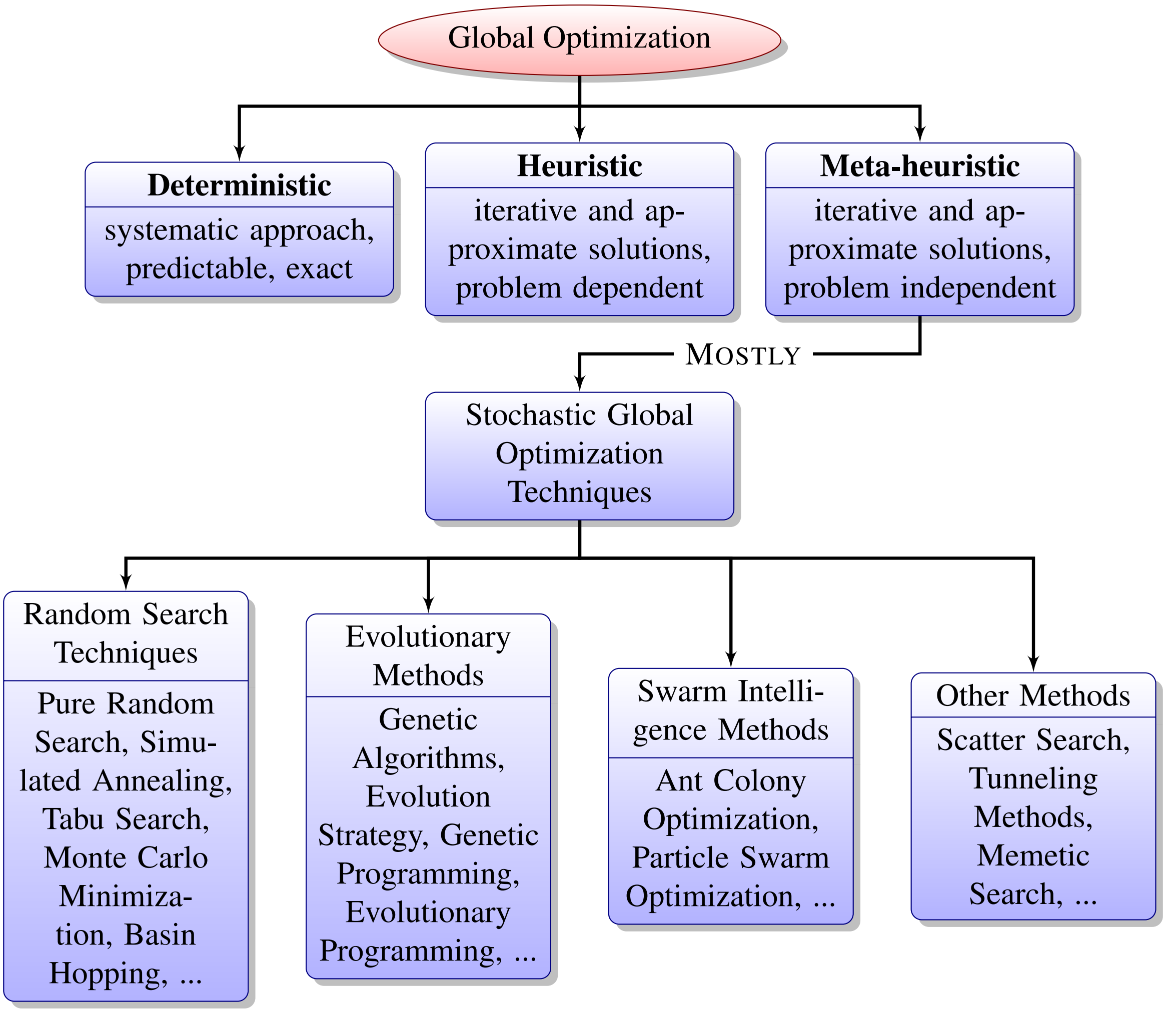

Global Optimization
Deterministic
systematic approach, predictable, exact
Heuristic
iterative and approximate solutions, problem dependent
Meta-heuristic
iterative and approximate solutions, problem independent
MOSTLY
Stochastic Global Optimization Techniques
Random Search Techniques
Pure Random Search, Simulated Annealing, Tabu Search, Monte Carlo Minimization, Basin Hopping, ...
Evolutionary Methods
Genetic Algorithms, Evolution Strategy, Genetic Programming, Evolutionary Programming, ...
Swarm Intelligence Methods
Ant Colony Optimization, Particle Swarm Optimization, ...
Other Methods
Scatter Search, Tunneling Methods, Memetic Search, ...

Figure 2.2: Overview of global optimization algorithms. Illustration reproduced following the book of Gade Pandu Rangaiah.[8]

In principle, each of these points can themselves can be turned into a successful global optimization algorithm. However, combinations of these aspects usually provide a better performance. Therefore, most algorithms in common use try to optimally employ several of the above mentioned points. The most important classes of stochastic meta-heuristic algorithms are listed below. In the following each of them are described in more detail with recent application examples.

- Classical approaches: Molecular Dynamics,[22–24] Simulated Annealing,[25,26] or Monte Carlo[27]

- Evolutionary algorithms: Genetic algorithms[8,9,18,28]

- Swarm algorithms: Particle Swarm Optimization,[29,30] Ant Colony System,[31–34] Artificial Bee Colony[35]

- Hypersurface deformation methods: Diffusion Equation Method,[36] Potential Energy Smoothing and Search,[37,38] Basin Hopping,[7] MCMM/LMOD[27,39–41]

- Stochastic algorithms: Scatter Search,[42] Tabu-Search[43,44] (details in section 3)
- Others: Conformational Space Annealing,[45] StartOpt[46–48]

2.1.1 Classical approaches

Although molecular dynamic approaches do not directly belong to stochastic global optimization, they are included in the enumeration of search algorithms and are summarized as classical approaches. They are commonly used for the sampling of dynamic properties,[19,49–51] but can also be used as conformational search or global optimization approaches, especially when combined with local optimization (Molecular Dynamics with Minimization, MDM).[22–24] MD can be efficiently used to explore the closer neighborhood. However, the performance of locating the global minimum for complex systems has to be further improved.[52] One approach which tries to overcome the problem of a small investigated conformational space is the local elevation method proposed by Huber.[53] Here, the concept of memory is introduced into a molecular dynamics algorithm. Thus, new areas in the search space are preferred instead of sampling a small number of low-energy regions. Other classical optimization and sampling algorithms adapted to global optimization or conformational search are for example the simulated annealing (SA)[25,26] or standard Monte Carlo (MC)[27] simulations. Since classical approaches are usually less efficient than other global optimization approaches and are mostly employed to cover other properties like dynamical behaviors or free energy calculations, no further details about these algorithms will be presented here.

2.1.2 Evolutionary algorithms

One very important group of meta-heuristic algorithms are evolutionary algorithms.[8,9,18,28] They use the mechanisms of biological evolution by means of natural selection (survival of the fittest) proposed by Charles Darwin.[54] The algorithm starts with an initial population of individuals. These can either be chosen randomly or with prior knowledge about the search space. Calculation of the objective function (fitness function) and a recombination scheme (for original evolutionary algorithm: "survival of the fittest") produces a new population with a hopefully better solution of the objective function. These iterations are repeated until some convergence criteria are met. The different variants differ mainly in the recombination schemes. A principle scheme of an evolutionary algorithm is given in Figure 2.3.[8,9,18,28] Genetic algorithms are a particularly efficient variant of evolutionary algorithms.[9] Here, the recombination scheme consists of mating between two parents (starting individuals) by crossing over to produce one or two children solutions. Further variability is included by the allowance of mutation of a child by a given probability (i.e. changing a small

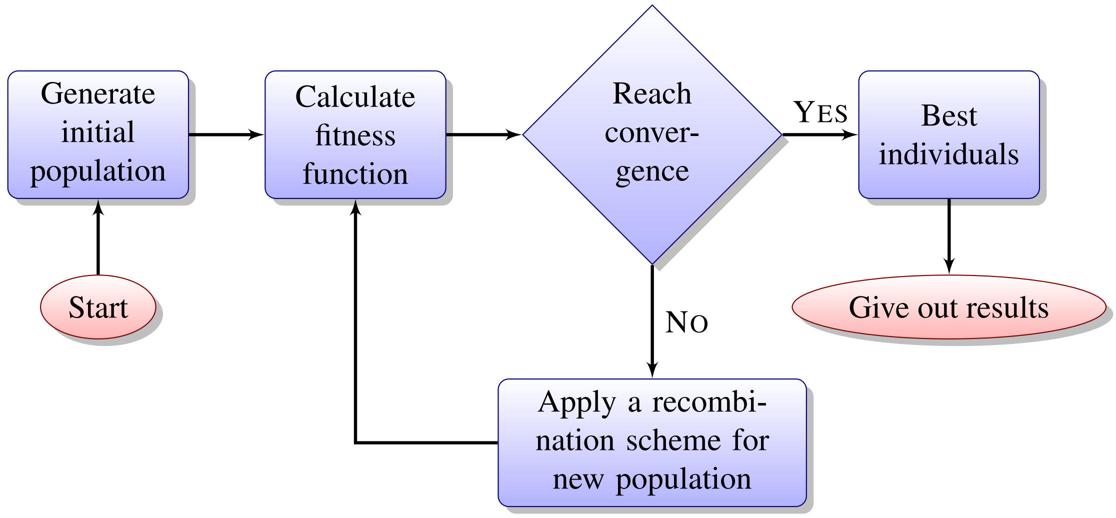

Generate initial population
Calculate fitness function
Reach convergence
YES
Best individuals
Start
NO
Give out results
Apply a recombination scheme for new population

Figure 2.3: Flowchart of an evolutionary algorithm.[28]

number of elements/bits within the new solution). Genetic algorithms are widely spread in computational chemistry and the application areas range from material sciences and cluster optimization[55–60] to optimization of biochemical systems.[61–65] An extensive overview about recent developments and applications is given in the review of Bernd Hartke.[9] Hence, the reader is referred to this article for further details.

2.1.3 Swarm algorithms

Swarm algorithms employ a collective intelligence from either natural individuals like flocks of birds, schools of fish, or colonies of ants,[18] or artificial individuals like, for example, the Borg from the science fiction serial Star Trek.[66] Swarm intelligence is usually self-organized, decentralized, and distributed over the whole swarm and emerges through the cooperation of a large number of homogeneous particles in an environment.[18] The information itself can either be stored throughout the whole swarm or through the use of markers like pheromones in ants or dancing in bees. The term *swarm algorithms* was introduced by Beni in 1989.[67] Several variants of swarm algorithms are known. The most important are Particle Swarm Optimization (PSO) first mentioned by Reynolds[29] and developed by Kennedy,[30] Ant Colony Optimization (ACO) proposed by Dorigo,[31–34] and the very recent Artificial Bee Colony (ABC) algorithm.[35]

Particle Swarm Optimization In Particle Swarm Optimization (PSO), an initial randomly distributed swarm of particles with randomly assigned velocities is simulated. At each update step, the new velocity ($v_i(t+1)$) of a particle is calculated using its current velocity ($v_i(t)$), the best known global position in the search space (p_{gbest}) and the best personal position of the particle (p_i^{best}, see Equation 2.1). The position of a particle is then updated using Equation

2.2. As the simulation proceeds, the swarm moves into the direction of the globally best solution until eventually most particles are located at the global optimum. PSO usually does not require gradient information and is therefore very fast. However, as for most optimization algorithms the additional use of gradients can accelerate the method. The convergence is usually slightly better than the one of genetic algorithms. Even so, as the scanned phase space depends on the velocities and the initial distribution of the particles, the performance drastically depends on the initial setup.[8,18,68]

$$\begin{aligned} v_i(t+1) = v_i(t) &+ \left(c_1 \times rand() \times \left(p_i^{best} - p_i(t)\right)\right) \\ &+ \left(c_2 \times rand() \times \left(p_{gbest} - p_i(t)\right)\right) \end{aligned} \tag{2.1}$$

$$p_i(t+1) = p_i(t) + v_i(t) \tag{2.2}$$

PSO algorithms are widely used for global optimization tasks in various research areas. A recent investigation employed a PSO algorithm to optimize the parameters of a PID-controller (proportional-integral-derivative controller,) which was used for image encryption in wireless data transaction. The study also revealed a superior performance of PSO in comparison to evolutionary programming.[69] Like the optimization of the parameters of a PID-controller, Bieler *et al.* employed PSO based algorithms to optimized both, the geometrical design and the operation of the manufactured sensor of mass spectrometers.[70] PSO algorithms are not only used for the optimization of hardware components. Prasad and Souradeep compared PSO to Markov chain Monte Carlo based optimization of cosmological parameter estimation of cold dark matter using Wilkinson Microwave Anisotropy Probe data.[71]

Besides applications in engineering and physics, PSO is also often applied in several areas of chemistry. Park *et al.* combined global optimization based on genetic algorithms and particle swarm optimization to assist the discovery of novel material from high-throughput experiments.[72] Similarly, PSO optimization is used for data mining and clustering,[73,74] analysis of SAR data and the discovery of activity cliffs within SAR studies,[75–77] the evaluation of QSAR studies,[78,79] or the parameter estimation of model systems for dynamics in biological systems.[80] Further complex optimization tasks like phase equilibrium and stability[81–83] or secondary structure predictions of RNA systems[84] can also be solved by PSO algorithms. Chuang *et al.* proposed a version of the PSO algorithm to locate CpG-islands in the human genome. CpG-islands are important for the regulation of gene expression and are therefore one mechanism of epigenetic gene regulation. A methylated CpG-islands leads to an unexpressed gene.[85]

In a recent study, Liu[86] implemented a new PSO approach into the docking program AutoDock. A comparison of the four state-of-the-art docking approaches, GOLD, Dock, FlexX, and AutoDock employing a genetic algorithm, to the new PSO approach in AutoDock, showed its efficiency and a higher accuracy. Therefore, the PSO variant in AutoDock is a promising approach for virtual screening.[86] Similarly, PSO algorithms are used for docking in the Open-Source project Paradocks.[87]

Ant Colony Optimization Figure 2.4 shows an illustration of the ACO algorithm. It employs the principle, that ants lay down a pheromone once they have found a food source. These pheromones decay with time. Therefore, the longer a path takes to travel, the more pheromone evaporates. Initially, the ants wander around randomly. However, the more ants take the same shorter path, the more intensive the pheromone trail gets and the more intensive the shortest path is used. This finally leads to a path which is used by the whole collective.

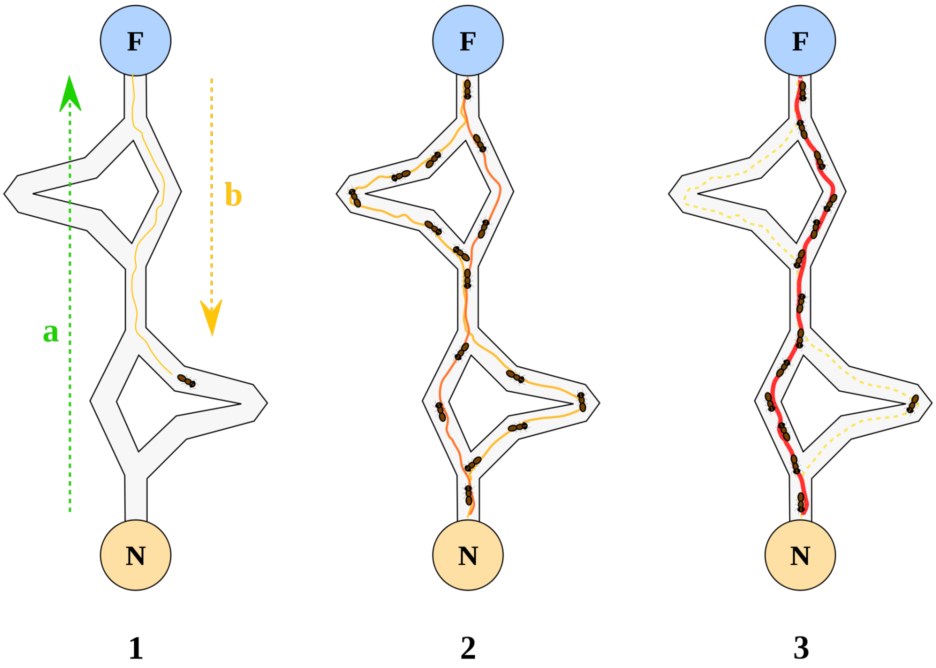


Figure 2.4: Illustration of the Ant Colony Optimization algorithm.[88] 1) First ant finds the food source (F), using some path a) or b), then returns to the nest (N) leaving behind a pheromone trail . 2) Other ants follow the possible paths, with reinforcement on shortest path. 3) Ants follow shortest path as the pheromone trail on longer paths evaporates.

The ant colony optimization algorithm is now widely used in different research areas. First applications were concerned with the solution of the Traveling Salesman Problem[32] with recent developments on the parallelization of the ACO for TSP.[89] Furthermore, ACO is used for the generation of keys for the encryption of binary images.[90]

Besides application on computer science, ACO algorithms are nowadays widely used in computational chemistry. Korb *et al.* implemented an ACO approach for the optimization of protein ligand interactions which are very important for structure-based drug design, virtual screening, or docking.[91] The performance of the protein-ligand ANTSystem (PLANTS) algorithm was compared to the state-of-the art docking program GOLD which uses a genetic algorithm and several improvements in pose prediction were observed.[92,93] The new algorithm was further improved by the implementation of a GPU-accelerated version.[94] Another very important topic in pharmaceutical computational chemistry approaches are quantitative structure-activity relationships (QSAR), where relations between structural aspects of a molecule and its activity with a certain target are investigated. Thus, the best leading structure for a specific behavior can be located. Recently, several groups have employed ACO based algorithms for solving such optimization problems.[95–98] Li *et al.* combined an ACO algorithm with support vector machines to investigate the methylation sites of proteins. Protein methylation is one of the most common post-translation modifications to influence the behavior of proteins.[99] ACO algorithms are also used in vibrational spectroscopy, like IR or Raman, to select the most important spectral wavelengths from a large number of available variables.[100,101]

Artificial Bee Colony The ABC algorithm proposed by Karaboga[35] employs the intelligent behavior of honey bees. As in nature, three groups of bees are used: employees, onlookers and scouts. The first half of the colony consists of employees and the second half of onlookers. Only one employee exists for each food source. In the beginning, all employees are scouts which have to look for a food source. Once a scout has found a food source, it becomes an employee. The employee starts to utilize the food source. As soon as the food source of one employee is exhausted, the employee becomes a scout. A visualization of the algorithm can be seen in Figure 2.5.

Although ABC algorithm are rather new, several application areas to real life problems exist already,[102] like optimization of scheduling problems,[103] protein structure prediction,[104–106] or the minimization of bond-cleavage-induced perturbations in QM/MM calculations.[107] Further improvements to ABC algorithms include for example the combination with evolutionary algorithms.[63,108]

2.1.4 Hypersurface deformation algorithms

The high complexity and the huge amount of possible local minima leads to the multiple-minima problem which makes global optimization and conformational search very difficult.[15] A possible approach is the deformation of the potential energy surface to decrease the number of minima or make them accessible more easily. Hypersurface deformation with respect to optimization of nonlinear problems was first mentioned by Stillinger in 1988,[109] and soon afterwards used for conformational search by Piela in 1989.[36]

Several possibilities for deformation of the underlying potential energy surface were

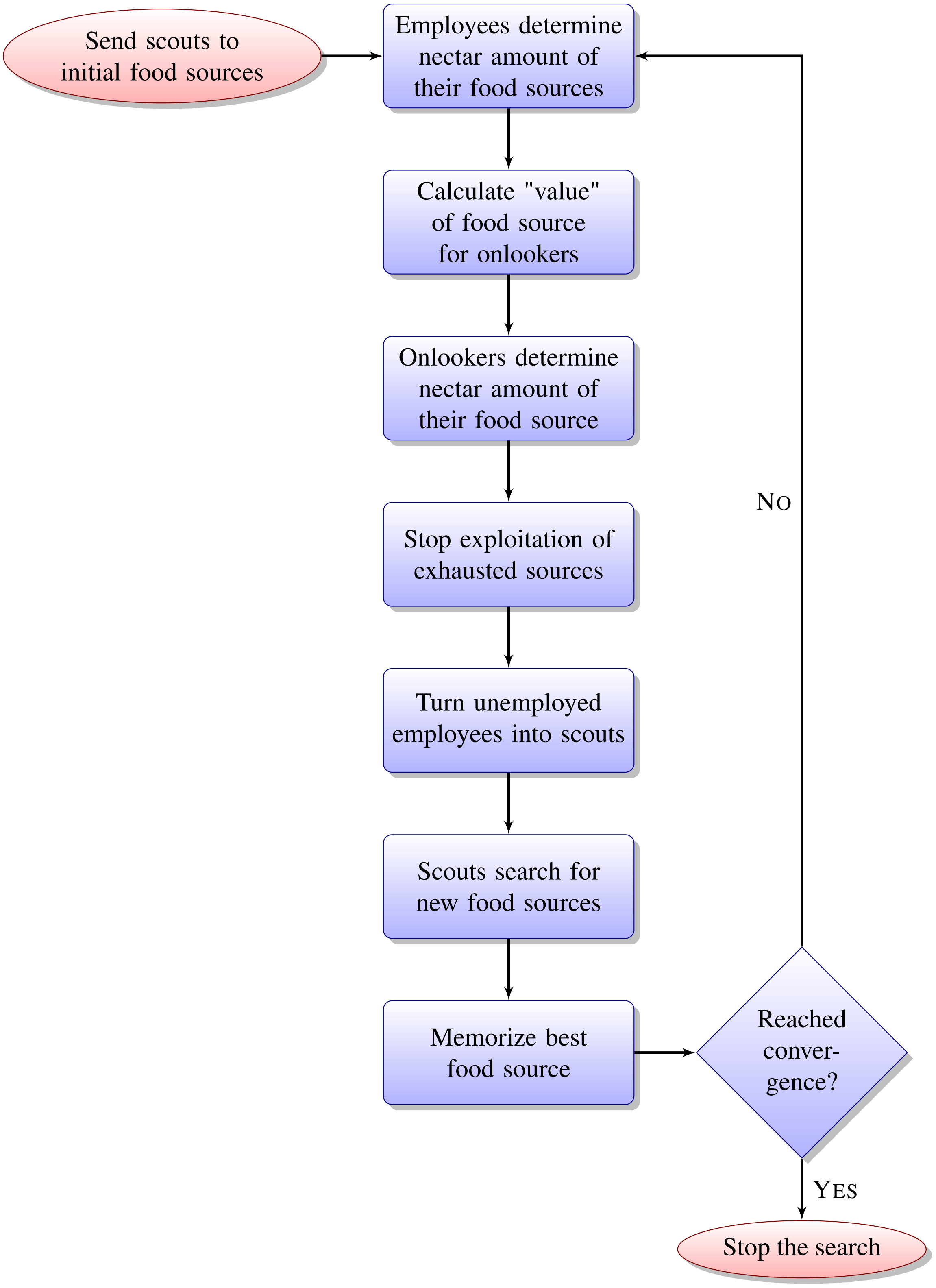

Send scouts to initial food sources
Employees determine nectar amount of their food sources
Calculate "value" of food source for onlookers
Onlookers determine nectar amount of their food source
Stop exploitation of exhausted sources
Turn unemployed employees into scouts
Scouts search for new food sources
Memorize best food source
Reached conver-gence?
No
Yes
Stop the search

Figure 2.5: Flowchart of Artificial Bee Colony algorithm.[35]

proposed in the past,[16,36–38,109–119] however, the problem of back mapping is the key procedure as sometimes the global minimum of the smoothed surface can change dramatically by back mapping to the original surface. Therefore, not all approaches perform equally well depending on the quality of the back mapping.[6] The most important deformation techniques include the Diffusion Equation Method (DEM),[36] the Distance Scaling Method,[114] Monte Carlo with Minimization (MCM),[120] Basin Hopping (BH),[115] or the Potential Energy Smoothing and Search (PSS) method.[37,38] Basin Hopping seems to be one of the most widely used and most successful hypersurface transformation algorithms, since many complex global optimization tasks are efficiently solved by BH optimization.[7] The Basin Hopping approach is very efficient in sampling wide areas of the phase space, however, close lying basins maybe hard to localize. Therefore, several approaches exist which combine the Basin Hopping with a more local global optimization approach to obtain the best of both approaches. Examples are the Monte Carlo Multiple Minima/Low-mode sampling (MCMM/LMOD)[27,39–41] approach or the recently implemented Tabu-Search based algorithm[52] which is also described in this work. In the following the DEM, the PSS, the BH and the MCMM/LMOD approach are described in more detail. The new Tabu-Search algorithm as a part of this work is described in more detail later (section 3).

Diffusion Equation Method The Diffusion Equation Method (DEM) was first mentioned by Piela[16,36] and is one of the first deformation algorithms applied to conformational search. It employs a similar idea as the approach proposed by Stephenson and Binsch[121] who applied their method to the analysis of NMR data. The principle of DEM is a smooth deformation of the hypersurface to make shallow potentials disappear and eventually obtain one single minimum. The algorithm thereby assumes that shallower potentials will disappear more easily than deeper ones. As the final minimum on the transformed surface can deviate from the real global minimum of the original potential, the deformation is gradually reversed with local optimizations on each deformation step.

The first iteration of deformation of the original potential $f(x)$ can be obtained by adding its second derivative which is zero at inflection points (see Equation 2.3, β is a positive small constant). The Nth iteration will lead to the result shown in Equation 2.4, were $\frac{t}{N}$ is written instead of β.

$$f^{[1]}(x) = f(x) + \beta f''(x) = \left(1 + \beta \frac{d^2}{dx^2}\right) f(x) \tag{2.3}$$

$$f^{[N]}(x) = \left(1 + \frac{t}{N} \frac{d^2}{dx^2}\right)^N f(x) \tag{2.4}$$

For the limit of $N \to \infty$, the transformation operator $T(t)$ in Equation 2.5 is obtained. Equivalently, the transformation operator can be written as a diffusion equation (Equation 2.6) which is written for higher dimensions using the Laplace operator (Equation 2.7). An example application for a one-dimensional case is shown in Figure 2.6.

Recently, the DEM approach was also applied to crystallography[122] and Goldstein *et al.*

proposed a new hybrid algorithm for global optimization of peptides consisting of a DEM approach, a simulated annealing procedure and evolutionary programming.[118]

$$T(t) = \exp\left(t\frac{d^2}{dx^2}\right) \tag{2.5}$$

$$\frac{\partial^2 F}{\partial x^2} = \frac{\partial f}{\partial t} \tag{2.6}$$

$$\Delta F = \frac{\partial F}{\partial t} \tag{2.7}$$

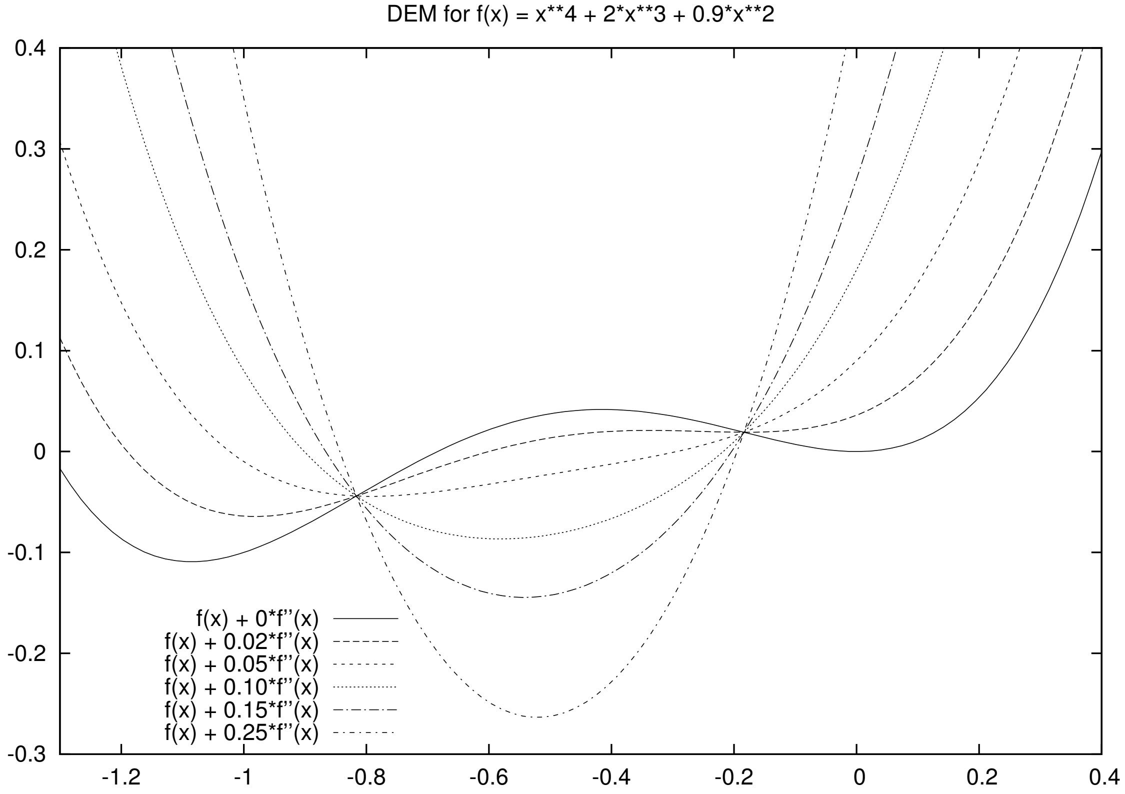


Figure 2.6: Illustration of the Diffusion Equation Method applied to $f(x) = x^4 + 2.0x^3 + 0.9x^2$. The original potential f(x) is deformed by addition of its second derivative. As it can be seen, for $\beta > 0.05$ only one minimum is left. Reverse deformation combined with local optimizations will lead to the global minimum of the original potential. Recreated following the publication of Piela.[36]

Potential energy smoothing and searching Ponder *et al.* proposed a new variant of the DEM-method which improves the performance of the original approach.[37,38] In the original DEM-method, it is assumed that the global minimum of the original potential energy surface is obtained by gradually reversing the deformation process coupled with local optimization to the closest minimum. However, it may happen for narrow basins that a wider potential is preferred by DEM and therefore simple local optimization will guide the DEM into other basins which are not the global minimum. The authors of DEM argued, that these narrow potentials are often not of chemical interest. In the new potential energy smooth and search (PSS) algorithm, the local optimization step is replaced by an approach which allows it to leave a local minimum and search for a neighboring minimum which is lower in energy. Thus, the final minimum after the PSS search is equal or lower to the one obtained by DEM. The new PSS algorithm was further compared to a molecular dynamics simulated annealing approach by its application to predictions of transmembrane helix packing. The PSS was superior in both computational efficiency and the accuracy in locating the final minimum.[38] An example application is *N*-Acetyl-Ala-Ala-*N*-Methylamide which investigated the similarity and possible correlations between PSS and simulated annealing.[117] In 2002, Grossfield and Ponder proposed an improved version of the PSS algorithm,[123] which employed a modified potential smoothing kernel. Instead of exponential functions as employed in the original smoothing procedure[37] in the new shifted-tophat or stophat approach, smoothing is employed by a hyper sphere.

Basin Hopping Currently, the Basin Hopping (BH) approach proposed by Wales[115] is a very widely used global optimization approach.[58,124–132] Its principle is based on the Monte Carlo with Minimization approach by Li and Scheraga.[120] The potential energy surface is transformed into a staircase-shaped surface by minimizing each random point to its next local minimum. This transformed surface is investigated by a Metropolis Monte Carlo algorithm. Although rather simple, the approach possesses a remarkable efficiency. Figure 2.7 shows a one-dimensional example of a transformed surface in the Basin Hopping approach. The different approaches differ mainly in the implemented step approach for Monte Carlo and the use of the coordinate system. For conformational search, internal dihedral coordinates have been shown to be most efficient.[7] Due to the hopping steps, the BH approach performs a non-local and widespread search of the phase space. Therefore, the efficiency of BH can be improved dramatically by combining it with a more local search approach, like done in MCMM/LMOD[27,39–41] where the Low Mode Search (LMOD) algorithm is used or a recent implementation of a Tabu-Search based algorithm described in the present thesis.[52]

Monte Carlo Multiple Minima - Low Mode Sampling MCMM/LMOD is implemented, for example, in the MacroModel program[133] and consists of two algorithms, the Monte Carlo Multiple Minima[27] and the Low-Mode sampling approach.[40,41] The MCMM approach employs a similar principle as MCM of Scheraga[120] although several differences exist. The strategy for choosing random steps is modified and new starting points are chosen by other criteria than in MCM. Furthermore, MCMM is designed to locate all low-lying con-

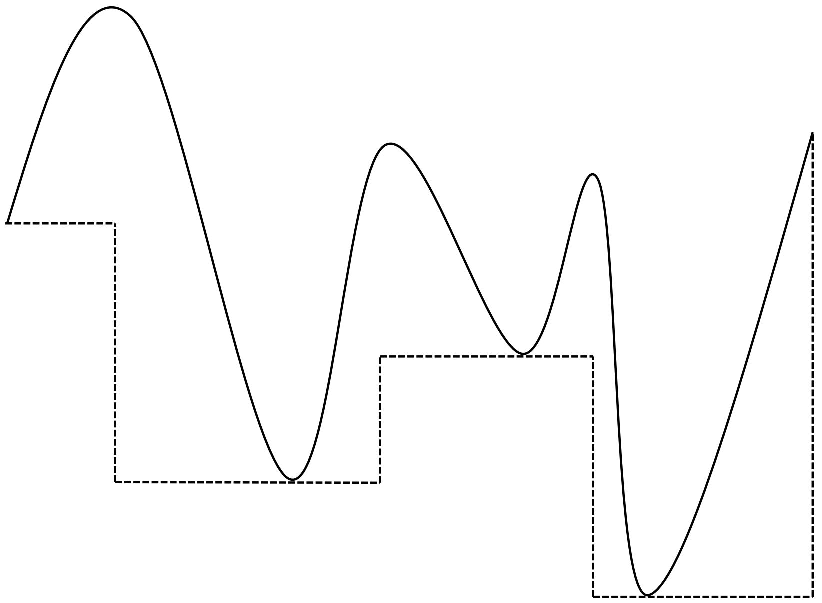


Figure 2.7: Illustration of the transformed surface within the Basin Hopping approach.

formers instead of only finding the global minimum. Special care is taken for ring structures in the molecules. A direct benchmark of MCM and MCMM revealed a better or similar performance of MCM for small systems, however, for bigger systems and for the location of other low-lying conformers, MCMM displayed a better performance.[27,39]

Low-Mode sampling is based on the mode-following concept for locating transition states. However, it employs a "brute-force" approach for simplification of the method.[40] The initial minimum structure is taken for a normal mode analysis and several low-frequency modes are stored. The number of stored eigenvalues is determined by a user-defined frequency threshold. All saved modes are systematically employed for searching by perturbing the initial minimum following the mode direction. LMOD follows the initial low-mode eigen vector linearly until the potential energy is bigger than a user-defined threshold. The final point is minimized using a local optimization algorithm. The initial normal-mode analysis is only performed once and, therefore, the Hessian matrix is only evaluated at the very beginning of the search. There is no guarantee, that the LMOD search ends in a new or better minimum. However, the authors claim that most of the time barriers are crossed by the proposed procedure.[40] Modifications of the LMOD procedure furthermore allow to follow the exact minimum energy path employing the original mode-following concept.[41] The new procedure (c-LMOD) allows for a more accurate conformational search, however, the computational costs also increase. For bigger systems such as peptides, the original LMOD procedure is recommended. For very large systems a completely Hessian-free low-mode search was developed (LLMOD) which provides significant performance improvements for searching the conformational space of proteins.[134]

In the combined MCMM/LMOD approach, once LMOD has searched the closest neighborhood for all low-lying conformers, the algorithm switches to a MCMM search to obtain new starting structures. From these, again a LMOD search is performed. Theses steps are repeated until user-defined stopping criteria (such as a maximum number of steps) are reached. First applications of the MCMM and LMOD approaches included conformational search[27,39–41] and docking, especially flexible docking.[41,135] Nowadays, the combined MCMM/LMOD is widely used. Several conformational search studies on various systems have been presented[136–143] like enzyme inhibitors,[137,141] silsesquioxanes,[138] maleimide systems,[139,140] or on model systems for the Brevetoxin A.[142] Parish investigated the performance of MCMM and LMOD and found that, while for a small systems both the single algorithm and several combinations perform equally well, the best method for larger systems is a hybrid consisting of a 50:50 combination of both algorithms.[136] Further application areas cover docking,[144] binding mode analysis,[145,146] molecular modeling of enzyme inhibitors,[147] and investigations on fluorescent proteins.[148] Li *et al.* employed the MCMM/LMOD approach to create starting structures of proteins where no crystal structure is available. They started from a very similar protein and punctually mutated single amino acids and combined this with conformational search to obtain a final structure of the desired protein.[149]

2.1.5 Stochastic algorithms

Most of the algorithms already described contain probabilistic or stochastic elements. The difference of the "stochastic" algorithms described in this section in comparison to the already mentioned methods is the lack of an inspiring nature system or a metaphorical explanation. The inspiring or metaphorical description is mostly used to classify the algorithms above (Particle Swarm optimization, surface deformation,...). Therefore, the following algorithms are simply summarized as "stochastic" algorithms.[8,18] Important representatives are the Scatter Search[42] or the Tabu-Search[43,44] with recent modifications.[52,150–155]

Scatter Search Scatter Search was first introduced by Glover in 1977[42] and was designed as a heuristic for solving integer programming problems. Scatter Search is sometimes referred to evolutionary computations and contains similar aspects as Tabu-Search.[18] The main objective of Scatter Search is the maintenance of a set of both diverse and high-quality solutions. The method stayed idle until Glover revised it in 1998 by publishing a template for Scatter Search[156] introducing the five methods which are essential for the efficiency of scatter search. Further advanced features in scatter search are obtained by the way of implementation of the five methods.

1. **Diversification Generation Method**: Generation of a collection of diverse trial solutions starting from a random trial or seed solution.
2. **Improvement Method**: Transformation of a trial solution into one or more enhanced solutions. It is not essential that either input or output is a feasible solution. However,

the output is usually expected to be so. If no enhanced solution can be obtained, the output will be the same as the input.

3. **Reference Set Update Method**: Building and maintenance of a reference set consisting of the best found solutions for providing efficient access from other parts of the algorithm. The goal is to ensure both diversity and high-quality solutions.

4. **Subset Generation Method**: Generation of subsets of the reference solutions as a basis for creation of combined solutions.

5. **Solution Combination Method**: Using the output of the Subset Generation Method for transformation into one or more combined solutions.

Summarizing the five methods described above, the principle of Scatter Search is to exploit useful information about the global optimum from a set of diverse and elite solutions stored in the reference set. The search starts by setting up an initial reference set of solutions as diverse as possible but also feasible. New information can be exploited by recombination of members of the set. The iterative process employed in Scatter Search partitions the reference set into subsets which are successively recombined. The recombined solutions are ranked whether they stay in the reference set or not. The iterations are repeated until some convergence criteria are met.
Scatter Search is a topic of current research and several possibilities for advanced Scatter Search algorithms are proposed.[157–159] Scatter Search is nowadays applied to non-linear multi-objective optimization problems,[152,160–164] global optimization of computationally expensive dynamic models,[165] or 3D image registration.[166] Besides, Scatter Search is applied to vehicle routing problems with time windows.[167] Although Scatter Search already provides a good performance, it is further improved when combined with tabu search methodologies.[167] Caballero *et al.* also reported a hybrid consisting of Scatter and Tabu-Search.[162] As Scatter Search is very close to genetic programming and Tabu-Search, explicitly assignable application examples are rare and fuzzy.

Tabu-Search The Tabu-Search algorithms are described in more detail in section 3.5.

2.1.6 Other approaches

Conformational Space Annealing The Conformational Space Annealing (CSA) was developed by Lee, Scheraga, and Rackovsky in 1997.[45] It was designed as a new algorithm for the efficient determination of the global minimum conformation as well as further low energy structures close to the GM.[45] It combines essential aspects from the buildup procedure proposed earlier[168,169] and genetic algorithms.[170] The flowchart of the algorithm can be found in Figure 2.8.

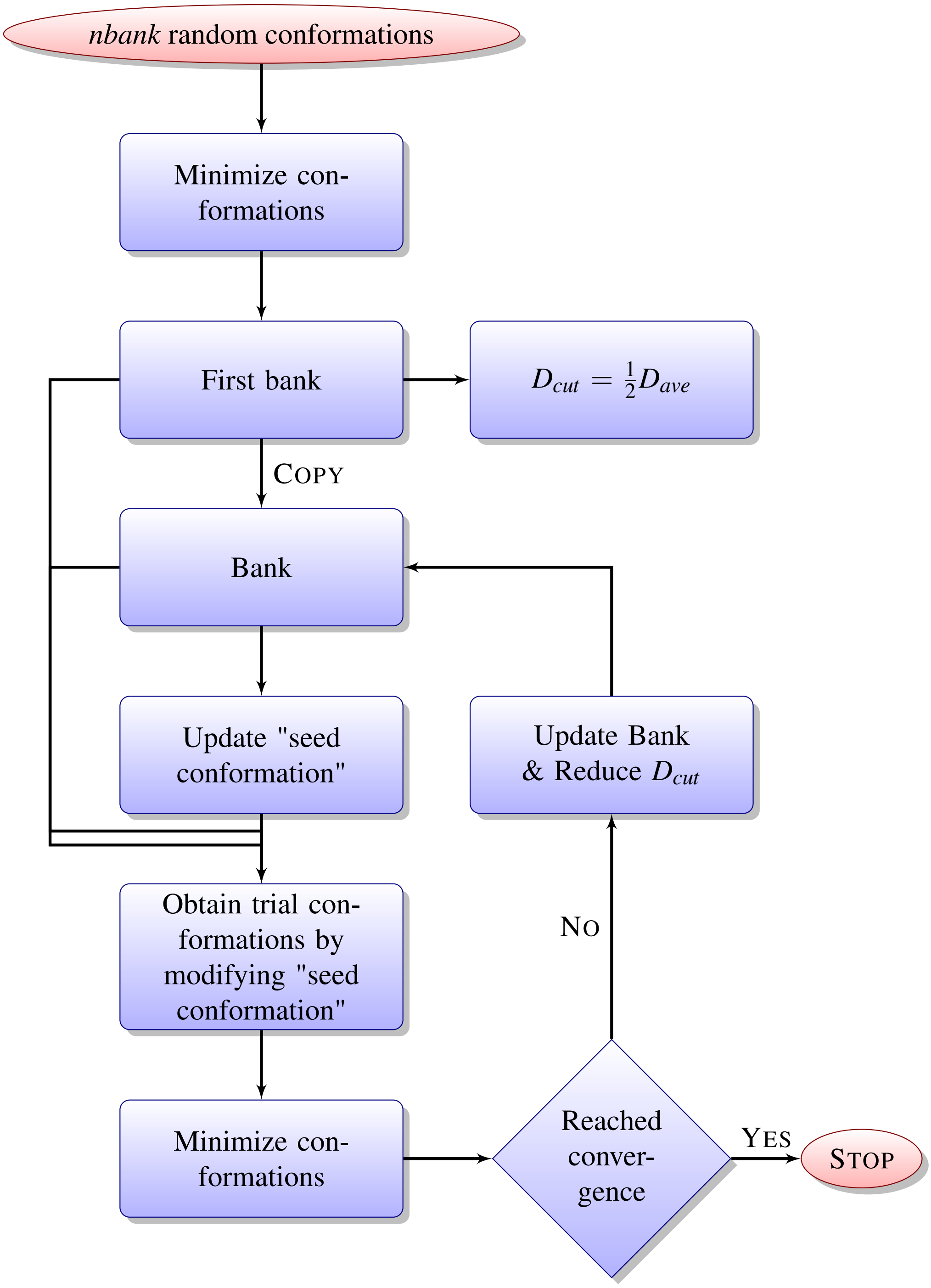

nbank random conformations
Minimize conformations
First bank
$D_{cut} = \frac{1}{2}D_{ave}$
COPY
Bank
Update "seed conformation"
Update Bank & Reduce D_{cut}
Obtain trial conformations by modifying "seed conformation"
NO
Minimize conformations
Reached convergence
YES
STOP

Figure 2.8: Flowchart of the Conformational Space Annealing.

The method is initialized with a number (e.g. 50) of randomly distributed conformations which is similar to the initial population in a genetic algorithm. The energy minimized structures are summarized in the so-called "bank". The conformations in the bank are refined in a way to cover the largest possible conformational space with coincidently lowest possible energy. Each conformation in the bank can be seen as a representative of a group of local minima inside a certain cutoff radius (D_{cut}). The total number of groups (here 50) is never changed. There are two possible operations on the groups. Firstly, each representative of a group can be replaced by a conformation of the same group with a lower energy. Secondly, once a new group is found with a lower energy than one of the already saved groups, it replaces the highest energy group. The value D_{cut} roughly defines the size of a group. Therefore, each conformation has to be checked for its distance D_{ij} (Equation 2.8) to all other conformations to decide whether it already belongs to a group or not.

$$\begin{aligned} D_{ij} = \sum_{k=1}^{n} \min \Bigg[& \operatorname{mod}\left\{ \left(\theta_k^i - \theta_k^j\right), \operatorname{sym}(k) \right\}, \\ & \left\{ \operatorname{sym}(k) - \operatorname{mod}\left\{ \left(\theta_k^i - \theta_k^j\right), \operatorname{sym}(k) \right\} \right\} \Bigg] \end{aligned} \tag{2.8}$$

$\operatorname{sym}(k)$ defines the symmetry of a dihedral angle and is set to 360°, 180°, and 120° for dihedral angles with no symmetry, twofold symmetry, and threefold symmetry, respectively.

In each step, D_{cut} is reduced by a certain factor which introduces the principle of *conformational space annealing*. New trial conformations are created by updating the seed conformations first. To introduce a search as diverse as possible, seeds are selected far apart from each other. Then, information is combined from the first bank, the current bank and the seed conformations to obtain new structures. The combination towards new structures is done by replacing randomly chosen dihedral angles from the different reference data sets.

The principle of starting from a initial population and creating new trial solutions by the recombination of earlier structures provides parallels to genetic algorithms. The creation of new conformations by recombination of dihedral angles shows the similarities to the buildup procedure. The search is stopped once a certain convergence criterion is reached like a final value of D_{cut} or a previously defined global minimum energy.[45,171] The CSA algorithm is widely and efficiently used in the field of conformational search and protein structure prediction.[5,45,171–190] Due to its nature of constructing a set of conformations which are further investigated and refined, it can be parallelized very efficiently.[173] Several investigations have been published that investigate protein structures and folding behaviors employing CSA.[45,171–173,176–179,183,185–187] In its first application[45] it was already shown that CSA is superior to other algorithms such like MCM[120] or the electrostatically driven Monte Carlo.[191,192] The great efficiency of CSA for protein structure prediction can also be seen from the latest Critical Assessment of Protein Structure Prediction (CASP9, 2010), where four approaches took part which were based on CSA.[193] However, several further application areas of CSA, beside protein structure prediction, can be found. Lee *et al.* employed the CSA algorithm to optimize the parameters of a potential energy function with respect to results of the Critical Assessment of Protein Structure Prediction (CASP) 3 and 4 from 1998 and 2000, respectively.[174] Conformational search is also of great importance for molecular modeling and docking. In 2005, the CSA algorithm was already applied to docking problems employing the Tinker program package comparing the performance to MCM.[180] For smaller search problems, MCM and CSA exhibit similar performance. For systems with a more complex search space, the CSA algorithm becomes more efficient.[180] Recently, the CSA algorithm was also implemented into AutoDock[188], which allows for very efficient docking studies, and into the Charmm program package.[186] The structural characteristics of compstatin, a peptide based inhibitor, was investigated by Song *et al.*.[181] Several further studies investigated protein-protein interactions,[182] multiple sequence alignment, an essential process in bioinformatics,[184] sampling of flexible protein loops,[190] or the determination of structures from NMR assisted by CSA.[189] The generality of the CSA approach was shown in a study of Lennard-Jones clusters up to the size of 201, where all known low lying energy structures were determined by CSA. The search was performed without any further restriction of the global optimization such as knowledge of the final global minimum.[175] Together with the above mentioned studies, CSA seems to be a generally applicable global optimization algorithm with high efficiency.

StartOpt As part of the Conformational Analysis and Search Tool (CAST) described later in this work (subsection 4.1), the StartOpt[46] subroutine is implemented for the preparation of reasonable starting structures. StartOpt is based on earlier work in the diploma thesis of Christoph Grebner, where the RingSearch procedure was first implemented.[46] The efficiency of this approach was shown in a comparison of different conformational search algorithms (Simulated Annealing, Molecular Dynamics with Minimization, Monte Carlo with Minimization, and a Tabu-Search based algorithm), where different starting structures were used. Employing structures provided by RingSearch, the efficiency of the conformational search algorithms could be improved significantly.[52] The RingSearch algorithms uses chemical intuition and searches the given structure for possible five-, six-, or seven-membered ring structures build-up through hydrogen bonds. Thus, the conformational space can be efficiently prescanned with an inexpensive approach. Intra-molecular ring structures are especially important for peptide systems, where the backbone system can form seven-membered rings.[46] Motivated by the efficiency of the RingSearch procedure, the FOLD and the SolvAdd algorithm were developed during the diploma theses of Johannes Becker[47] and Daniel Weber[48], respectively, and implemented into the CAST program. A performance benchmark of FOLD and SolvAdd is subject to future research.

2.2 Reaction path determination

The reliable description of molecular systems and their reactions requires knowledge about characteristic points of the underlying energy landscape. Besides the global and local minima, saddle points of first order are the most important stationary points. Minima correspond to stable arrangements on the hypersurface while transition states (saddle points of first order)

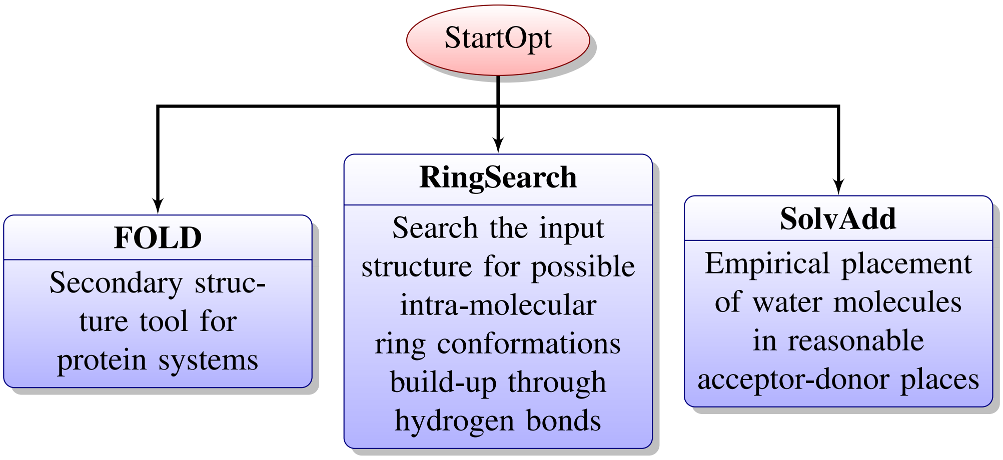

StartOpt
FOLD
Secondary structure tool for protein systems
RingSearch
Search the input structure for possible intra-molecular ring conformations build-up through hydrogen bonds
SolvAdd
Empirical placement of water molecules in reasonable acceptor-donor places

Figure 2.9: The StartOpt algorithm developed in the Engels group. Currently, three subroutines are available: RingSearch,[46] FOLD,[47] and SolvAdd.[48]

describe the energetically highest points in a transition from one minimum to another.[7,194]
In the previous section, a variety of algorithms was reviewed to determine local and global minima. In this section, the focus is set on the description and investigation of transition states and reaction mechanisms. Mathematical details can be found in subsection 3.4.

When investigating transitions in complex systems like large clusters or proteins, the underlying reaction mechanisms can be very complicated. In the easiest case, two minima (reactant and product state) are connected via one single transition state. However, usually several different transition states lie between the initial and final state. Furthermore, it is possible that several distinct pathways exist each having similar reaction rates.[7,194,195]

For transitions with one single intermediate transition state, usual transition state search algorithms can be used. They are divided into single-ended and double-ended approaches. Widely used and very efficient single-ended algorithms are e.g. the eigenvector-following approach proposed by Cerjan and Miller[196] and improved by Wales[197] and the Dimer-method proposed by Henkelman[198] and improved by Heyden and Kästner.[199,200] These methods are used when the final point of the transition is not known. Providing an initial guess of the transition state, they converge to the closest saddle point of first order.

When both, the reactant and product state, are known and the connecting minimum energy path (MEP) is searched usually double-ended search methods are employed. Commonly used methods include the growing string approach[201] or the Nudged Elastic Band (NEB) method[202] with recent improvements.[203–206] The double-ended methods, also called chain-of-state-methods, use a series of images to represent the minimum energy path. Usually, the resulting transition state has to be further optimized for example by using the eigenvector-following or Dimer method.[207]

However, for more complex reactions where several intermediate minima and transition states exist the situation becomes much more difficult. This was also nicely described by Kadanoff who classified such systems as having "many chaotically varying degrees of freedom interacting with each other".[208] Here, the above mentioned simple approaches are hard to apply. Examples are rearrangement reactions of cluster systems,[209–211] the folding of proteins,[124,195,207], enzymatic reaction mechanisms,[212,213] or reactions of crystalline complexes.[214,215]

One of the first methods for investigating complex transitions, especially rare events, is the transition path sampling (TPS) developed by Dellago in 1998.[209,216–218] TPS can be summarized as "throwing ropes over rough mountain passes, in the dark".[218] It can be used to study rare events without the requirements of knowing the mechanism, transition state, or reaction coordinates. It is based on a biased Monte Carlo sampling also called importance sampling which focuses on chain of states creating dynamical trajectories rather than upon individual states. The importance sampling is hereby generalized to trajectory space to create transition path sampling. "Throwing ropes" can be seen as shooting short trajectories, whereas "in the dark" corresponds to the very complex landscapes under investigation.

Another well-known approach is the discrete path sampling (DPS) proposed by Wales in 2002.[210] DPS is a two-state method with an initial and a final state. The transitions are characterized as connected sequences of minima and saddle points of first order. After an initial global optimization of the energy landscape, a database of minima is created. This database is used to incrementally connect the initial and the final state. The connection of two minima is done via an implementation of the doubly nudged elastic band.[205] Essential steps are grouping the minima into states belonging either to the initial, the final, or an unconnected state and the way the shortest path is chosen.[207] Candidates for transition states are tightly converged by using an hybrid eigenvector-following approach.[219,220] The output of DPS is a database of pathways via local minima and transition states. These information can be used to calculate thermodynamic and kinetic properties.[210] The DPS was successfully applied to rearrangements of cluster systems,[210,211] free energy and dynamic calculations of met-enkephalin,[195] or the folding behavior of proteins.[124,207,221,222]

Besides, many other approaches exist to investigate transition pathways. Parrinello proposed the metadynamics algorithm[214,223,224] and Grubmüller developed a conformational flooding approach.[225] Voter proposed the parallel replica exchange dynamics for extending the time scale of usual molecular dynamic simulation and thus being able to investigate infrequent events.[226] Teresa Head-Gordon presented a new sampling algorithm based on the determination of unique substates from instantaneous normal modes.[227] Further recent developments are κ-dynamics,[228] dominant reaction pathways,[229] locally scaled diffusion maps,[230] or the use of support vector machine for optimizing transition states.[231]

Very recently, Martin Jansen presented the prescribed path method for investigating energy landscapes.[215] The prescribed path algorithm is based on a sampling starting from a predefined path (the prescribed path). In particular, small barriers orthogonal to the reaction coordinate can be cross allowing for the determination of several relaxed pathways.

In this work, a new algorithm was developed which is described in more detail in section

4.4. Its principle is similar to the prescribed path method and the discrete path sampling, however, the algorithmic details are different. The new PathOpt algorithm is based on global optimization in a (n-1) dimensional hyperplane (n being the number of search variables). The hyperplane is perpendicular to the initial reaction coordinate. Thus, minima on this reduces hyperplane correspond to traces to transition states from reactant to product state. By optimizing these points to the closest saddle point of first order (only one negative eigenvalue of the Hessian), transition states lying between the two initial states can be found. Due to its close relationship to DPS or the prescribed path method further improvements might be gained by combining these approaches.

3 Theory

3.1 Energy Landscapes

The investigation of the structural and dynamical behavior of complex chemical systems, such as large clusters or biomolecules, strongly correlates with the investigation of the underlying potential energy function (PES). The exact description is very hard, however, predictions can be made using knowledge of the stationary points of the PES. Stationary points of a PES are points with a vanishing gradient such as minima, maxima, or saddle points. Methods which investigate these points can be summarized as energy landscape methods. The concept of energy landscapes was first proposed by Bryngelson and Wolynes[12] in the context of free energy surfaces, and a detailed summary on energy landscapes is given in the textbook of David J. Wales.[7] The most important stationary points of a (potential) energy landscape are local minima and maxima, the global minimum and maximum, as well as transition states (first order saddle points). An illustration of a multi-dimensional PES is given in Figure 3.1.On the one hand, the behavior of the energy landscape is defined by the system under investigation. On the other hand it depends on the underlying theoretical model for the energy calculations. More details on possible models are given in subsection 3.2. To identify the various points of the PES, different algorithms and approaches are required. Local extrema are located using local optimization algorithms (subsection 3.3), and first order saddle points can be found using transition state search algorithms described in subsection 3.4. Finally, the global extrema are investigated employing global optimization methods. Here, the focus is set on Tabu-Search algorithms (subsection 3.5). An overview of global optimization algorithms was given in the introduction (subsection 2.1).

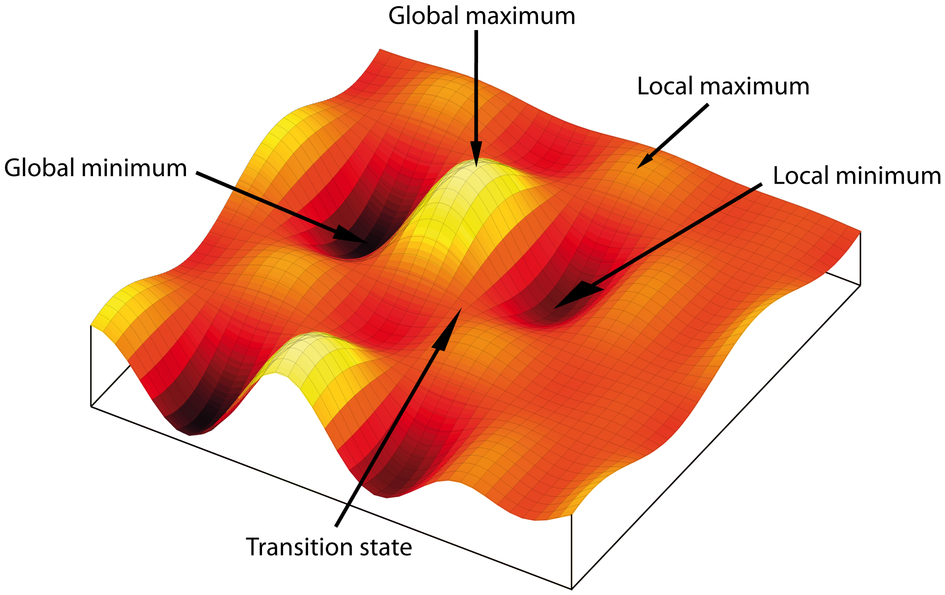


Figure 3.1: Illustration of a multi-dimensional potential energy surface with important stationary points.

3.2 Theoretical models

3.2.1 General aspects

As already described, the shape of the potential energy surface or energy landscape strongly depends on the employed theory. The basic idea of the visualization of a potential energy surface depends on the possibility to separate different degrees of freedom. The most important is the separation of electron and nucleus motion, which is also known as the Born-Oppenheimer approximation. This leads to the electronic Schrödinger equation which can only be solved approximately.[19] These *ab initio* or *first principles* calculations include electronic properties. Models range from semi-empiric methods like PM3, MNDO, or AM1, through Hartree-Fock, Density Functional Theory, Perturbation Theory - methods, Coupled Cluster methods, as well as multi reference approaches. However, in order to produce a surface as a function of nuclear coordinates, a grid of function values in the configurational space with succeeding fitting or interpolation is required. *Ab initio* dynamics alternatively calculate the required potential energy 'on-the-fly'. However, such approaches are still not feasible for large and complex systems with more than a few hundreds of atoms as the computational effort is far too big. Therefore, less accurate, mostly empirical, methods are applied. These force field methods are usually the theory of choice for energy landscape investigations of very complex systems. They employ classical mechanics and Newton's second law (equation of motion). In principle, force field methods (or molecular mechanic methods) can be described by a Ball and Spring model. They use "simple" equations where the parameters are fitted to experimental or theoretical data. These methods are very fast, but usually, the accuracy is only guaranteed for the systems they are parametrized for. New and unknown systems might be treated very badly. Furthermore, no bond breaking can be treated in general.[7,19]

The classical force fields (class I) only contain bonded terms (bond, angle, and torsional equations) as well as van der Waals and Coulomb interactions. Important representatives are the OPLS-AA,[232–235] Amber,[236–238] and the Charmm[239–242] force field. More accurate force fields also include cross terms (e.g. the MM3 force field[243–247]), polarizabilities (e.g. the AMOEBA force field[248–250]), or further terms. They are summarized as class II and III force fields. To reduce the computational effort, for example in protein calculations, united atom approaches are present (e.g. GROMOS[251–253]) where unpolar hydrogen atoms are not explicitly treated, but are included in the connected atoms (e.g. the carbon atom). Very big systems such as complete viruses can be treated by coarse grained force fields (e.g. the MARTINI forcefield[254–256]). In the next subsections, the most important energy and derivative terms of classical force fields are described in more detail.

3.2.2 MM-Energy terms

In general, the force field energy terms can be divided into bonded and non-bonded interactions. The non-bonded interactions can be further split into van der Waals (vdW) and electrostatic terms (see Figure 3.2).

$$E_{FF} = E_{bonded} + E_{non-bonded} \tag{3.1}$$

$$= E_{bond} + E_{angle} + E_{imp} + E_{torsion} + E_{vdw} + E_{el} \tag{3.2}$$

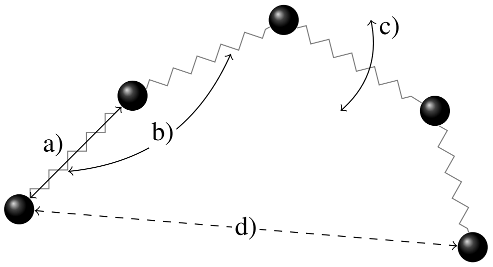


Figure 3.2: Illustration of existing interactions in a force field.
a) bond stretching, b) angle bending, c) torsional bending, d) non-bonded interactions.

The classical force fields (class I) only comprise the charges for electrostatic descriptions. The most famous class I force fields for biological systems are OPLS-AA, Charmm and Amber. On the whole, they are comprised of the formulas described in the following subsections. There may be smaller deviations, but the general structure is the same for these force fields.

More accurate force fields try not only to reproduce geometries or relative energies, but also vibrational frequencies (class II force fields). Further improvements allow parameters to depend on neighboring atoms (modeling of hyper conjugation) and include polarization effects (class III force fields). Important force fields are the MM2[257,258] and MM3[243–247] force fields, MMFF94[259–264] or the polarizable force field AMOEBA[248–250].

3.2.2.1 ***E_{bond}* terms[19,49]** The bonded energy terms, also called stretch energy, describe the stretching of a bond between two atoms A and B. There are several ways to describe this function. The easiest approach is to write the energy term as a Taylor expansion around an equilibrium distance R_0.

$$E_{bond} = E_0 + \frac{dE}{dR}(R_{AB} - R_0) + \frac{1}{2}\frac{d^2E}{dR^2}(R_{AB} - R_0)^2 \tag{3.3}$$

In most classical force fields, the Taylor expansion is terminated at the second order. The term E_0 is set to zero (constant shift of the energy). The second term becomes zero as stretching is done around the equilibrium distance and the derivation of the energy with respect to the distance R becomes zero.

This leads to a harmonic potential (Equation 3.4), which is often a sufficient approximation to equilibrium geometries.

$$E_{bond} = k^{AB}\left(R_{AB} - R_0\right)^2 = k^{AB}\left(\Delta R_{AB}\right)^2 \tag{3.4}$$

Of course, the accuracy can be enhanced by including higher order terms as well, however, at the cost of increased calculation time as well as more parameters that have to be fitted. Furthermore, polynomial functions do not have the right energy behavior for larger distances. Polynomial functions either become $+\infty$ or $-\infty$ for R going to $+\infty$. However, the right behavior is convergence towards a particular value, the dissociation energy. A simple function satisfying this criterion is the Morse potential shown in Equation 3.5.

$$E_{bond} = D\left(1 - e^{-\alpha\Delta R}\right)^2 \tag{3.5}$$

$$\alpha = \sqrt{\frac{k}{2D}} \tag{3.6}$$

Here, D is the dissociation energy and α is related to the force constant.
The Morse potential describes the behavior of the stretch energy quite accurately over a broad distance range. For longer distances, the force resulting from the Morse potential is relatively small. This can lead to a slow convergence to the equilibrium bond length in geometry optimizations or simulations. Since the polynomial functions describe the systems quite well near the equilibrium distance, many force fields employ the much simpler polynomial functions.

3.2.2.2 E_{angle} **terms[19,49]** The energy terms of E_{angle} describe the bending by an angle θ formed by three bound atoms A, B, and C. Like E_{bond}, E_{angle} is often described by a Taylor expansion around an equilibrium angle which is terminated after the second order (Equation 3.7).

$$E_{angle} = k^{ABC}\left(\theta_{ABC} - \theta_0\right)^2 \tag{3.7}$$

This description is usually very accurate. Higher accuracy is obtained by including higher order terms.

3.2.2.3 E_{imp} **terms[19,49]** When a central atom B is surrounded by three atoms A, B, D (i.e. a trigonal center), an improper torsion or out-of-plane term has to be defined. This term is used to describe the pyramidalization of the trigonal center. Furthermore, the term creates a barrier for inversion. Usually, the force constants have to be very high to describe the barriers properly. The improper torsion energy can be described by a harmonic potential, where either the out-of-plane angle χ or the distance of the central atom B to the plane ABD is used (see Equation 3.8 and Figure 3.3).

$$E_{imp}(\chi) = k^B \chi^2 \text{ or } E_{imp}(d) = k^B d^2 \tag{3.8}$$

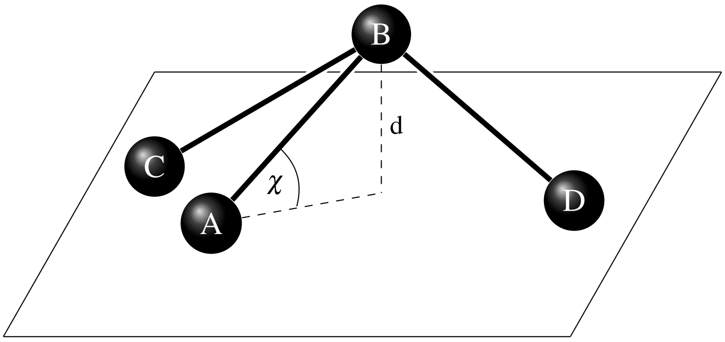


Figure 3.3: Illustration of an improper torsion.

3.2.2.4 $E_{torsion}$ **Terms**[19,49] The torsional energy term describes the rotation around a bond B-C embedded in a sequence of four atoms A-B-C-D. Here, atoms A-B, atoms B-C, and atoms C-D are bound to each other. The torsional angle ω is defined as the angle between the plane ABC and BCD (see Figure 3.4). In contrast to E_{bond}, E_{angle}, and E_{imp}, the torsional energy has to be periodic in the angle ω (after 360°, the same value has to be returned). Furthermore, the energy needed to distort a torsional angle is rather small and large deviations from the minimum structure can occur. Therefore, a Taylor expansion is not a good solution. To comprise both aspects, a Fourier series is defined:

$$E_{torsion} = \sum_{n=1}^{k} V_n \cos n\omega \tag{3.9}$$

The variable n gives the kind of rotation (e.g. $n = 1$ is a rotation around 360°, $n = 2$ is a rotation periodic to 180° and so on), V_n gives the size of the rotation barrier. The combination of the n terms can describe different energy profiles. A rotation in ethane, for example, is periodic by 120°. As all hydrogen atoms are equal, all minima posses the same energy. The Fourier series can only include $= 3, 6, 9, ...$ terms. In contrast, a rotation in ethene is periodic by 180°, i.e. only terms with $n = 2, 4, ...$ can occur. Looking at the rotation around the central bond of the butane molecule, three minima will occur. However, the two *gauche* and the *anti* conformation as well as the barriers separating the minima have different energies. This can be described by introducing an $n = 1$ term. Further situations can be described by different combinations of terms as well as different force constants V_n. Most force fields include terms up to $n = 3$ which are normally sufficient enough to describe most situations. Some force fields like Charmm, however, also comprise higher order terms ($n = 4$ or $n = 6$)

Usually, the zero point of the Fourier series is shifted by adding a factor of one, leading to the very popular formula for torsional angles used in most force fields (Equation 3.10). The $+$ and $-$ signs are chosen to comprise several properties of the terms. For $n = 1$, the term has a minimum at 180°. The $n = 2$ expression contains two minima, one at 0°, the other at 180°. The three-fold rotational term ($n = 3$) has three minima at 60°, 180°, and 360°.

$$E_{torsion} = \frac{1}{2}V_1\left[1+\cos(\omega)\right] + \frac{1}{2}V_2\left[1-\cos(2\omega)\right] + \frac{1}{2}V_3\left[1+\cos(3\omega)\right] \tag{3.10}$$

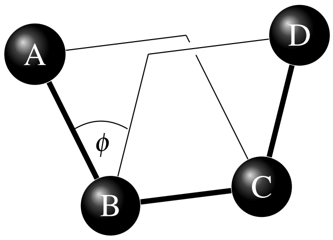


Figure 3.4: Illustration of a torsion angle.

3.2.2.5 $E_{non-bonded}$ **Terms[19,49]** As mentioned above, the non-bonded energy terms can be divided into van der Waals and electrostatic interactions. The non-bonded energy terms are the computationally most intensive interactions. The consideration of only pairwise interactions will yield a nearly quadratic dependence on the number of atoms. For very big systems like proteins, this cannot be calculated completely due to computational costs and memory problems. Therefore, cutoff distances are often included.

The simple description of the van der Waals and electrostatics terms (Lennard-Jones and Coulomb law, see subsections below) show a wrong behavior with respect to the intermolecular distance. However, error cancellation leads to a qualitatively correct total energy. To improve the distance behavior, more accurate terms are included. The following subsection discusses possible expressions for vdW and electrostatic energies.

van der Waals energies The van der Waals energy describes a repulsion or attraction between two unbound atoms which is not caused by (atomic) charges. E_{vdW} is zero at very large distances and goes to infinity for short distances. The latter can be explained in quantum chemistry by the overlap of two electron clouds. The negatively charged electrons cause a strong repulsion.

Due to induced dipole-dipole interactions there is a weak attraction between two atoms. Theoretically derived, this attraction is proportional to the inverse sixth power of the distance of two atoms. Of course, there are also higher order interactions, like induced dipole-quadrupole, quadrupole-quadrupole, etc., interactions. The force associated to this potential is also called "London" or "dispersion" force. See Figure 3.5 for an illustration. The repul-

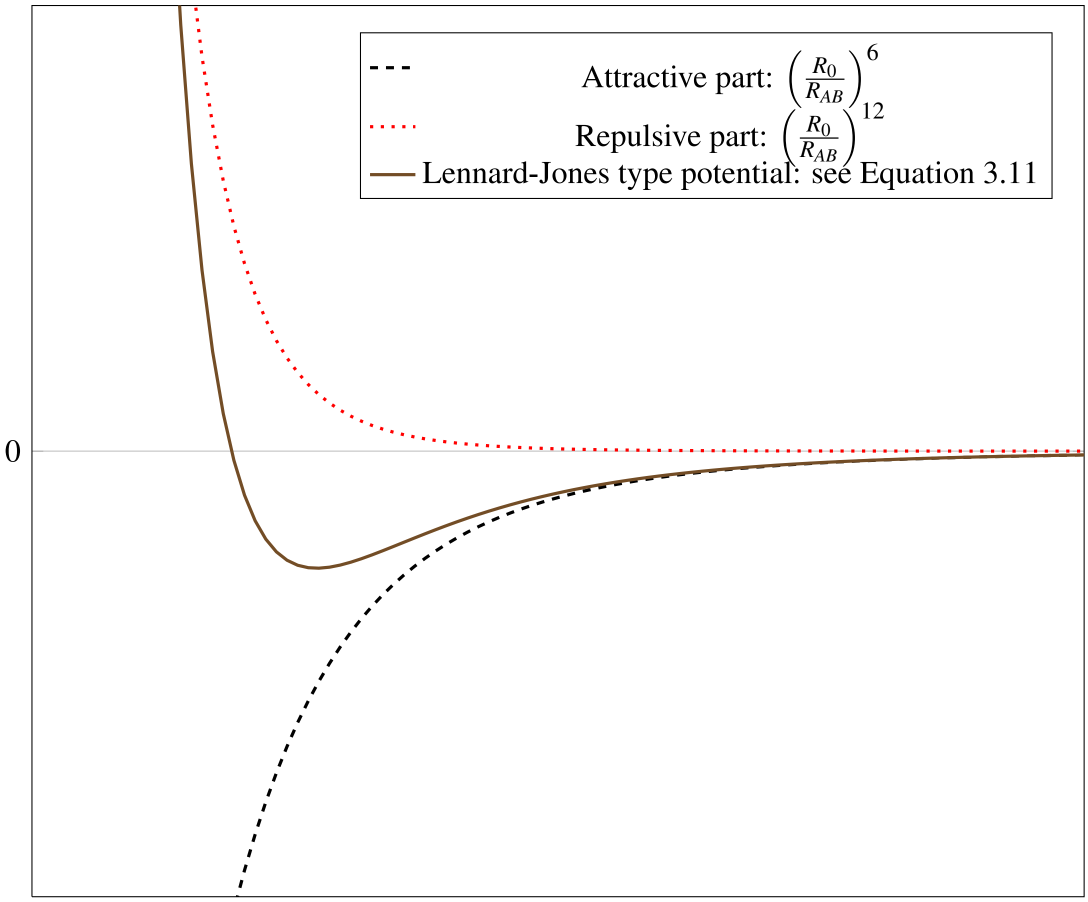


Figure 3.5: Repulsive and attractive part of a Lennard-Jones potential as well as the combined function (see Equation 3.11).

sive part cannot be derived theoretically. Like the attractive part, it should approach zero for $R \to \infty$, with the exception of approaching zero faster. This leads to a very popular function for E_{vdW}, the Lennard-Jones (LJ) potential (Equation 3.11)

$$E_{vdW-LJ} = \varepsilon \left[\left(\frac{R_0}{R_{AB}} \right)^{12} - 2 \left(\frac{R_0}{R_{AB}} \right)^{6} \right] \tag{3.11}$$

R_0 is the minimum energy difference and ε is the depth of the minimum. The exponent of the repulsive part is chosen to be 12 for computational advantages. In fact, other exponents are shown to be better, but the LJ-potential is computationally the most convenient.

Other approaches, like the buffered 14-7 (Equation 3.12) or the Buckingham potential (Equation 3.13) have a better description of the vdW interaction, but are also computationally more demanding. If very accurate results are necessary, Morse potentials can deliver better results than Buckingham potentials.

$$E_{vdW-buff-14-7} = \varepsilon \left(\frac{1.07R_0}{R+0.07R_0} \right)^7 \left(\frac{1.12R_0^7}{R^7+0.12R_0^7} - 2 \right) \tag{3.12}$$

$$E_{vdW-Buckingham} = \varepsilon \left[\frac{6}{\alpha-6} e^{\alpha\left(\frac{1-R}{R_0}\right)} - \frac{\alpha}{\alpha-6} \left(\frac{R_0}{R} \right)^6 \right] \tag{3.13}$$

All potentials given above depend on a minimum interatomic vdW-distances of the two atoms A and B as well as the dielectric constant ε. There are several combining rules. Usually, the dielectric constant is chosen as the geometrical mean. The vdW distance is either taken as the sum of the two atomic vdW-distances (Charmm) or as the geometric mean (OPLS-AA), depending on the used force field (see Equations 3.14 to 3.16).

$$R_0^{AB} = R_0^A + R_0^B \tag{3.14}$$

$$\text{or: } R_0^{AB} = \sqrt{R_0^A R_0^B} \tag{3.15}$$

$$\varepsilon^{AB} = \sqrt{\varepsilon^A \varepsilon^B} \tag{3.16}$$

Electrostatic interactions In the classical force fields, the electrostatics are treated by point charge interactions of positively and negatively charged atoms (Coulomb energy, Equation 3.17). Another approach is the description by a dipole moment (Equation 3.18), where the χ is the angle between the two atomic dipole vectors $\vec{\mu}_A$ and $\vec{\mu}_B$ and α_A or α_B is the angle of $\vec{\mu}_A$ or $\vec{\mu}_B$, respectively, with the distance vector of A and B. The standard biomolecular force fields Amber, Charmm, or OPLS-AA use point charges, whereas the MM2 or MM3 force fields use dipole moments.

$$E_{el-charge} = \frac{q_a q_b}{\varepsilon R_{AB}} \tag{3.17}$$

$$E_{el-dipole} = \frac{\mu_A \mu_B}{\varepsilon \left(R_{AB}\right)^3} \left(\cos\chi - 3\cos\alpha_A \cos\alpha_B \right) \tag{3.18}$$

These two simple approximations of the electrostatic interaction energy perform quite similarly, provided that the parameters are fitted properly.

If more accurate descriptions are desired, multipoles and polarizabilities have to be included as well. An example of a force field which includes such terms is the AMOEBA (Atomic Multipole Optimized Energetics for Biomolecular Applications) force field by Jay Ponder.[248–250] Van der Waals interactions are implemented by a buffered 14-7 based functional. The electrostatic interactions are described by a permanent part and an electronic polarization (i.e. induced) part. The permanent atomic multipoles include monopole (charge), dipole, and quadrupole moments.

$$M_i = \left[q_i, \mu_{ix}, \mu_{iy}, \mu_{iz}, Q_{ixx}, Q_{ixy}, Q_{ixz,\ldots,Q_{izz}} \right]^t \tag{3.19}$$

Here, q_i is a point charge located at the atomic center i, μ is a dipole, and Q is a quadrupole. The interaction energy between two atoms i and j separated by the distance r_{ij} is given by $U_{elec}^{perm} = M_i^t T_{ij} M_j$ with:

$$T_{ij} = \begin{bmatrix} 1 & \frac{\partial}{\partial x_j} & \frac{\partial}{\partial y_j} & \frac{\partial}{\partial z_j} & L \\ \frac{\partial}{\partial x_i} & \frac{\partial^2}{\partial x_i \partial x_j} & \frac{\partial^2}{\partial x_i \partial y_j} & \frac{\partial^2}{\partial x_i \partial z_j} & L \\ \frac{\partial}{\partial y_i} & \frac{\partial^2}{\partial y_i \partial x_j} & \frac{\partial^2}{\partial y_i \partial y_j} & \frac{\partial^2}{\partial y_i \partial z_j} & L \\ \frac{\partial}{\partial z_i} & \frac{\partial^2}{\partial z_i \partial x_j} & \frac{\partial^2}{\partial z_i \partial y_j} & \frac{\partial^2}{\partial z_i \partial z_j} & L \\ M & M & M & M & O \end{bmatrix} \left(\frac{1}{r_{ij}}\right) \tag{3.20}$$

For describing electronic polarization, which refers to a distortion of the electron density under the influence of an external field, in AMOEBA the classical dipoles are induced at each polarizable atomic site. The molecular polarization is done with an interactive induction model.[249] For further details see ref. 248–250.

A recent improvement by Tafipolsky *et al.*[265] describes a new term for accurate intermolecular potentials with a physically grounded electrostatic expression. Combining the new approach for treating the electrostatic interactions with the AMOEBA force field provides a much more accurate description of the intermolecular interaction energies for polycyclic aromatic hydrocarbons.

Cutoff distance As the non-bonded interaction (electrostatics and vdW) have to be calculated for all possible atom pairs, the amount of interactions nearly scales quadratically. Usually, the 1-2 interaction (i.e. directly bonded atoms) is neglected and the 1-3 and 1-4 interactions are scaled by a constant factor. Nevertheless, the amount of non-bonded atom pairs can be extremely large leading to problems with the necessary computational time and the amount of memory used. Therefore, the concept of cutoff-distances is introduced. All atom pairs which have a distance bigger than a given cutoff (usually 14 Ångström) are neglected during the calculation (i.e. introducing a scaling factor S, Equation 3.21). This modifies the potential energy curves.

$$S_{elec} = S_{vdW} = \begin{cases} 1, & \text{for } R^{AB} \leq R_{cut}, \\ 0, & \text{for } R^{AB} > R_{cut}. \end{cases} \tag{3.21}$$

This very crucial approach leads to a discontinuous potential energy curve of the non-bonded energy. which can be problematic for dynamic simulations and geometry optimizations. Therefore, the cutoff-concept can be improved by introducing a switching function, which depends on the distance R^{AB} (Equation 3.22 and 3.23). There, two distances have to be specified. First, a switch-distance R_{switch} and, second, a cutoff distance R_{cut}, with R_{switch}

being bigger than R_{cut}, e.g. 20 Å and 14 Å. This leads to a smooth decay of the energy curves and, therefore, to a continuous energy function. See the NAMD User's Guide for illustrations and more detail.[266]

$$S_{elec}\left(R^{AB}\right) = \begin{cases} \left(1 - \frac{\left(R^{AB}\right)^2}{\left(R_{switch}\right)^2}\right), & \text{for } R^{AB} \leq R_{switch}, \\ 0, & \text{for } R^{AB} > R_{switch}. \end{cases} \tag{3.22}$$

$$S_{vdW}\left(R^{AB}\right) = \begin{cases} 1, & \text{for } R^{AB} \leq R_{cut}, \\ \frac{\left(\left(R_{switch}\right)^2 - \left(R^{AB}\right)^2\right)^2 \left(\left(R_{switch}\right)^2 + 2\left(R^{AB}\right)^2 - 3\left(R_{cut}\right)^3\right)}{\left(\left(R_{switch}\right)^2 - \left(R_{cut}\right)\right)^3}, & \text{for } R_{cut} < R^{AB} \leq R_{switch}, \\ 0, & \text{for } R^{AB} > R_{switch}. \end{cases} \tag{3.23}$$

3.2.2.6 Cross terms[19,49] All terms described above only depend on one particular property (e.g. bending, stretching, ...). These expressions are common to most force fields. In fact, the different contributions are often coupled. As a simple example, one can think of a water molecule H_2O. The equilibrium angle is about 104.5° and the equilibrium distance of an O-H bond about 0.958 Å. If the angle is compressed, the optimum bond length will increase as well, i.e. the bending motion is coupled to a stretching motion. To account for such effects, an expression coupling the two terms can be included (Equation 3.24).

$$E_{bond/angle} = k^{ABC}\left(\theta^{ABC} - \theta_0^{ABC}\right)\left[\left(R^{AB} - R_0^{AB}\right) - \left(R^{BC} - R_0^{BC}\right)\right] \tag{3.24}$$

Such cross terms can be defined for any desired combination such as $E_{bond/bond}$, $E_{angle/angle}$, $E_{bond/torsion}$ or $E_{angle/torsion/angle}$. These cross terms give a better description of the situations in a molecule, but further computational efforts are the consequence.

3.2.3 MM-Derivatives

The following subsection is based on ref. 19,49,267. When investigating a PES, employed methods often do not only depend on the potential energy, but also on the first derivatives (gradients) or even the second derivatives (Hessian-matrix) of the energy with respect to the Cartesian coordinates. Most local optimization techniques use gradient information to find a minimum, transition state search algorithms utilize the gradient and often the Hessian matrix to locate a saddle point. The simple molecular dynamics algorithm needs the gradient information to define velocities and movements.

The derivative of an energy term can be obtained by differentiating the energy expression with respect to the used variables. All described energy terms are calculated using internal coordinates, whereas the algorithms usually employ Cartesian coordinates. Hence, the obtained formulas for derivatives in internal coordinates have to be converted to Cartesian coordinates. A good overview of how to derive the gradient expression is given in the documentation to the "Consistent Force field".[267] As the second derivatives are much more complex, they are not shown here. The derivation, however, follows the same principles. Equations 3.25 to 3.29 show the equations for the first derivatives in internal coordinates. They are obtained by differentiating the energy expressions given in the sections above by the stated variables.

$$\frac{\partial E_{bond}}{\partial R_{AB}} = 2k^{AB}\left(R_{AB} - R_0\right) \tag{3.25}$$

$$\frac{\partial E_{angle}}{\partial \theta_{ABC}} = 2k^{ABC}\left(\theta_{ABC} - \theta_0\right) \tag{3.26}$$

$$\frac{\partial E_{torsion}}{\partial \omega_{ABCD}} = -\frac{1}{2}V_1 \sin(\omega) + V_2 \sin(2\omega) - \frac{3}{2}V_3 \sin(3\omega) \tag{3.27}$$

$$\frac{\partial E_{vdW-LJ}}{\partial R_{AB}} = \varepsilon\left[-12\left(\frac{R_0}{R_{AB}}\right)^{12}\frac{1}{R_{AB}} + 6\left(\frac{R_0}{R_{AB}}\right)^{6}\frac{1}{R_{AB}}\right] \tag{3.28}$$

$$\frac{\partial E_{el-charge}}{\partial R_{AB}} = \frac{q_a q_b}{\varepsilon\left(R_{AB}\right)^2} \tag{3.29}$$

Possible implementations of these derivatives are not discussed at this point. There are a lot of possibilities of how to program efficiently which are not further discussed here. One possibility is for example the temporary storage of the $\left(\frac{R_0}{R_{AB}}\right)^6$ and $\left(\frac{R_0}{R_{AB}}\right)^{12}$ values.

For using the calculated derivatives for further algorithms, they have to be translated into Cartesian coordinates. This is done by further differentiating the internal coordinates by the Cartesian x, y, and z coordinates.

Equation 3.30 shows the general expression for differentiating for two variables. First, the internal coordinate part is derived (Equation 3.31). x_a belongs to the atom for which the derivatives are calculated, x_b is the atom which stays constant (the partial derivatives have to be performed in each direction (x, y, z) and for each involved atom). Equation 3.33 shows the derivative of r with respect to x. Equation 3.34 finally gives the expression for the first derivative of the stretching energy in Cartesian coordinates with respect to the x-coordinate of the first atom. This derivation has to be calculated for each coordinate and each atom (i.e. six derivatives for the stretching energy). All derivations containing interatomic distances (i.e. vdW and electrostatic terms) can be derived in the same manner.

$$\frac{\partial E_{bond}(r)}{\partial x_a} = \frac{\partial E_{bond}(r)}{\partial r}\frac{\partial r}{\partial x_a} \tag{3.30}$$

$$\frac{\partial E_{bond}(r)}{\partial x_a} = 2k^{AB}\left(R_{AB} - R_0\right)\frac{\partial r}{\partial x_a} \tag{3.31}$$

$$\text{with: } r = R_{AB} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2} \tag{3.32}$$

$$\frac{\partial r}{\partial x_a} = \frac{(x_a - x_b)}{r} \tag{3.33}$$

$$\Rightarrow \frac{\partial E_{bond}(r)}{\partial x_a} = \frac{2k^{AB}\left(R_{AB} - R_0\right)\left(x_a - x_b\right)}{R_{AB}} \tag{3.34}$$

The partial derivatives for angle terms (e.g. bending and torsional energy terms) are derived in a similar way. The bending angles or torsional angles are defined as angles between vectors (bending) or planes (torsions). The angles and relevant sine and cosine function can then be described by their trigonometric definitions using dot and cross products.[268] Thus, the energy expressions can be defined in Cartesian coordinates. The derivatives are then defined by the different combinations of such terms. A detailed description about the implementation can be found in the documentation of the Consistent Force Field;[267] on page 113 ff. for the angle terms and page 119 ff. for the torsional terms.

3.3 Local optimization algorithms

The determination of local minima is quite trivial because it is only necessary to search downhill in all directions. A huge amount of different algorithms is available to solve these tasks efficiently. Simple optimization algorithms, like the Simplex method,[269] only use function values to determine a local minimum. This may be efficient for small optimization problems, but it becomes too slow for functions with many dimensions.[19] Therefore, first derivative or even second derivative information has to be taken into account.
The most commonly used algorithms are Steepest Descent, Conjugate Gradient, and Newton or Quasi-Newton algorithms. One of the most efficient algorithms for solving large optimization problems belongs to the Quasi-Newton algorithms, in particular the Limited memory Broyden-Fletcher-Goldfarb-Shanno algorithm by Nocedal.[270–275] In the following subsection, the different algorithms are described in more detail.

3.3.1 Steepest Descent

The most simple approach to using gradient information within a local optimization is the Steepest Descent (SD) algorithm.[19,269] As the gradient vector **g** always points into the direction of the biggest function increase, the function value can be lowered by following the opposite direction. This means the search direction **d** is given by the negative gradient (Equation 3.35).

$$d = -g \tag{3.35}$$

The minimization is then implemented by:

$$x_{n+1} = x_n - \lambda \cdot g = x_n + \lambda \cdot d \tag{3.36}$$

The step size λ is determined by a line minimization. In principle, the SD algorithm will always approach a minimum if the line minimization is performed accurately enough.[19]
However, the SD method has two big disadvantages. First, the SD method often oscillates around the real minimum energy path because a subsequent step will be perpendicular to the previous step. This comes from the nature of determining the step direction. The energy can be further lowered by following the gradient component along the previous search direction.[19,269] Second, as the step size is determined by taking the decrease of the function values or the gradient values into account, it will be smaller when approaching the minimum. Therefore, especially for long narrow valleys, the convergence slows down dramatically.[19,269]
The main problem of the SD method, the partial canceling of the previous step by perpendicular search directions, is improved by the Conjugate Gradient (CG) method.

3.3.2 Conjugate Gradient

The CG algorithm takes the gradient information of the current and the previous step to construct a search direction which is "conjugate" to the previous search direction.
The search direction is given by Equation 3.37. For the first step, $d_0 = -g_0$. Each subsequent direction is a combination of the gradient at the current position and the previous search direction.

$$d_i = -g_i + \beta_i d_{i-1} \tag{3.37}$$

For the determination of β, several possibilities are available. Some commonly used methods are the *Fletcher-Reeves* (FR), the *Polak-Ribiere* (PR) or the *Hestenes-Stiefel* (HS) approaches (Equation 3.38 to 3.40).

$$\beta_i^{FR} = \frac{g_i^t g_i}{g_{i-1}^t g_{i-1}} \tag{3.38}$$

$$\beta_i^{PR} = \frac{g_i^t (g_i - g_{i-1})}{g_{i-1}^t g_{i-1}} \tag{3.39}$$

$$\beta_i^{HS} = \frac{g_i^t (g_i - g_{i-1})}{d_{i-1}^t (g_i - g_{i-1})} \tag{3.40}$$

For an exact quadratic function, these approaches are the same. However, in real world problems, the functions are not exactly of quadratic behavior and the approaches differ. Furthermore, the CG method often has to be restarted (i.e. setting β to zero) during an optimization procedure. It has been shown, that the PR approach behaves somewhat better than the others and has the tendency to restart more smartly. Therefore, this approach is often preferred in practice.[19,269]

3.3.3 Newton-Raphson method

The Newton-Raphson (NR) method makes use of the second derivatives (Hessian matrix).[7,19,269] It expands the function to be optimized to second order around a given point x_0:

$$f(x) \approx f(x_0) + g^t(x - x_0) + \frac{1}{2}(x - x_0)^t H(x - x_0) \tag{3.41}$$

To find the biggest change in energy, the condition $\frac{df(x)}{dx} = 0$ (i.e. the gradient in the second order expansion 3.41 has to be zero) has to be fulfilled leading to the Newton-Raphson step:

$$(x - x_0) = -H^{-1}g \tag{3.42}$$

When the Hessian matrix is diagonalized and therefore the coordinate system transformed by a unitary transformation, the NR step can be written as:

$$\Delta x' = -\frac{f_i}{\varepsilon_i} \tag{3.43}$$

f_i is the component of the gradient which points into the direction of the ith eigenvector of the Hessian matrix with the eigenvalue ε_i (i.e. the projection of the gradient onto the ith eigenvector).

There are several problems coming with the Newton-Raphson method. A minimum is defined by a gradient with a value of zero and a Hessian matrix with only positive eigenvalues. When the eigenvalues of the Hessian are positive, the NR method will converge to a local minimum. Nevertheless, if one eigenvalue is negative, the step in this direction increases the energy and the NR method can converge to a transition state, i.e. a stationary point with one negative eigenvalue (first order saddle point). In general, the NR algorithm will only converge to the nearest stationary point, regardless of its character (minimum, maximum, saddle point).

A second problem comes with the step size. As the inverse Hessian is used and the step size is constructed using the eigenvalues of the Hessian matrix, the step size can become unreasonably large when an eigenvalue comes close to zero. This can take the search variables outside a reasonable search range. This problem can be overcome by using a maximum step size as upper boundary. To ensure a correct step direction (e.g. only positive eigenvalues of the Hessian when optimizing to a local minimum) a shift parameter λ is introduced:

$$\Delta x' = -\frac{f_i}{\varepsilon_i - \lambda} \tag{3.44}$$

By choosing λ to be below the lowest eigenvalue of the Hessian, the denominator is always positive thus leading to a local minimum.

A further problem of the NR method simply comes from computational aspects. When looking at large scale problems, diagonalization of the Hessian matrix is very time-consuming. Furthermore, the second derivative can be too expansive to calculate.

To summarize, the quadratic convergence of the NR method, which is usually observed close to a stationary point (further away a more linear convergence is often obtained), makes it a very efficient and fast algorithm. By using a shift parameter λ, many of the problems of NR methods can be solved. As long as the computation and diagonalization of the second derivative matrix is not too computationally demanding, it is a highly recommendable method.

For very big optimization problems, where the Hessian is not accessible or the storage can not be managed anymore, Quasi-Newton methods have to be used. A very famous representative is the L-BFGS algorithm[270–275] described in the following section.

3.3.4 Quasi-Newton algorithms: the L-BFGS method

Quasi-Newton methods also use the Hessian matrix for deriving step directions and step lengths. But the Hessian matrix does not have to be calculated explicitly. It will be approximated using gradient calculations of the current and previous steps. Therefore, the Hessian as well as previous steps have to be saved in memory leading to the same storage problems as the NR-methods. A very efficient Quasi-Newton algorithm uses the updating scheme of the Hessian proposed by Broyden-Fletcher-Goldfarb-Shanno (BFGS algorithm (Equation 3.46)[271–275]).

Here, g_k and g_{k+1} are the gradients of the function to be minimized and H is its Hessian with:

$$s_k = x_{k+1} - x_k \quad \text{and} \quad y_k = g_{k+1} - g_k \tag{3.45}$$

the BFGS update formula is given as:

$$\overline{H} = H + \frac{s \cdot s^T}{y^T \cdot s}\left[\frac{y^T \cdot H \cdot y}{y^T \cdot s} + 1\right] - \frac{1}{y^T \cdot s}\left[s \cdot y^T \cdot H + H \cdot y \cdot s^T\right] \tag{3.46}$$

The Limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method by Nocedal[270,271] is the most prevalently used Quasi-Newton algorithm for optimization with many independent variables. The L-BFGS is based on the BFGS method[272–275], but the storage of the Hessian (and the inverse of the Hessian) is optimized for minimum memory requirements.

The implementation of the L-BFGS method is nearly identical to the one of BFGS. They only differ in the matrix updating scheme where recursive calculations are used. In L-BFGS, the correction vectors are stored separately. When the requested storage is depleted, the oldest information is deleted and the newest is saved (first in - first out strategy). The user can specify the variable m which gives the number of saved corrections. While the iteration number k is lower than m, the L-BFGS algorithm is the same as the BFGS method. When $k > m$, the new approximate Hessian H_k is obtained by using the information of m previous steps. Therefore, the user can modify the amount of used memory. Of course, the larger the variable m, the better the convergence, however, bigger values of m are also coupled with a larger amount of used memory.

3.4 Transition state search algorithms

Besides the determination of local minima, transition states or saddle points of first order are also of great importance for the description of the energy landscape and possible reactions or transitions. However, the determination of transition states is usually much harder than finding a local minimum. A saddle point of first order is a local maximum in one direction and a local minimum in all other directions. Therefore, one coordinate has to be maximized while all others are minimized and the search algorithm has to move on a knife's edge to locate the point of interest. In general, it is not possible to employ standard optimization algorithms. Several approaches for locating transition states have been suggested. A reasonable division of these algorithms can be done due to their initial set-up into single-ended and double-ended algorithms. Double-ended approaches are used to locate a transition state between two end points, which are usually local minima. Single-ended methods only require one starting point and search for the corresponding transition state. The starting point can either be a local minimum or a guess of the transition state. Certainly, the convergence of a single-ended method is faster when a reasonable guess of the transition state is already provided .[7] In principle, the simple Newton-Raphson methods could be used for optimization to a stationary point of any order. The initial number of imaginary frequencies defines the order of the saddle point. Therefore, Newton-Raphson only converges to a saddle point of first order when an appropriate starting structure with one imaginary frequency is used. Obtaining such starting structures is already a difficult task which limits the practical use of Newton-Raphson for transition state search. Several different approaches have been suggested to avoid this problem and to improve the transition state search.[7,19] Widely used and very efficient algorithms are the eigenvector-following proposed by Cerjan and Miller[196], the Dimer-method[198] or the Nudged Elastic Band approach.[202] These three algorithms are described in further detail in the following section.

3.4.1 Eigenvector-following

The Eigenvector-following approach was first proposed by Cerjan and Miller.[196] Further improvements and applications were proposed by Wales.[197,219,276–280] It is based on the ideas of the standard Newton-Raphson approach. However, as already discussed, Newton-Raphson will only lead to a transition state with one negative eigenvalue if the starting structure possesses only one negative eigenvalue as well. The method is not appropriate for applications with general starting structures. Cerjan and Miller introduced the required flexibility and robustness by introducing a Lagrange multiplier.[196] The Lagrange multiplier is chosen in a way so that following one particular mode will always lead to an energy increase. First, the potential function is approximated with a Taylor expansion up to the second order:

$$V(X) \approx V(\Delta x) = V_0 + D \cdot \Delta x + \frac{1}{2}\Delta x \cdot K \cdot \Delta x \tag{3.47}$$

with:

$$V_0 = V(a) \tag{3.48}$$

$$D = \frac{\partial V(a)}{\partial a} \tag{3.49}$$

$$K = \frac{\partial^2 V(a)}{\partial a^2} \tag{3.50}$$

Equation 3.47 is searched for extrema under the constraint of the fixed step size $\Delta x \Delta x = \Delta^2$. This constraint is introduced by the Lagrange function $L(\Delta x, \lambda)$:

$$L(\Delta x, \lambda) = V_0 + D \cdot \Delta x + \frac{1}{2}\Delta x \cdot K \cdot \Delta x + \left(\frac{\lambda}{2}\right)\left(\Delta^2 - \Delta x \Delta x\right) \tag{3.51}$$

Solving $\frac{\partial L}{\partial \Delta x}$ gives the step size:

$$\Delta x = (\lambda 1 - K)^{-1} \cdot D \tag{3.52}$$

The value of λ, where the transition state is located, is labeled as λ_0. For $\lambda_0 > 0$, the step size given in Equation 3.52 is used. For $\lambda_0 < 0$, the search does not converge to the correct transition state. Therefore, in this case $\lambda = 0$ is chosen. The step size is scaled to an appropriate value.

Wales modified the Lagrangian function L[197] by introducing Lagrange multipliers λ_i for each vibrational mode and choosing an optimal step size for each direction:

$$L(\Delta x, \lambda) = -V_0 - D \cdot \Delta x - \frac{1}{2}\Delta x \cdot K \cdot \Delta x + \frac{1}{2}\sum_i \lambda_i \left(\Delta x \Delta x - \Delta^2\right) \tag{3.53}$$

The final step size in the eigenvector-following approach of Wales is given by Equation 3.54.

$$\Delta x_i = \pm \frac{2D_i}{|K_i|\left(\sqrt{1 + 4\frac{D_i^2}{K_i^2}}\right)} \tag{3.54}$$

The plus sign is used for maximization, the minus sign for minimization.

3.4.2 Dimer method

Most transition state search algorithms require the second derivative (Hessian matrix) of the potential energy for locating a transition state. The Dimer-method, first proposed by Henkelman,[198] is a single-ended search algorithm which only needs the first derivatives. It employs an approach for estimating the Hessian at a given point, which was already presented by Voter in his hyperdynamic method.[281,282] Furthermore, several improvements to the original approach are described, demonstrating the efficiency of the new approach.[199,200,283] First, the original approach proposed by Henkelman is described. Then, changes suggested by Heyden and Kästner are discussed further. An illustration of the principle of the Dimer method can be found in Figure 3.6.

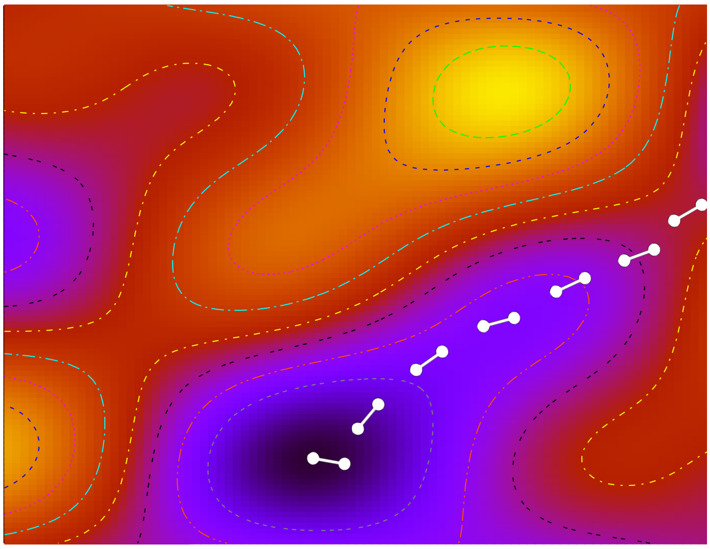


Figure 3.6: Illustration of a transition state search using the Dimer method.

The method is initialized with a first dimer. The dimer is a pair of images slightly displaced from their middle point. The distance and direction are defined by ΔR and $\hat{\tau}$, respectively. Either an initial guess of the transition direction or a random vector can be used. Starting from the middlepoint R_0, the dimer is created following Equation 3.55. When two starting points for the dimer are provided, $\hat{\tau}$ is calculated with Equation 3.56 and 3.57.

$$R_{1/2} = R_0 \pm \Delta R \hat{\tau} \tag{3.55}$$

$$\vec{\tau} = \frac{\left(\vec{R_1} - \vec{R_2}\right)}{2} \tag{3.56}$$

$$\hat{\tau} = \frac{\vec{\tau}}{|\vec{\tau}|} \tag{3.57}$$

At each rotational step, the energy and forces acting on the two dimer endpoints are evaluated. Using a finite difference approach, the curvature at a given point is calculated using

Equation 3.58. During the rotation, E_0 and ΔR are constant. Therefore, the curvature is directly proportional to the dimer energy $E = E_1 + E_2$. The minimum curvature can be found by minimizing the dimer energy. The middlepoint energy can be calculated using the known forces at the endpoints and the dimer energy (Equation 3.59). As it can be seen easily, all required properties are obtained by the forces and energies of the dimer endpoints. The energy and force of the middlepoint does not need to be evaluated.

$$C_\tau = \frac{(F_2 - F_1)\cdot\hat{\tau}}{2\Delta R} = \frac{E - 2E_0}{(\Delta R)^2} \tag{3.58}$$

$$E_0 = \frac{E}{2} + \frac{\Delta R}{4}(F_1 - F_2)\cdot\hat{\tau} \tag{3.59}$$

The performance of the dimer method heavily relies on the algorithm for minimizing the curvature or energy E within the rotation. The first optimization algorithm discussed by Henkelman[198] is a Newton based approach (i.e. steepest descent). The rotational force is given by the forces acting on the dimer endpoints (Equation 3.60).

$$F_R = -(F_2 - F_1) + [(F_2 - F_1)\cdot\hat{\tau}]\,\hat{\tau} \tag{3.60}$$

Besides the dimer direction $\hat{\tau}$, a second vector $\hat{\Theta}$ is defined which is perpendicular to the dimer direction. Within the modified Newton approach, $\hat{\Theta}$ is a unit vector parallel to F_R. These two vectors define the rotational plane. The rotational force can also be represented in scalar form which is used to describe the magnitude of the force (Equation 3.61). Scaling of the scalar force by ΔR makes it independent from the dimer distance.

$$F = \frac{F_R\cdot\hat{\Theta}}{\Delta R} \tag{3.61}$$

Minimizing F gives the optimal rotation. $F_R\cdot\hat{\Theta}$ describes the dot product of the rotational force and the unit vector $\hat{\Theta}$ lying perpendicular to the dimer. When the dot product becomes zero, the two vectors are perpendicular to each other and therefore, the rotational force is pointing in the dimer direction, i.e. the minimal rotational force is obtained. Performing a small finite rotational step $d\theta$, the change in the rotational force can be approximated with a finite difference approach as shown in Equation 3.62. The rotation is performed following Equation 3.63. The second dimer endpoint is obtained with Equation 3.55 and the corresponding forces F_1^*, F_2^*, and $F^* = F_1^* - F_2^*$ are calculated.

$$F' = \frac{dF}{d\theta} \approx \left|\frac{F^*\hat{\Theta}^* - F\hat{\Theta}}{d\theta}\right|_{\theta = d\theta/2} \tag{3.62}$$

$$R_1^* = R + \left(\hat{\tau}\cos d\theta + \hat{\Theta}\sin d\theta\right)\Delta R \tag{3.63}$$

The optimal rotational angle $\Delta\theta$ bringing F to zero is obtained via the Newton's method:

$$\Delta\theta \approx \frac{F\hat{\Theta} + F^{*}\hat{\Theta}^{*}}{-2F'} \tag{3.64}$$

However, the step performed with the Newton method overestimates the rotational angle. A better description of the rotational angle can be obtained using an expression of the dimer energy depending on the angle θ. Expansion with a Taylor approach within the rotational plane finally leads to the expression:

$$\Delta\theta = \theta_0 = -\frac{1}{2}\arctan\left(\frac{2F_0}{F_0'}\right) \tag{3.65}$$

where F_0 and F_0' are given by Equation 3.66 and 3.67 with $\theta = \theta_0$. The constant A depends on the curvatures in both directions of the rotational plane. However, it does not have to be evaluated as it cancels out in Equation 3.65.

$$F = A\sin\left[2\left(\theta - \theta_0\right)\right] \tag{3.66}$$

$$F' = \frac{dF}{d\theta} = 2A\cos\left[2\left(\theta - \theta_0\right)\right] \tag{3.67}$$

After a successful rotation, the dimer is translated towards the saddle point. As already discussed, a saddle point of first order is a maximum along the lowest curvature mode and a minimum along all other coordinates. Minimizing the dimer energy already orients the dimer along the lowest curvature. However, the rotational force tends to pull down the dimer into a minimum. Therefore, the translational force is modified so that the dimer is pushed towards the saddle point. For convex regions (all modes have a positive curvature), whereas a different force is used for non-convex regions (Equation 3.68).

$$F^{\dagger} = \begin{cases} -F^{\parallel} & \text{if} \quad C > 0 \\ F_0 - 2F^{\parallel} & \text{if} \quad C < 0 \end{cases} \tag{3.68}$$

$$F^{\parallel} = \left(F_0 \cdot \hat{\tau}\right)\hat{\tau} \tag{3.69}$$

The rotational and translational steps are iterated until convergences criteria are fulfilled, such as maximum number of steps or a small enough gradient norm.

Kästner proposed several improvements to the Dimer approach.[200] First, the dimer is build up between the middlepoint R_0 and R_1. The second dimer endpoint, R_2, is skipped. Thus, energy and gradient calculations can be saved. Kästner further found that several subsequent rotation steps, to minimize the curvature before a translational step, are superior to a one by one iteration. Therefore, the dimer is rotated until the rotational angle is smaller than a certain threshold. In the first step, a rotational angle Φ_1 is estimated (Equation 3.70). After rotation about the estimated angle, the energy and gradients (F'), and the optimal rotational angle are calculated (Equation 3.72). An illustration of such a rotational step is given Figure 3.7.

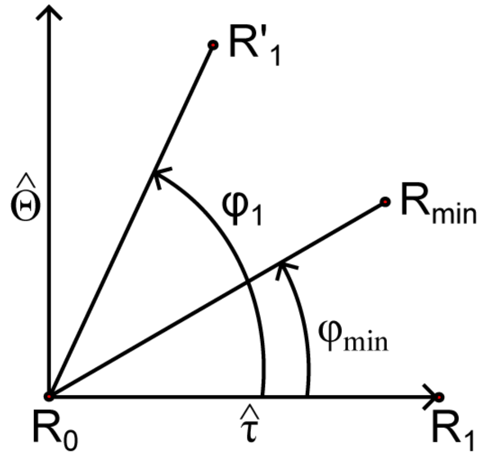


Figure 3.7: Illustration of a rotational step using one trial rotation to calculate the optimal rotational angle.

$$\Phi_1 = -\frac{1}{2}\arctan\frac{\frac{\partial C_\tau}{\partial \Phi}}{2|C_\tau|} \tag{3.70}$$

$$\frac{\partial C_\tau}{\partial \Phi} = \frac{2\left(F_1 - F_0\right)\cdot\Theta}{\Delta} \tag{3.71}$$

$$\Phi_{min} = \frac{1}{2}\arctan\frac{b_1}{a_1} \tag{3.72}$$

$$b_1 = \frac{1}{2}\frac{\partial C_\tau}{\partial \Phi}\Big|_{\Phi=0} \tag{3.73}$$

$$a_1 = \frac{C_\tau\Big|_{\Phi=0} - C_\tau\Big|_{\Phi=\Phi_1} + b_1 sin\left(2\Phi_1\right)}{1 - cos\left(2\Phi_1\right)} \tag{3.74}$$

The gradient at the new dimer endpoint R_{min} can either be calculated or estimated to save one further gradient calculation:

$$\begin{aligned} F_{min} =& \frac{\sin\left(\Phi_1 - \Phi_{min}\right)}{\sin\left(\Phi_1\right)}F_1 + \frac{\sin\left(\Phi_{min}\right)}{\sin\left(\Phi_1\right)}F_1' \\ &+ \left(1 - \cos\left(\Phi_{min}\right) - \sin\left(\Phi_{min}\right)\tan\left(\frac{\Phi_1}{2}\right)\right)F_0 \end{aligned} \tag{3.75}$$

Finally, the use of a more sophisticated optimization algorithm like the L-BFGS method dramatically improved the convergence of the Dimer-method.[200]

3.4.3 Nudged Elastic Band method

In contrast to single-ended search algorithms which are mainly used to locate a transition state exactly, double-ended approaches can be used to determine the minimum energy path (MEP). The MEP represents the lowest energy path connecting two states: an initial and a final state. The maximum of the MEP can be seen as the transition state and is (at least very close to) the saddle point of the path. The Nudged Elastic Band (NEB) method was proposed by Jònsson.[202] It belongs to the chains-of-states method as several points (images) along the path are used to describe the transition. Methods which are very close to the NEB are, for example, the string method[284] or the growing string method.[201] The main difference is the way in which the images are kept equally distant. Within the NEB approach, a spring force is applied to ensure the equal distances. The initial pathway is typically described by a linear fit between the starting point and the end point (from now on called the band), although sometimes different choices might be better. To relax the NEB images to the MEP, a force projection is used. The force projection consists of potential forces acting perpendicularly to the band and spring forces acting along the path. A schematic view of both an NEB and MEP path can be found in Figure 3.8

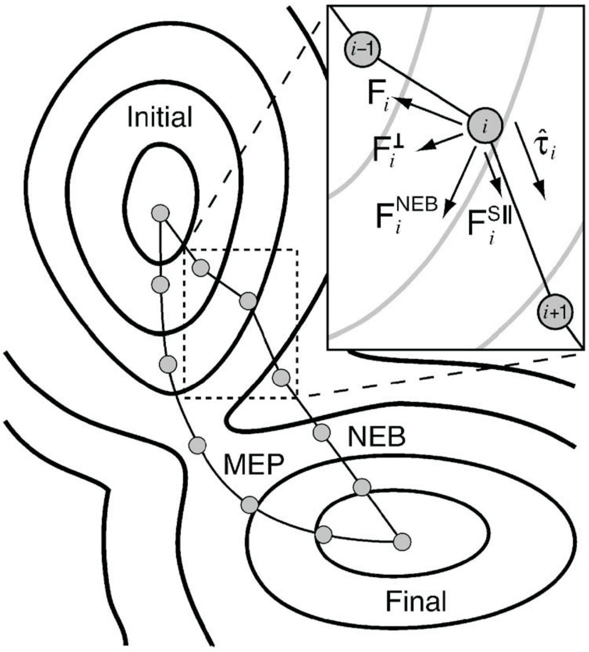


Figure 3.8: Schematic view of an NEB path including the participating forces. Picture taken from the website of the Henkelman research group with permission from Graeme Henkelman.[285]

In the original NEB method[202,286], the complete NEB forces are given by the sum of two independent components:

$$F_i^{NEB} = F_i^{\perp} + F_i^{S\|} \tag{3.76}$$

$F_i^{\perp}$ is the force perpendicular to the band given by:

$$F_i^{\perp} = -\nabla(R_i) + \nabla(R_i) \cdot \hat{\tau}_i \hat{\tau}_i \tag{3.77}$$

and $F_i^{S\|}$ is the spring force ensuring equal spacing of the images:

$$F_i^{S\|} = k\left(|R_{i+1} - R_i| - |R_i - R_{i-1}|\right) \hat{\tau}_i \tag{3.78}$$

$\hat{\tau}_i$ is the tangent at one image pointing in the direction of the product. In the original NEB method the tangent was estimated from the two nearest images R_{i-1} and R_{i+1}. The most simple way is the normalized line segment between the two images:

$$\hat{\tau}_i = \frac{R_{i+1} - R_{i-1}}{|R_{i+1} - R_{i-1}|} \tag{3.79}$$

A slightly better solution is the normalization of the vector τ_i ($\hat{\tau}_i = \tau_i / |\tau_i|$) produced by cutting the two unit vectors in half:

$$\tau_i = \frac{R_i - R_{i-1}}{|R_i - R_{i-1}|} + \frac{R_{i+1} - R_i}{|R_{i+1} - R_i|} \tag{3.80}$$

Further analysis of the tangents of the images revealed that sometimes kinks can occur. Therefore, in a new implementation of the NEB an improved approach is proposed for estimation of the tangent of the NEB-images[203]. Instead of using both adjacent images, only the one with higher energy (V) is used.

$$\tau_i = \begin{cases} \tau_i^{+} \text{ if } V_{i+1} > V_i > V_{i-1} \\ \tau_i^{-} \text{ if } V_{i+1} < V_i < V_{i-1} \end{cases} \tag{3.81}$$

with

$$\tau_i^{+} = R_{i+1} - R_i \quad \text{,and} \quad \tau_i^{-} = R_i - R_{i-1} \tag{3.82}$$

If both images are either lower or higher in energy, the tangent is taken as a weighted average of the two vectors.

$$\tau_i = \begin{cases} \tau_i^{+} \Delta V_i^{max} + \tau_i^{-} \Delta V_i^{min} \text{ if } V_{i+1} > V_{i-1} \\ \tau_i^{+} \Delta V_i^{min} + \tau_i^{-} \Delta V_i^{max} \text{ if } V_{i+1} < V_{i-1} \end{cases} \tag{3.83}$$

with

$$\Delta V_i^{max} = \max\left(|V_{i+1} - V_i|, |V_{i-1} - V_i|\right) \tag{3.84}$$

and

$$\Delta V_i^{min} = \min\left(|V_{i+1} - V_i|, |V_{i-1} - V_i|\right) \tag{3.85}$$

Finally, the tangent is normalized as shown above.

When investigating MEPs, the saddle point is of particular interest. In the normal NEB, the images tend to slip down to the starting points. This leads to the smallest density of images near the saddle point. To avoid this problem, the Climbing Image (CI) NEB was developed.[204]

In this method, the image highest in energy (image *l*) is taken as the best estimate of the saddle point. It does not feel a spring force during optimization and climbs to the saddle point by a reflection of the force along the tangent:

$$F_l^{CI} = F_l - 2F_l \cdot \hat{\tau}_i \hat{\tau}_i \tag{3.86}$$

An overview of the NEB method with a comparison of methods for finding minimum energy paths was given by Sheppard in 2008.[287]

Trygubenko and Wales[205] proposed a Doubly Nudged Elastic Band (DNEB) which should perform better for MEPs with very large forces. It adds a further spring component perpendicular to the path.

The component of the spring force is calculated with

$$F_i^S = k\left[(R_{i+1} - R_i) - (R_i - R_{i-1})\right] \tag{3.87}$$

and the component perpendicular to the tangent is taken as

$$F_i^{S\perp} = F_i^S - F_i^S \cdot \hat{\tau}_i \hat{\tau}_i \tag{3.88}$$

The DNEB force is then the component of the $F_i^{S\perp}$ orthogonal to $F_i^{\perp}$

$$F_i^{DNEB} = F_i^{S\perp} - F_i^{S\perp} \cdot F_i^{\perp} F_i^{\perp} \tag{3.89}$$

The addition of this force to the complete NEB force (for all images but the climbing image) is the DNEB method:

$$F_i^{NEB} = F_i^{\perp} + F_i^{S\parallel} + F_i^{DNEB} \tag{3.90}$$

Carr *et al* successfully applied the DNEB approach together with algorithms to find connected pathways for reactions with many intervening transition states.[124,207]

3.5 Tabu-Search optimization

An overview of Tabu-Search algorithms was given in the PhD thesis of Svetlana Stepanenko.[288] The following section is based on this thesis, recent publications of our group,[52,153–155] and the text book "Stochastic Global Optimization".[8]

The TS method is a metaheuristic algorithm which was first proposed by Glover.[43,44,289]

TS belongs to the local search techniques with the possibility to escape the trap of a local optimum. It can be implemented as a deterministic procedure. When the TS is supported by probabilistic diversification parts it can be further modified to a stochastic algorithm.[8]
The algorithm usually employs a *steepest descent - modest ascent* strategy. The closest local optimum is located following the direction with the biggest function decrease, while a local minimum is left by following the modest ascent direction of the closest neighborhood. Usually, only a subset of the neighborhood is searched for the modest ascent, as a complete exploration would slow down the method.

The main feature of the TS algorithm is the adaptive memory design which allows for a responsive exploration.[8] This way, the TS algorithm can recognize solutions that have been visited before. A new solution is only allowed if it is not included in the so-called Tabu-List (TL) which summarizes recently visited solutions. Therefore, all new solutions have to be compared to the previous ones. Once a new solution is accepted, it is also added to the TL. In the most simple approach, TS combines the local search method with anti-cycling memory-based rules (i.e. the TL). This prevents the search from getting trapped in a local minimum. After a local minimum is located, the next neighbor can be located by an uphill move. As the downhill move to the previous minimum is already set tabu, the search has to continue with uphill moves until the minimum is left. Thus, an important parameter is the size of the TL. It determines the number of moves which are set tabu and therefore determines for how long a solution is not allowed to be visited again. Usually, the TL is organized with the First In First Out strategy. An increasing size of the TL on the one hand reduces the probability of getting trapped in a minimum but on the other hand limits the search. Therefore, the optimal size has to be determined depending on the problem under investigation. Besides the TL further concepts like Tabu-Directions and Tabu-Regions can also be employed.[153]

To avoid a too strict prohibition of solutions due to the TL, further aspiration criteria can be implemented. The most simple criteria is the acceptance of solutions with a better function value than the current best solution independently from the TL.

Other important aspects are the intensification and diversification. Intensification is meant to intensify the search within a certain region. In general, it can be carried out in two ways. On the one hand, the search can be focused on promising attributes of the solution space. On the other hand, attractive regions can be revisited for further investigation.

The diversification search is used in the completely opposite direction. It is used to guide the search towards new unexplored regions. The most simple approach is the restart of the search from a new randomly obtained solution. More sophisticated approaches generate a new solution with information from previous solutions from a long-term memory.

Several different implementations of TS algorithms are developed and some of them are described in the following. Battiti and Tecchiolli described the reactive TS,[290,291] where the TL is dynamically adapted. A fast operating mechanism is used to increase the size of the TL when cycling occurs, and a slow mechanism is used to decrease the size when insufficient

moves are allowed. The continuous reactive TS is a generalization of the reactive TS.
The directed TS described by Hedar and Fukushima[292] employs three different search strategies: exploration, diversification and intensification. It can be seen as a multi-start method. For local search, the Nelder-Mead (also known as downhill simplex method) or the Adaptive Pattern Search are used for generating trial solutions for the exploration procedure. The exploration and diversification search are repeated before the intensification starting from the best solutions is begun. Furthermore, the concepts of tabu-regions, semi-tabu regions, and multi-ranked TL were introduced.
Siarry and Berthiau described the continuous TS[293] for optimizing functions of continuous variables. The enhanced continuous TS[294] described later is an improvement of the continuous TS by introducing diversification and intensification concepts into the approach of Siarry *et al.*[293] The enhanced continuous TS follows the basic approach of Glover as close as possible. After a diversification step to localize the most promising regions, the intensification is used to identify the best solution within this area.
In the Engels group, the new TS algorithms Gradient Tabu Search (GTS),[153] Gradient Only Tabu Search (GOTS), and Tabu Search with Powell's Algorithm (TSPA)[154] were described. GTS employs gradients for a fast localization of the closest minimum, while analytical diagonal elements are used in the modest ascent part.[153] In the GOTS approach, the diagonal elements of the Hessian are replaced by a grid of function values to avoid the calculation of the second derivatives. The TSPA further neglects the gradients for local optimization by the implementation of Powell's algorithm which only employs function values.[154,269] In comparison, the GOTS approach seems to be the most promising approach. Therefore, it builds the basis of the present thesis.
Currently, the field of applications of TS algorithms is gaining importance. An overview is given in ref. 288 and 8. Very recently, Shen *et al.* applied a modified TS to the variable selection for developing an analysis system in QSAR studies.[295] Here, a mechanism to share information about the best position of all iterations and the personal position is introduced into the generation of new neighbor solutions. Furthermore, TS algorithms are applied to protein structure predictions as described by Zhang *et al.*[296] and Dotu *et al.*[297] Rusu and Wriggers presented the VolTrac method, a combination of a genetic algorithm and a bidirectional expansion with a TS approach. The new algorithm is used to trace alpha helical structure elements in cryo-electron microscopy.[298,299]
The examples listed above, as well as the comparison of GOTS to other well-known global optimization algorithms, outline the efficiency of TS approaches. The most important aspects for efficient TS algorithms is a proper choice of starting structures, an efficient diversification procedure, a powerful modest ascent part and a proper implementation of Tabu restrictions rules. Among other things, the present work investigates how the starting structure, the diversification search, and the modest ascent part can be improved in comparison to the original GOTS approach.

4 Method development

A main topic of this work was the development and improvement of Tabu-Search based global optimization algorithms, as well as the development of new approaches for their application. The new developments and implementations are described in the following chapter. It starts with a description of the general design of the Conformational Analysis and Search Tool (CAST) in subsection 4.1. Most of the algorithms described in this work were implemented into the CAST program. The next subsection describes the new Tabu-Search based algorithms in more detail (subsection 4.2) and subsection 4.3 presents a new approach for the solvation of molecules. This is followed by the description of a new algorithm for determining reaction pathways (subsection 4.4). The section concludes with the presentation of graphical user interface (GUI) developments (subsection 4.5).

4.1 Development of the Conformational Analysis and Search Tool - CAST

4.1.1 General design of the software

The Conformational Analysis and Search Tool is written in C++ in an object-oriented way to offer a flexible and modular work environment. An overview of important functionalities is given in Figure 4.1. Of course the flowchart only contains an overview and not all the capabilities of the program.

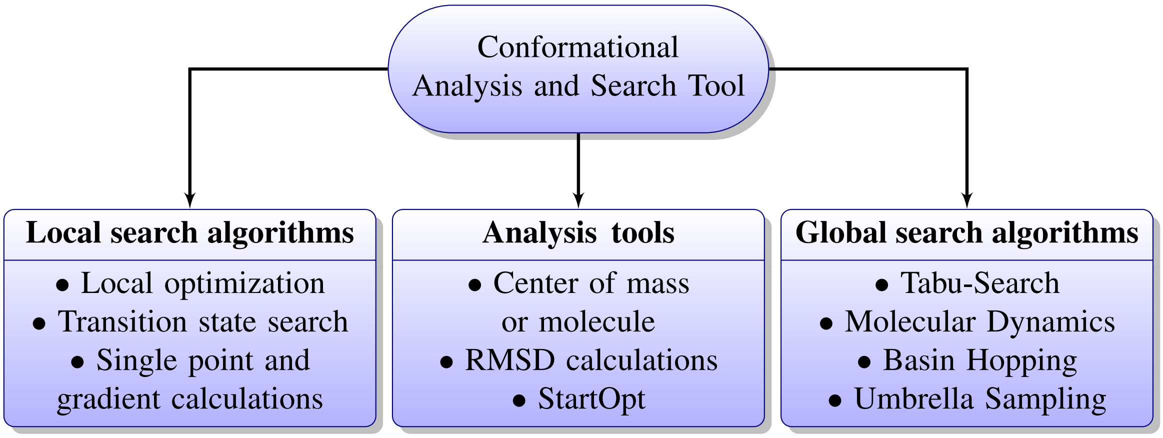


Figure 4.1: General design of the Conformational Analysis and Search Tool (CAST).

To allow for the best possible portability, the implemented algorithms are encapsulated into classes. Thus, the different approaches are easily interchangeable between different algorithms. The main class (Energy_all) contains all information on the coordinates and employed force fields. In the beginning, this class is initialized with all necessary information (coordinates, force field parameters, ...). Afterwards, the concerning classes for force field calculations are initialized. The main features can be divided into local search algorithms, global search algorithms, and analysis tools. Currently, CAST contains several local optimization algorithms like Steepest Descent, Conjugate Gradient, and the L-BFGS algorithm. Furthermore, the Nudged Elastic Band (NEB) and the Dimer method are implemented for transition state search. Of course, simple energy and gradient calculations are also possible. The global optimization algorithms are mainly based on the Tabu-Search approach proposed by Svetlana Stepanenko[153–155] and further improved and modified within this work. In addition, other algorithms like Molecular Dynamics, Umbrella Sampling, and Basin Hopping are implemented. Finally, several tools are implemented which allow for data analysis like the StartOpt algorithm,[46–48] RMSD calculations, calculation of center of mass/molecule, or the determination of stereogenic centers. Most force field calculations can be performed with the internal force field implementations. Currently, the OPLS-AA, Charmm22, Amber99, and Amoeba force field are supported. The main class also contains interfaces to other programs, which were implemented to enlarge the functionality of the program. The interfaces include MOPAC, DFTB+, TeraChem, Tinker, and OpenBabel and are described below in more detail. In the following, different aspects of the CAST program with their implementations are extensively described.

4.1.2 Force field implementations

The original Gradient Only Tabu Search (GOTS) algorithm developed by Svetlana Stepanenko employed the conformational search calling the ChemShell program via different perl scripts. Thus, the necessary information is always written onto the hard drive disc. As force field calculations are usually very fast, most time is spent on writing and reading data. Furthermore, the communication with explicitly called perl scripts is rather slow.

The program package Tinker can be compiled as a library and linked to the main program. Thus, external perl scripts are not necessary. Tinker provides a huge number of force fields and algorithms. Therefore, the first step in optimizing the Tabu-Search was the implementation of Tinker. The modified version of Tabu-Search which uses Tinker for energy calculations instead of ChemShell already increased the speed up to 30 times. A detailed comparison of the different Tabu-Search versions is given in the Appendix in Table A.4. The original ChemShell version is not included as the calculation took too long. The Tabu-Search using Tinker is already reasonably faster than the original version. However, communication is still mainly done via hard disk drives. Therefore, the amount of CPU time spent for input/output (I/O) is measured using the procedure described below (see page 53). The tests revealed that I/O consumes a lot of CPU time for this version of Tabu-Search. Therefore, most of the I/O was removed by implementing the communication between Tinker and Tabu-Search using internal variables. This speeds up the program, especially when Basin Hopping is used within the diversification of Tabu-Search (for more detail about the algorithm see section 4.2).

Tinker is a very modular program and a lot of functionalities are not needed for the Tabu-

Search purposes. Therefore, the development of a new force field implementation was initiated. The final OPLS-AA based force field (FORCE, see benchmark in appendix A.4) has a much better performance than the version using Tinker for energy calculation. The force field employed the equations given in subsection 3.2. For simplification the source code is not shown here. At this point, Basin Hopping calculations are still done by calling the Tinker program. Therefore, the Tabu-Search without Basin Hopping diversification had a faster speedup, while the Tabu-Search-BH algorithm is still much slower.

The good results in the acceleration of Tabu-Search resulted in the development of a new molecular mechanics program called CAST (Conformational Analysis and Search Tool). The Tabu-Search program was reimplemented and optimized within the new program. Likewise, the Basin Hopping approach was implemented into CAST. Furthermore, the force field was also optimized during the F-Praktikum of Daniel Weber.[300] The results of the benchmark can be found in Table A.4. The great speedup of the new program can be seen very easily (up to more than 100 times as fast). However, the computational times needed for rather big systems (1UBQ in the benchmark) are still extremely long. This results from the inefficient scaling of the original *modest ascent* strategy of GOTS.

Therefore, the *modest ascent* strategy of Tabu-Search was improved by the adaptation of the Dimer-method.[198–200] For more detail see section 4.2. For smaller systems the Tabu-Search algorithms need similar resources because the original neighborhood search still converges reasonably fast. However, for bigger systems like 1UBQ, the Tabu-Search-Dimer approach is much faster (speedup by a factor of up to 40 within the benchmark).

Measurement of CPU time spent for I/O The described procedure gives a rough estimate of the amount of I/O the program produces. It is not an exact value as the program cannot work without any I/O. However, if I/O is a bottle neck in the program it can be found through this procedure.

1. You have to check whether the programs *iostat* and *time* are installed on your system.
2. Choose a hard disk drive (HDD) which is not used by any other process. NOTE: It really has to be a single HDD, a partition is not enough.
3. The program under investigation has to be run on this HDD. And remember: NO other process may write or read to or from this HDD.
4. Call *iostat* and write down the output for your HDD (e.g. sda, sdb...)
5. Call your program using *time* to get the CPU-time the program requires and write it down.
6. After the program has finished, call *iostat* again and write down the new values.

7. *iostat* gives you the number of blocks which were written and read from this HDD since mounting. Therefore, the difference between the values before and after executing the program is taken.

8. Find out the block size used on your HDD. E.g. for ext3 one can use dumpe2fs to obtain the block size. Normally, a block size of 4096 byte is typical (but it has to be checked anyway).

9. The small program called *seeker* written in C[301] gives you the number of blocks written to your HDD per second as well as the time needed to read one random block.

10. The blocks per second have to be converted into kB/sec (taking into account the block size from step 8).

11. Convert the written blocks (from step 7) to written kB and calculate (with the writing rate in kB/sec) the time needed to write this amount of block in seconds. THIS IS YOUR CPU TIME NEEDED TO PRODUCE THE OUTPUT.

12. *seeker* gives you a value for reading speed in ms/block. Therefore, you can calculate the time needed to read this amount of blocks. THIS IS YOUR CPU TIME NEEDED TO RECEIVE INPUT.

13. Subtract the OUTPUT and INPUT time from the overall CPU time (from step 5). THIS IS THE CPU TIME THE PROGRAM NEEDS (THEORETICALLY) WITHOUT IO. Divide it by the total CPU time. Now, you have a scaling factor for calculating the CPU time which would be needed if no IO is produced.

4.1.3 Interfaces

Currently, CAST contains three interfaces for semi-empiric and *ab initio* programs as well as two interfaces for force field libraries. MOPAC[302] and DFTB+[303] are interfaced for semi-empiric calculations, while TeraChem[304] is used as an *ab initio* program. Further force fields are provided by Tinker[37,250,305–308] and OpenBabel[309,310]. The interfaces to MOPAC and DFTB+ are implemented via system calls to the concerning executables. The necessary information is exchanged via files on the hard disk drive. Usually, when semi-empiric calculations are used most time is spent within the semi-empirical program. Therefore, the interface based on system calls is sufficient. The situation is different for TeraChem, as a lot of time is spent on initializing the hardware (i.e. GPUs) and therefore it is preferable to initialize the program only once. More details are given in subsection 4.1.4. The main task of the interfaces is to manage the correct communication between the two programs. CAST needs either energy values, energy and gradient information, or local optimizations. Therefore, CAST has to provide correct input files for the different programs and, after the desired task is done, it has to extract the obtained data from the output files. These subroutines are implemented in the class *Energy_all* (see Listing 4.1). For each calculation (energy, energy&gradient, local optimization), a public subroutine is implemented which can be called throughout the CAST program. The complete data conversion and extraction is done by private subroutines. First, the information of coordinates used in CAST has to be converted into the proper file formats. MOPAC uses standard Cartesian coordinates with several lines for MOPAC specific input information like the employed method. DFTB+ supports its own file format, the so-called *gen* format. The subroutines *extractenergy* and *getgradients* evaluates the output files of MOPAC and DFTB+, respectively, and extracts the energy values and gradients. Finally, the information is given back to the calling subroutine within CAST. Thus, changes to the interface can easily be made in the main class. Furthermore, new programs can be interfaced quite easily.

In contrast to MOPAC, DFTB+, and TeraChem, Tinker and OpenBabel can be compiled as libraries. Tinker is written in Fortran while OpenBabel is written in object-oriented C++. Therefore, all required subroutines can be directly called from the program by linking CAST against the desired libraries. The main program only has to provide subroutines for the communication between the program and the library. In the case of Tinker, these subroutines have to be declared by *extern "C"* as Fortran subroutines have to be called by a C++ program. OpenBabel is also written in C++ which makes the implementation more easy and straightforward. The source code of the interfaces is not shown here, but all subroutines calling Tinker or OpenBabel are placed within the *Energy_all* class. A documentation about the development with OpenBabel is given in the API of OpenBabel 2.3.0.[311]

Listing 4.1: Subroutines for implementing the interfaces to MOPAC and DFTB+. The subroutines are implemented in the class Energy_all.

```cpp
class Energy_all {
 private:
   ...
   //! DFTB+
   void xyz2gen();
   void gen2xyz();
   double extractenergyDFTB();
   void dftbgetgradients();
   //! MOPAC
   void tinkertomopacopt();
   void tinkertomopacsingle();
   void tinkertomopacgrad();
   void mopactotinker();
   double extractenergyMOPAC();
   void mopacgetgrad();
 public:
    ...
    //! subroutines for dftb+
    double dftbopt();
    double dftbgrad();
    double dftbenergy();
    //! subroutines for MOPAC
    double mopacopt();
    double mopacgrad();
    double mopacenergy();
};
```

4.1.4 Interfacing TeraChem via MPI

Usually, *ab initio* calculations are by far too expensive for global optimizations. However, recent advances in accelerating *ab initio* calculations using graphical processing units (GPU) have reached a point where a global optimization is affordable. To allow for an efficient communication between the GPU-accelerated *ab initio* program TeraChem[304] and the CAST program, an MPI (Message Passing Interface) based interface between these two programs was implemented. The development was done in cooperation with the group of Todd Martinez at the university of Stanford. A flowchart of the interface can be found in Figure 4.2. The communication process is based on a server-client connection. CAST serves as the server which manages the communication ports. TeraChem deals as a client and waits for assignments from CAST. CAST as well as TeraChem are started with MPI support. After the MPI_Communicator and MPI_Status variables are declared in the *main* function of CAST, the connection between CAST and TeraChem is established. From now on, TeraChem waits for a tag sent by CAST. Four different tags were implemented. *tag=0* will stop the connection and clear all data properly. This tag is sent at the end of the program. *tag=1* calls for an energy calculation of TeraChem while *tag=2* calls for energy and gradients. In principle, CAST could perform all other calculations including local optimizations. However, it has been found that the performance is much better when the local optimization is done within TeraChem. Therefore, *tag=3* asks for a local optimization within TeraChem. The better performance mainly comes from a faster convergence of the self consistent field (SCF) calculations as molecular orbital information like coefficients can be stored and reused.

4.1.5 Analysis tools

The huge amount of information obtained by conformational search studies and the partially very complex systems make it necessary to employ a set of analysis tools. CAST provides classes for the calculation of root mean square deviations, the calculation of different centers of molecules as well as the determination of stereogenic centers. The algorithms are described in the following.

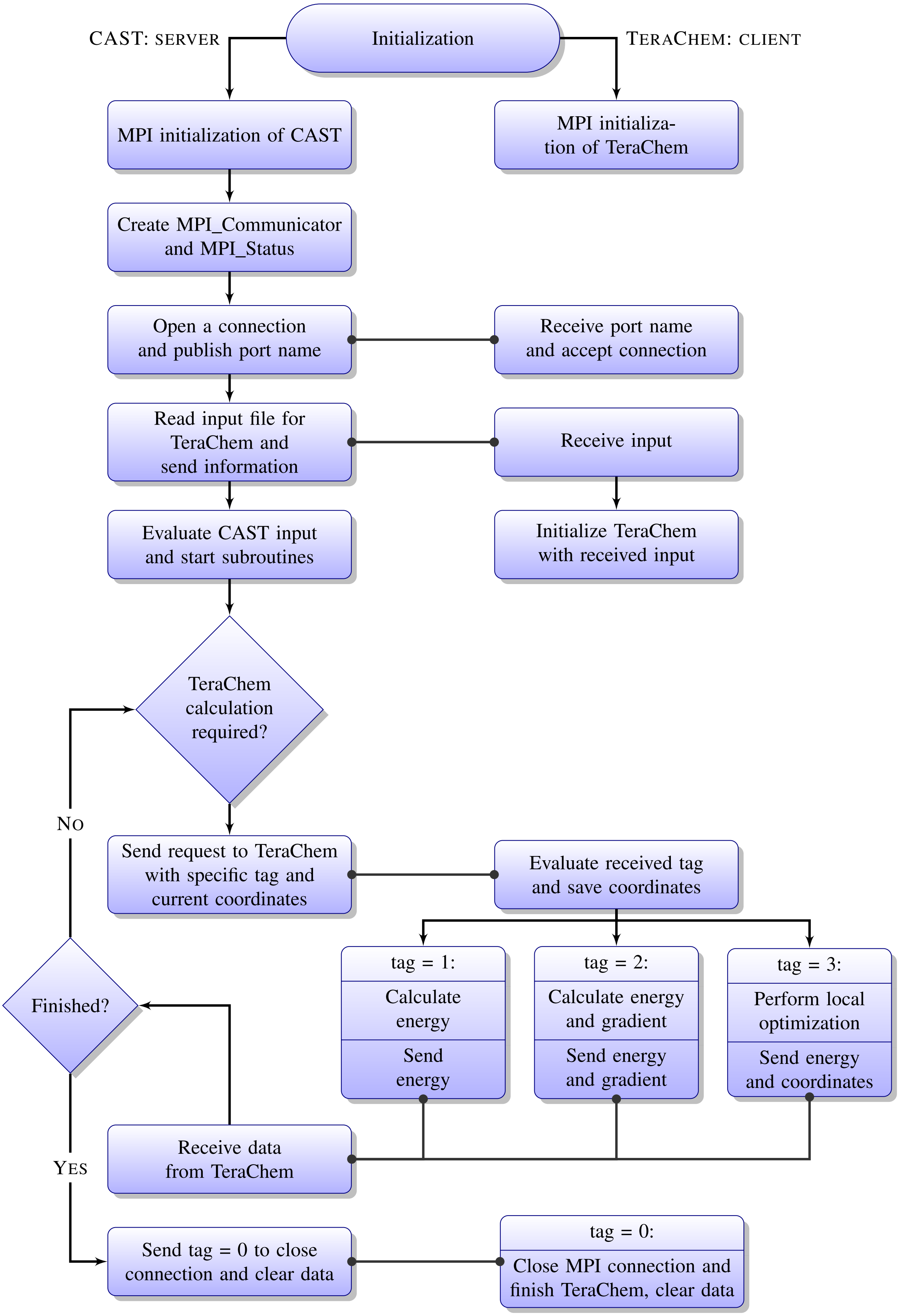

Initialization
CAST: SERVER
TERACHEM: CLIENT
MPI initialization of CAST
MPI initialization of TeraChem
Create MPI_Communicator and MPI_Status
Open a connection and publish port name
Receive port name and accept connection
Read input file for TeraChem and send information
Receive input
Evaluate CAST input and start subroutines
Initialize TeraChem with received input
TeraChem calculation required?
NO
Send request to TeraChem with specific tag and current coordinates
Evaluate received tag and save coordinates
tag = 1:
Calculate energy
Send energy
tag = 2:
Calculate energy and gradient
Send energy and gradient
tag = 3:
Perform local optimization
Send energy and coordinates
Finished?
Receive data from TeraChem
YES
Send tag = 0 to close connection and clear data
tag = 0:
Close MPI connection and finish TeraChem, clear data

Figure 4.2: MPI-based interface between TeraChem and CAST.

Root Mean Square Deviations The root mean square deviations (RMSD) are used to calculate the deviation of a test point to a reference point. They can be used to compare single atoms, groups or the complete system. By calculating the RMSD for each point of a trajectory, the dynamical deviation of a system can be investigated. The larger the RMSD, the bigger the deviation from the reference point. The most simple case is the RMSD calculation for a molecule in Cartesian space. There, the squared difference in the Cartesian coordinates for each atom is calculated. This value is normalized by the number of atoms and the root is taken (see Equation 4.1).

$$d_{ij} = \left[\frac{1}{N}\sum_{k=1}^{N}\left(r_k^i - r_k^j\right)^2\right]^{1/2} \tag{4.1}$$

Comparing the RMSD of conformations, the Cartesian RMSD value is not the best choice. Two structures which are conformationally very close but have different geometric centers will display a very high RMSD. Of course, these two structures could be aligned before calculating the RMSD. This is indeed useful for data analysis. However, within a conformational search, it is not desirable to align each structure. Therefore, it is preferable to calculate the RMSD in internal dihedral coordinates. The principle is the same. Solely the periodic nature of dihedral angles has to be taken into account. This is implemented by Equation 4.2.

$$d_{ij} = \left\{\frac{1}{N}\sum_{k=1}^{N}\min\left[\left(\theta_k^i - \theta_k^j\right)^2, \left(2\pi - \theta_k^i - \theta_k^j\right)^2\right]\right\}^{1/2} \tag{4.2}$$

The different RMSD variants are implemented in the class *RMSD* (see Listing 4.2).

Listing 4.2: Class definition for calculating RMSD values.

```
class RMSD {
  private:
    ...
  public:
    ...
    //! RMSD in internal dihedral coordinates
    double rmsddihedral(double*, double*,int&);
    double rmsddihedral(Vec_DP&, Vec_DP&,int&);
    double rmsddihedral(std::vector<double>&, std::vector<double>&);
    //! RMSD in Cartesian coordinates
    double rmsdcart(double*, double*,int&);
    double rmsdcart(Vec_DP&, Vec_DP&,int&);
    double rmsdcart(Vec_DP&, std::vector<double>&,int&);
    double rmsdcart(std::vector<double>&, std::vector<double>&);
};
```

Center of Mass/Molecule It is often important to determine the center of a given system. In principle, two different centers can be calculated, the geometric center and the center of mass. The geometric center is simply the Cartesian middle point of the molecule (Equation 4.3).

$$< r_{\text{geometric}} >= \frac{\sum_{i=1}^{N_{\text{atoms}}} r_i}{N_{\text{atoms}}} \tag{4.3}$$

Often, the center of mass is also important. It can be calculated by mass weighted coordinates as given in Equation 4.4.

$$< r_{\text{mass}} >= \frac{\sum_{i=1}^{N_{\text{atoms}}} (m_i \cdot r_i)}{\sum_{i=1}^{N_{\text{atoms}}} m_i} \tag{4.4}$$

Both possibilities are implemented in the class *Center* (see Listing 4.3). The subroutines for calculating the concerning centers use a bool variable as an argument. *True* translates the molecule to the calculated center while *false* does not change the system at all.

Listing 4.3: Class definition for center of mass/molecule.

```
class Center{
 private:
   Coord::coordinates* coords;
   struct coord{
     double X, Y, Z;
     double Rmin, Rmax;
   };
   int natoms;
   int natomsforcenter;
   double totalmass;
   std::vector<double> distvect;
 public:
   Center(Coord::coordinates* coords, int);
   void Centerofmass(bool);
   void Centerofmolecule(bool);
   void distance(coord*);
   void translatetocenter(coord*);
   void writetinker();
   coord CoM;
   coord CoMol;
};
```

Stereocenter identification Many organic molecules contain stereogenic centers, e.g. a tetrahedral coordinated atom with four different ligands. These centers cannot be interchanged by simple conformational changes. However, many algorithms are able to invert a stereogenic center, especially the Basin Hopping approach implemented as diversification part within the Tabu-Search (subsection 4.2). Therefore, it is necessary to locate such centers and fix them during a conformational search. An illustration of a stereogenic center is given in Figure 4.3.

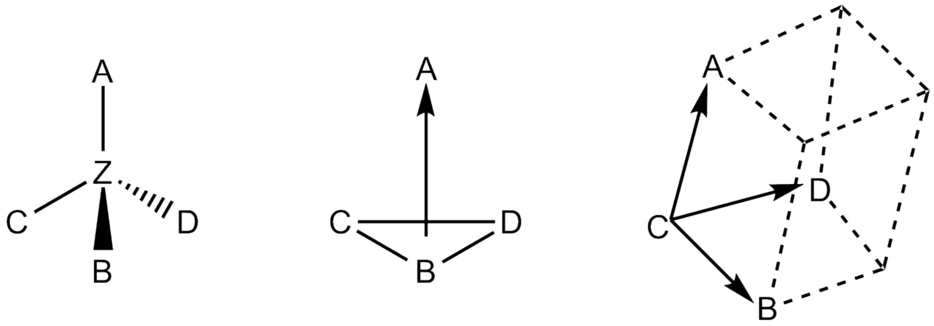


Figure 4.3: Scheme of a stereogenic center with the central atom Z. The atoms A to D have a different priority according to the CIP-rules.[312] The rhombic prism defined by the atoms A, B, C is used for classification of stereogenic center.

Listing 4.4: Class definition for determining and checking the stereocenters.

```
class StereoCheck {
private:
  struct stereocenter{
    int central_atom;
    int SubsA, SubsB, SubsC, SubsD;
    int orientation;
    double CA[3], CB[3], CD[3];
  };
  bool checkneighbours(stereocenter atom);
public:
  StereoCheck(Coord::coordinates &coords);
  Coord::coordinates* coords;
  void getStereoCenters();
  void getStereoDirections();
  bool compareStereo(std::vector<stereocenter> &stereo_centers);
  std::vector<stereocenter> stereo_centers;
};
```

To specify the orientation of a stereogenic center, the vector pointing from the plane BCD to the atom A is used. In the beginning of a search, the direction of this vector is determined. During the conformational search, the vector is not allowed to change its direction which would be equal to an inversion of the stereogenic center. The vector from the plane BCD to atom A is calculated using a triple product (Equation 4.5). The triple product gives the volume of a rhombic prism. The sign of the volume also gives the direction of the vector. As it is not necessary to determine each stereogenic center and classify it according the IUPAC, but only to determine the relative changes in the configuration, this information is sufficient.

$$\left(\vec{CB} \times \vec{CD}\right) \cdot \vec{CA} \tag{4.5}$$

The stereocheck is implemented after each optimized Basin Hopping step and after each local optimization in the Tabu-Search part. The class definition can be found in Listing 4.4.

4.1.6 Local optimization libraries

The local optimization algorithm is a crucial step in the global optimization within Tabu-Search. Therefore, an efficient approach is desired. The easiest way is the use of a library providing the necessary algorithms, as code debugging and maintenance of the algorithms is done by the distributor. One very popular collection of numerical approaches is the Numerical Recipes[269] providing code examples in C++. Using the third edition of Numerical Recipes, the algorithms Steepest Descent (SD) and Davidon-Fletcher-Powell (DFP), a quasi-Newton method, were implemented.
However, the Conjugate Gradient (CG) and Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithms provided by the numerical algorithm library ALGLIB[313] offer a better performance than the algorithms of Numerical Recipes. Of course, different combinations like a short optimization with SD or CG prior to an optimization with the more efficient BFGS algorithm are also possible.
For very large optimization problems, memory requirements become a problem for Newton and quasi-Newton methods. Therefore, the limited memory BFGS (L-BFGS) algorithm is much more efficient as it avoids the storage of the Hessian matrix. The L-BFGS library written in C provided by Naoaki Okazaki[314] is implemented into the CAST program and resulted in the best performance. Currently, this algorithm is the default choice for all optimization problems within the CAST program.

4.1.7 Transition state search algorithms

The localization of transition states (i.e. saddle points of first order) is of paramount importance for the understanding of reaction mechanisms. Two different methods have been implemented into the CAST program, the nudged elastic band (NEB) and the Dimer method. The NEB method belongs to the chain-of-state methods. A detailed description of the algorithm is given in subsection 3.4.3. The NEB method is used to calculate a minimum energy path between an initial and a final state. The algorithm is implemented into the class *NEB* and is initialized with a reference to an instance of the *Energy_all* class. The different variants of NEB (climbing image NEB, use of a better tangent estimation, doubly nudged elastic band) are specified by *bool* variables. An instance of this class is also used in the first step of the PathOpt algorithm (subsection 4.4). Tests of the algorithms are given in section 5.6.
A main bottleneck of the NEB method is the exact localization of the transition state. However, this can be done efficiently by the single-ended Dimer method (for a detailed description see subsection 3.4.2). Two different variants of the Dimer method have been imple-

mented. The original algorithm proposed by Henkelman[198] and a new variant in internal dihedral coordinates for an adaptation to the Tabu-Search (see subsection 4.2 for more detail about this method). The implemented algorithm following the publication of Henkelman[198] employs Cartesian coordinates. The algorithm can either be initialized with a random vector or a vibrational frequency vector for building up the Dimer. In the first case, the Dimer method contains statistical elements and converges to different transition states depending on the initial random vector. In the latter case, the algorithm always converges to the same saddle point of first order. The order of the saddle points was assigned by a frequency analysis with the program *vibrate* of Tinker program package.[37,250,305–308] The Dimer method in Cartesian coordinates is used within the PathOpt algorithm (subsection 4.4) for optimization to the closest transition states. An application example of this approach is given in subsection 5.6. The Dimer method in internal dihedral coordinates is used within the Tabu-Search and is discussed there (subsection 4.2).

4.1.8 Docking and X-Ray refinement module

The goal of this project was the establishment and testing of an algorithm for optimizing the orientation of a ligand within the active site of a protein. The project was partially done by Sebastian Brickel[315] and Lukas Pason[316] in the course their Bachelor theses. The flowchart of the complete algorithm is given in Figure 4.4. The CAST program requires force field parameter files in Tinker format. Therefore, the first step of the program is the conversion of force field files provided by e.g. the program NAMD[317] into a readable format for CAST.

The orientation of the ligand within the active site is influenced by the non-bonded interactions of the surrounding enzyme and water molecules. Therefore, when a ligand should be docked into an active site, these interactions have to be included. Usual docking algorithms employ empirical scoring functions to score the quality of a ligand pose.[318] In contrast, the global optimization with CAST uses force field energy functions. To reduce the computational effort, only the ligand and the closest surrounding is taken into account. Amino acids and water molecules that have previously been cut out have to be completed to describe the system properly. The tertiary structure of the enzyme should be retained. Therefore, parts of the protein surroundings have to be fixed during the global optimization. To account for flexible docking (induced fit) and reorientation of water molecules, parts of the enzyme and water can be allowed to relax after a minimum has been left. However, during the *modest ascent* of Tabu-Search, only the ligand is varied. After the ligand orientation is optimized, the new orientation is placed back into the complete protein for further calculations. Besides, the obtained data can be analyzed due to e.g. specific water-enzyme interactions, RMSD values of the systems, or different ligand orientations.

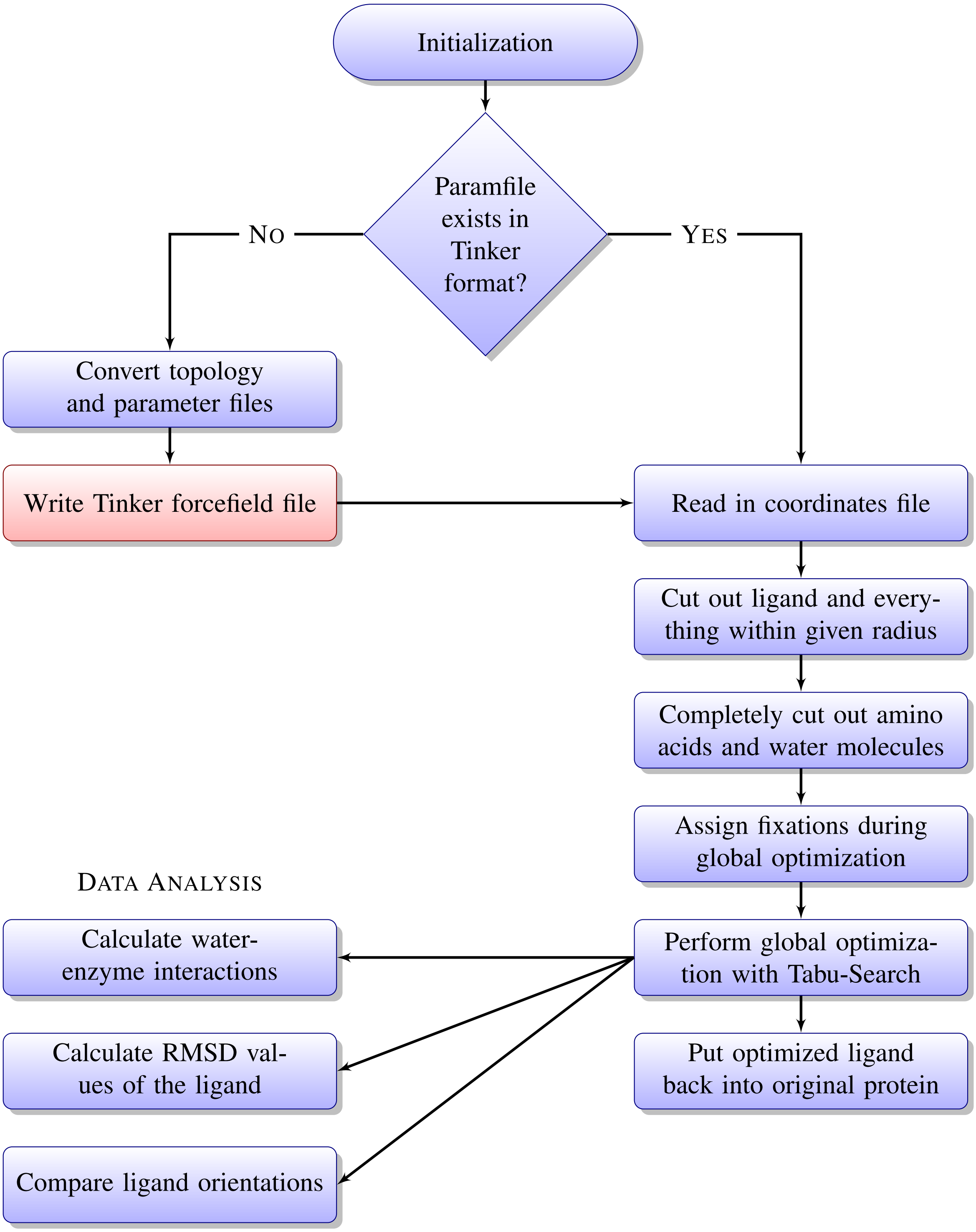

Initialization
Paramfile exists in Tinker format?
NO
YES
Convert topology and parameter files
Write Tinker forcefield file
Read in coordinates file
Cut out ligand and everything within given radius
Completely cut out amino acids and water molecules
Assign fixations during global optimization
DATA ANALYSIS
Perform global optimization with Tabu-Search
Calculate water-enzyme interactions
Calculate RMSD values of the ligand
Put optimized ligand back into original protein
Compare ligand orientations

Figure 4.4: Flowchart of the algorithm for performing global optimizations of ligand orientations within an active site of an enzyme.

4.2 Development of new Tabu-Search based global optimization algorithms

4.2.1 Implementation into CAST

The original Gradient Only Tabu Search (GOTS) developed by Svetlana Stepanenko[153,154,288] was implemented into the CAST program. Thus, the code was optimized and encapsulated into its own class. The implementations of the Tabu restriction rules were unchanged. Besides a general reimplementation of the Tabu-Search algorithm in object-oriented C++, the optimization included for example:

- Elimination of the perl scripts for calling ChemShell and coordinate conversion. These were replaced by C++ subroutines.
- Switching from ChemShell to internal force field implementations.
- Internal converter between Cartesian and internal coordinates.
- Removing unnecessary I/O as described in subsection 4.1.2.

The object-oriented implementation now allows for an easy modification of different subroutines like the *diversification search* or the *modest ascent*. The improvements are described below. Figure 4.5 shows the general flowchart of the Tabu-Search algorithm as implemented in CAST. During conformational search studies, fixations often have to be applied. The Tabu-Search implementation allows one to fix an atom in Cartesian coordinates during the complete search or to fix it only during the *modest ascent* part and let it relax in the *steepest descent* strategy. This allows for a relaxation of the fixed atoms into a new situation. The fixations for the modest ascent part are assigned during its initialization.
The optimizations within the Tabu-Search algorithm comprises three different topics:

- Efficient buildup of starting structures.
- Improvement of the diversification search.
- Better and more efficient modest ascent search.

These points are described in the following in more detail.

4.2.2 Basin Hopping and StartOpt

The improvements to Tabu-Search described in this subsection are published in ref. 52. The comparison of the efficiency of different global optimization algorithms with the Tabu-Search algorithm[52] revealed the slightly better performance of Basin Hopping in comparison to GOTS. Basin Hopping is a very wide scanning algorithm while the Tabu-Search is more efficient in the closer neighborhood. Therefore, the combination of the two algorithms

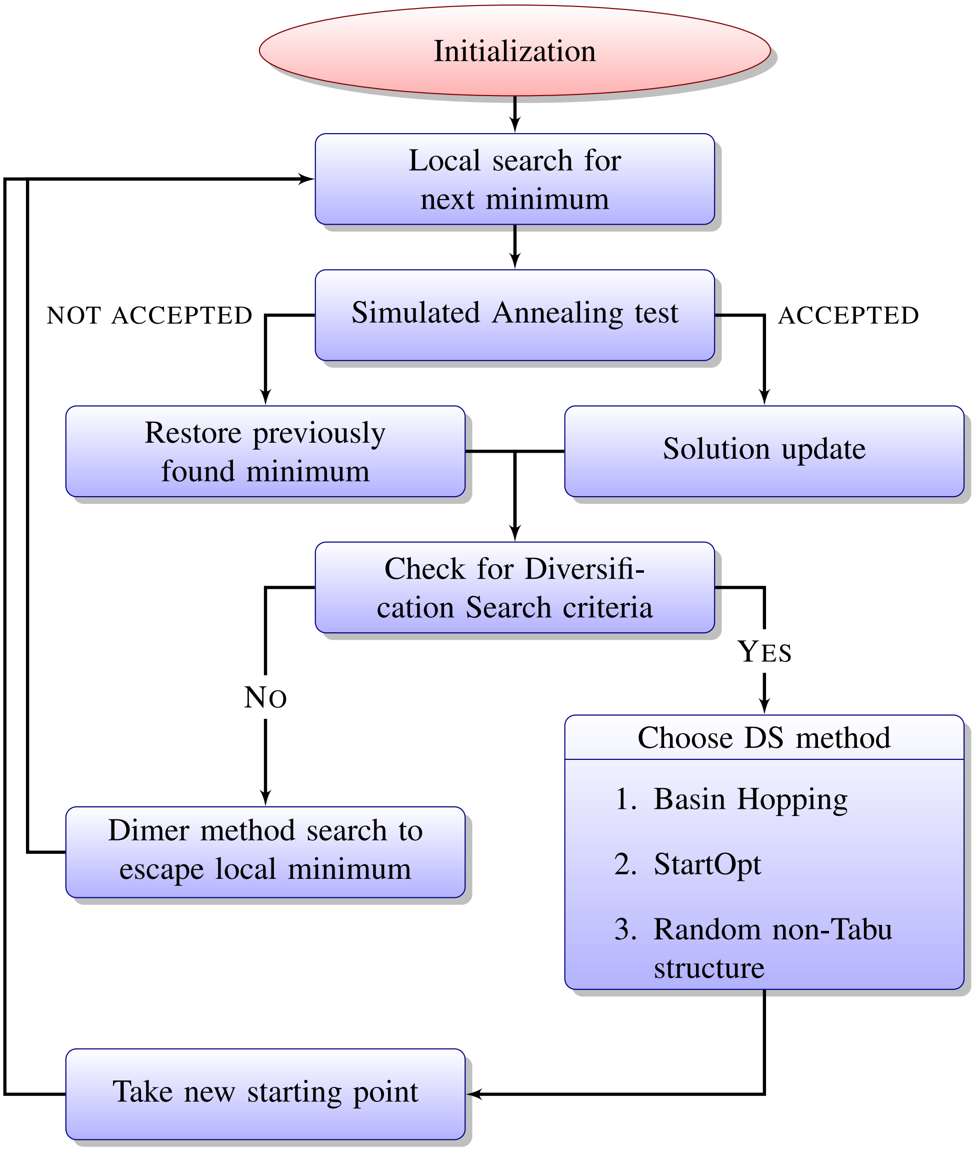
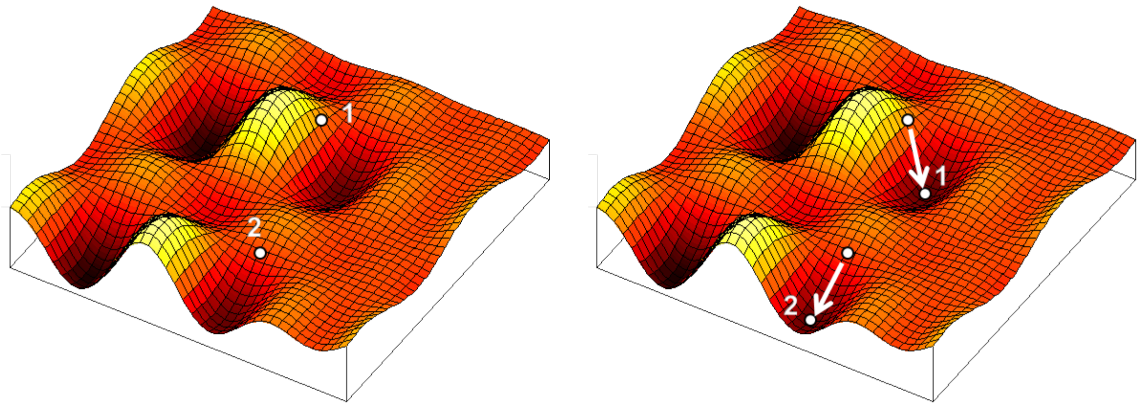

Initialization
Local search for next minimum
Simulated Annealing test
NOT ACCEPTED
ACCEPTED
Restore previously found minimum
Solution update
Check for Diversification Search criteria
NO
YES
Choose DS method
1. Basin Hopping
2. StartOpt
3. Random non-Tabu structure
Dimer method search to escape local minimum
Take new starting point

Figure 4.5: Flowchart of the latest Tabu-Search algorithm implemented in CAST combining all implemented modifications.

is expected to be be much more efficient. The benchmark indeed showed that the GOTS-BH algorithm is much more efficient than the other algorithms.[52] More details about the application example can be found in subsection 5.2. Here, the implementation of the Basin Hopping algorithm into the Tabu-Search approach is described in more detail.

Basin Hopping is a Monte Carlo based algorithm. The difference between simple Monte Carlo and Basin Hopping is illustrated in Figure 4.6. Before evaluating the Metropolis criteria to decide whether a trial point is accepted as new starting point or not, the trial point is minimized to its local minimum (i.e. basin). Thus, the probability of locating a low lying minimum is enhanced dramatically. The Tabu-Search itself is a locally very efficient algorithm. Therefore, rather wide random steps can be performed in the Basin Hopping sequences.

When applied to conformational search, the random steps can either be performed in Carte-



(a) Monte Carlo

(b) Basin Hopping

Figure 4.6: Comparison of simple Monte Carlo and Basin Hopping.

sian or internal coordinates. When using random steps in Cartesian coordinates (default value at the moment), all coordinates are varied at the same time. For random steps in internal coordinates, only dihedral angles are varied (see Equation 4.6). For definition of dihedral angles, the concept of Echenique and Alonso is applied.[319] Thus, a small random number m is calculated which determines the number of angles to be varied. Then, m random main dihedral angles k with their concerning dependent dihedrals are varied about a random step. In principle, Basin Hopping in internal coordinates should be more efficient. However, a detailed comparison between the two implementations was not performed.

$$\begin{aligned} m &= -\log\{\max(\text{Rand}[0;1], 0.0001)\} + 1 \\ k &= int\,\{N(\text{main dihedrals}) \cdot \text{Rand}[0;1]\} \end{aligned} \tag{4.6}$$

Often, BH finds a better minimum in the very first steps. However, experience revealed that the performance is improved when at least a small number of BH steps (10 BH steps in the recent implementation) is performed. As soon as a better non-Tabu minimum is found as the starting point, the BH sequence is stopped and Tabu-Search starts gain from this new structure. A maximum number of 100 BH steps is set for diversification runs. When no better structure is found after 100 BH steps, the last structure is taken if it is not already tabu. Otherwise, the BH search is continued for 80 steps. This is repeated with 60, 40, and 20 BH steps. If still no better or non-Tabu structure is found, the Tabu-Search is continued anyway from the last structure.

One disadvantage of the Basin Hopping algorithm is an easy inversion of a stereo center. Therefore, each structure found during the BH sequences is checked for all specified stereocenters. Steps, where a center was inverted, are rejected and the last point is taken again for a random step.

Besides the Basin Hopping algorithm, the use of reasonable starting structures can also increase the performance.[52] These structures can be obtained by the StartOpt algorithm[46–48] and can be used as starting points of the global optimization as done in the benchmark of different search methods.[52] However, the structures can also be used directly within the diversification search as new start points during the Tabu-Search. The desired structures have to be provided with the starting structure. During the diversification search, the program checks for alternative starting points. If new structures are provided, the algorithm takes these structures as new starting points. Otherwise, a Basin Hopping search is initialized. The different subroutines of the StartOpt algorithm are already described elsewhere.[46–48]

4.2.3 Dimer-Method

The original neighborhood search of GOTS employed a grid of energy function evaluation around the current point to determine the modest ascent. The direction with the lowest function increase is followed until the minimum is left. A detailed description of this algorithm can be found in the doctoral thesis of Svetlana Stepanenko.[288] A detailed performance investigation (see ref. 52 and Table A.4), however, revealed that the original modest ascent strategy is too inaccurate and inefficient. Therefore, as part of this work, a new approach was investigated. The Dimer-method[198–200] described in previous chapters is a single ended transition state search algorithm which only requires first derivatives and has a very efficient scaling behavior. Therefore, the Dimer-method was adapted to the Tabu-Search methodology. The implementation is also described in ref. 320.

In conformational search, the dihedral coordinates involving single or hydrogen bonds represent the softest degrees of freedom. It is therefore quite natural to vary only these coordinates in the modest ascent part of the Tabu-Search. The exact determination of the modest ascent as well as the transition state is, furthermore, not necessary as the Tabu-Search only needs to leave a minimum very quickly. This allows for rather large step sizes. The theory of the Dimer-method was already described in subsection 3.4.2. In the following subsection, details about the implementation into Tabu-Search are discussed.

A very important aspect is the proper definition of dihedral angles. Thus, the principle of Echenique and Alonso was followed,[319] which ensures a proper rotation of groups. The definition is straightforward for molecules. However, in larger molecular clusters it has to be guaranteed that the dihedral angles are defined between atoms in close proximity. To ensure the proper indexation of the dihedral angles, it is first checked whether indices of connected atoms can be used (which is the preferred situation as a dihedral angle between connected atoms is more natural). Thus, only atoms with lower atomic index numbers can be used. If no connected atom is present, the closest atom is taken by checking inter-atomic distances. During the *steepest descent - modest ascent* steps, the distances between atoms change. Therefore, the indexation is created during each initialization of the *modest ascent* part. At this point the fixations during the modest ascent are assigned as well. In contrast to optimization in Cartesian coordinates, several dihedrals can move an atom. Therefore, all dihedral angles depending on a fixed atom have to be fixed as well. Even with a careful definition of dihedral angles, atomic centers can sometimes come too close to each other. Therefore, a step-scaling routine was implemented which checks for inter-atomic distances which are too small. In such cases, the steps within the Dimer method are scaled down.

To employ the Dimer-method in torsional space, dihedral derivatives have to be used. Usually, the gradients are given in Cartesian coordinates. Therefore, the Cartesian gradients have to be converted into torsional derivatives. This is done by a projection of the Cartesian coordinate onto the torsional angle components. A torsional angle is defined by four atoms A, B, C, D. The two central atoms (B, C) define the rotatable bond. The projection is calculated and summed up for each atom bound to B and C giving the torsional gradients for the dihedral angle A, B, C, D. For details see Equation 4.7. Here, B is the first atom defining the rotation axis of the dihedral angle, C, is the second atom, X are all bound atoms. $\nabla(\Theta_i)$ is the dihedral gradient.

$$\overrightarrow{CB} = \begin{pmatrix} x_b - x_c \\ y_b - y_c \\ z_b - z_c \end{pmatrix} \quad ; \quad \hat{CB} = \frac{\overrightarrow{CB}}{|CB|} \quad ; \quad \overrightarrow{BX} = \begin{pmatrix} x_x - x_b \\ y_x - y_b \\ z_x - z_b \end{pmatrix}$$
$$\overrightarrow{tors_X} = \hat{CB} \times \overrightarrow{BX} \quad ; \quad \nabla(\Theta_i) = \sum_X \nabla(E) \cdot \overrightarrow{tors_X} \tag{4.7}$$

The initial dimer is generated by distorting the starting point by a unified random vector and a step size of 10 degrees for each dihedral angle.

$$\Phi_{dih1_i} = \Phi_{dih0_i} + \hat{\tau}_i \cdot 10^\circ \tag{4.8}$$

From this dimer, the energy and gradients are calculated for x_0 and x_1. x_0 represents the dimer midpoint. The dimer is only created in one direction to save one energy and gradient calculation following the publication of Kästner.[200] Therefore, x_0 is the starting point of the dimer and contains the rotational axis. x_1 is the dimer endpoint. Now, the first rotation is performed. The starting structure is a local minimum. Therefore, no translation can be performed for the first step as the gradient F_0 is zero (see Equation 4.19).

The rotation is performed until either an estimated angle Φ_1 or the calculated minimum angle Φ_{min} is smaller than a given threshold (10 degrees in the current implementation) or a maximum of rotational steps is reached (10).

First, the rotational force F_R is calculated:

$$F_R = -2\,(F_1 - F_0) + 2\,[(F_1 - F_0) \cdot \hat{\tau}]\,\hat{\tau} \tag{4.9}$$

With the rotational force, the step direction can be calculated using an optimization algorithm. For the steepest descent the step direction is:

$$\Theta = \frac{F_R}{|F_R|} \tag{4.10}$$

Using the more elaborate Conjugate Gradient algorithm, information about the previous step is taken into account:

$$\Theta = F_{Ri} - \gamma \cdot F_{Ri-1} \tag{4.11}$$

The curvature of the PES in the direction of τ can be obtained from the gradients F_1 and F_0

$$C_\tau = \frac{(F_1 - F_0) \cdot \hat{\tau}}{\Delta} \tag{4.12}$$

where Δ is the difference between the dihedral angles of x_1 and x_0.

The optimal rotation angle Φ_{min} can be obtained by minimizing the curvature C_τ in the plane spanned by τ and Θ. This would need one further gradient calculation. However, a rough estimate can be obtained from F_0 and F_1. The rotation is carried out only if the estimated angle Φ_1 is larger than the given tolerance Φ_{tol}.

$$\Phi_1 = -\frac{1}{2} \arctan \frac{\frac{\partial C_\tau}{\partial \Phi}}{2|C_\tau|} \tag{4.13}$$

$$\frac{\partial C_\tau}{\partial \Phi} = \frac{2\,(F_1 - F_0) \cdot \Theta}{\Delta} \tag{4.14}$$

The dimer is rotated about the product of Φ_1 and the corresponding Θ. This means that each dihedral angle of x_0 is rotated about this product to give a new dimer. For this new dimer endpoint x_1', energy and gradients F_1' are again calculated.

With these new structures the optimal rotation angle is calculated:

$$\Phi_{min} = \frac{1}{2} \arctan \frac{b_1}{a_1} \tag{4.15}$$

$$b_1 = \frac{1}{2} \frac{\partial C_\tau}{\partial \Phi} \Big|_{\Phi=0} \qquad a_1 = \frac{C_\tau \left|_{\Phi=0} - C_\tau \right|_{\Phi=\Phi_1} + b_1 sin\,(2\Phi_1)}{1 - cos\,(2\Phi_1)} \tag{4.16}$$

$$\frac{\partial C_\tau}{\partial \Phi} = \frac{2\,(F_1 - F_0) \cdot \Theta}{\Delta} \tag{4.17}$$

If the curvature at Φ_{min} is larger than the initial curvature, $\pi/2$ is added to the rotational angle.

At the new position x_{min}, the gradients can either be calculated or estimated to save one gradient calculation. The estimation is done by:

$$\begin{aligned} F_{min} =& \frac{\sin\left(\Phi_1 - \Phi_{min}\right)}{\sin\left(\Phi_1\right)} F_1 + \frac{\sin\left(\Phi_{min}\right)}{\sin\left(\Phi_1\right)} F_1' \\ &+ \left(1 - \cos\left(\Phi_{min}\right) - \sin\left(\Phi_{min}\right) \tan\left(\frac{\Phi_1}{2}\right)\right) F_0 \end{aligned} \tag{4.18}$$

After the rotation is converged, the dimer is translated. The translational force (F_T) is calculated according to:

$$F_T = \begin{cases} -\left(F_0 \cdot \hat{\tau}\right)\hat{\tau} & \text{if} \quad C_\tau > 0 \\ -F_0 + 2\left(F_0 \cdot \hat{\tau}\right)\hat{\tau} & \text{if} \quad C_\tau < 0 \end{cases} \tag{4.19}$$

The dimer is then translated by altering all dihedral angles about the translational force. Thus, the main dihedral and its concerning dependent dihedrals are translated about the same value.

4.3 Solvation with Tabu-Search

In the following, a new approach is presented for solvating molecules using the Tabu-Search algorithms described in the present work. The description and application of the algorithm is published in ref. 320. The focus lies on an accurate description of the micro-solvation. The micro-solvation has a significant influence on a wide range of processes.[320] Furthermore, an accurate description is important for the quality of QM/MM calculations. For bigger systems, the solvent shell cannot be created by a systematic approach. Therefore, reliable methods have to be employed. The standard approach for building up the solvent shell in QM/MM investigations comprises the following steps. The protein is surrounded by pre-optimized water spheres. Everything within a certain range of the protein is removed to avoid clashes of the nuclei and the complete system is equilibrated by long molecular dynamics (MD) runs involving heating and cooling periods. MD simulations qualitatively reproduce the physical motion of an NVT ensemble. Therefore, bulk effects should be covered accurately. Even so, the results for the important inner water molecules may be questionable, since conformational search MD simulations often do not find the global minimum structure.[52]

In the standard MD approach, additional potentials at the boundary are necessary to avoid evaporation of water molecules. Consequently, in the standard MD-based approach the whole solvent shell has to be added at once since additional potentials will distort the structure of the solute for small solvent shells. In the new Tabu-Search based approach, an evaporation of water molecules cannot take place as only dihedral angles are modified in the modest ascent part. Hence, the solvent shell can be built up step-wise. This has the advantage that the optimization of the very important first solvation shell which interacts directly with the solute (micro-solvation) can be performed more carefully since a considerably smaller system has to be treated. After optimization of the first shell the second, third, etc. shell is added. In each step, a global optimization is performed to adapt all water molecules to the increasing solvent shell. For these outer shells which do not interact directly with the solute but are only responsible to cover bulk effects, less precise requirements are necessary in these optimizations. It is of course also possible to add all water molecules at once; however, it has been shown that the step-wise approach is superior.

The algorithm (implemented as a python script) can perform the step-wise solvation using two different approaches. The first is using the external software VegaZZ[321] (or the corresponding command line driven version). The other option is the calculation of an Accessible Surface Area (ASA) first described by Lee and Richards[322]. The ASA can be calculated using the Shrake-Rupley algorithm.[323] The information obtained from the ASA can then be used for placing the water molecules around the system.

The approach is always initialized with the structure of the solute. The first step is the calculation of the ASA from which a maximum radius and an initial radius are estimated. Thus, the molecule is approximated as a sphere and the initial radius of the solvent shell is obtained by:

$$R_{in} = \sqrt{\frac{ASA}{4\pi}} \tag{4.20}$$

In the next step of the solvation procedure, the solute is surrounded by water molecules within the radius R_{in} around the geometric center. The water molecules can be placed directly from the solvation script by using either the Vega command line program or the ASA-values. Up to a given size of the ASA-value, a water molecule is placed near the concerning atom at a vector pointing from the geometric center of the solute to the concerning atom. This system is then globally optimized using the Tabu-Search algorithm. The best solution obtained from this global optimization is then used as next starting point of the solvation. The radius is iterated by a given factor and the next solvation shell is placed around the geometric center of the solute which is followed by a global optimization. These iterations are continued until the final solvent shell size is obtained. The step-wise built-up of the solvent shell offers some flexibilities. They can be used to obtain information about the accuracy of the underlying force field and the starting structure which is often taken from experiment. In the first variant, the solute is fixed to the (experimental) starting structure during the build-up of the solvent shells. The equilibration of the solute can already be started after the build-up of the first few solvent shells, but in the present work the optimization was performed after the build-up was completed. In the following, this strategy will be called *Tabu-Search fixed*. In a second variant, the solute is equilibrated from the very beginning (*Tabu-Search free*).

The outcome of both variants will be different if either the force field or the (experimental) starting structure contain errors. As shown later (section 5.4), in variant 1 (*Tabu-Search fixed*) the obtained structure of the enzyme is biased with respect to the (experimental) starting structure, since the cage formed by the solvent shells restricts geometrical changes of the

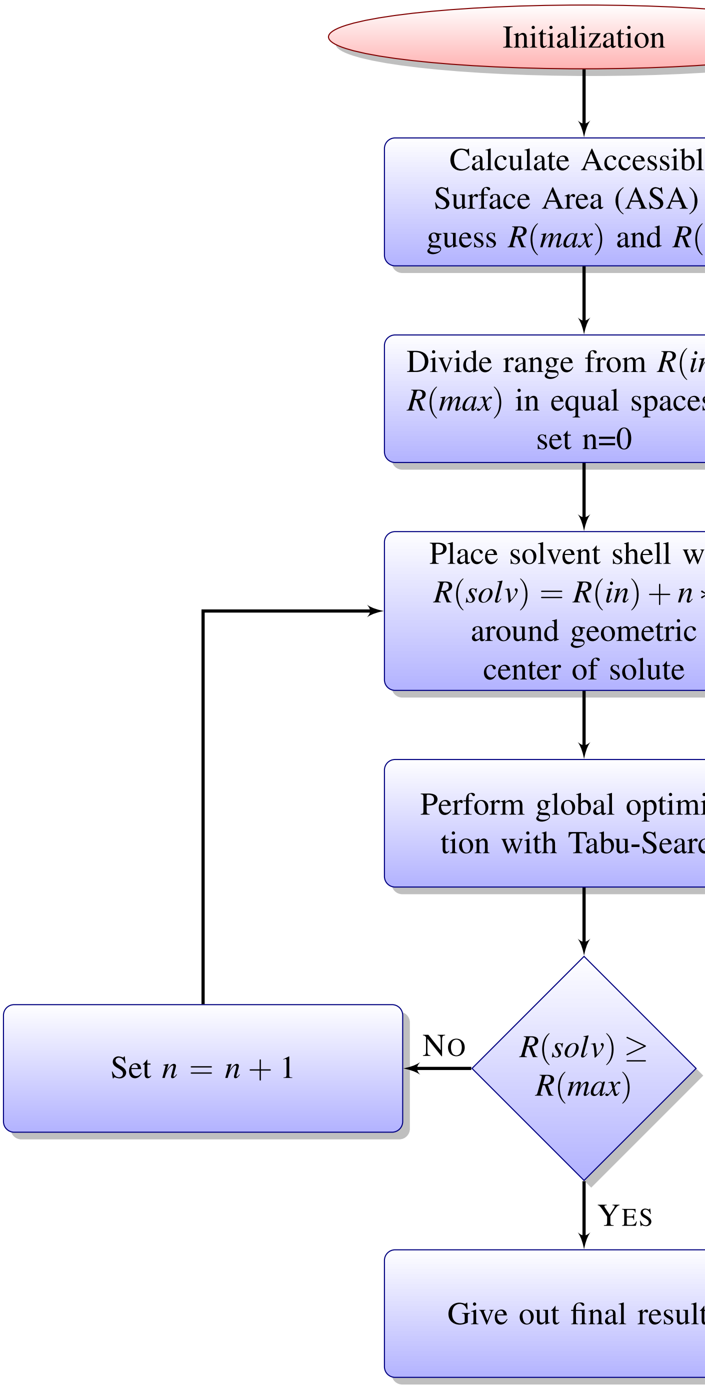

Initialization
Calculate Accessible Surface Area (ASA) to guess $R(max)$ and $R(in)$
Divide range from $R(in)$ to $R(max)$ in equal spaces (x) set n=0
Place solvent shell with $R(solv) = R(in) + n * x$ around geometric center of solute
Perform global optimization with Tabu-Search
$R(solv) \geq R(max)$
NO
Set $n = n + 1$
YES
Give out final results

Figure 4.7: Flowchart of the solvate algorithm using Tabu-Search based global optimization.

solute. This results from the modest ascent strategy. It preferentially varies the relative orientation of the outer solvent shell molecules since such variations need less energy efforts than modifications in the more rigid inner part. This is particularly the case if a whole residue changes its orientation. Nevertheless, smaller adaptions to the solvent shell are still possible. Hence, if an erroneous (experimental) starting structure is taken it will be retained in variant 1. In variant 2 (*Tabu-Search free*), since only very small hindrances are present, the solute will adopt geometries as calculated according to the underlying computational method (e.g. the force field) during the build-up of the first few solvation shells. So, if the starting structure is wrong, but the force field is sufficiently accurate, the correct structure is obtained in the second variant. If the (experimental) starting structure is correct but the force field is wrong, variant 1 will still provide the correct geometry but variant 2 will yield a wrong structure. Hence, significant differences in the outcome of both procedures point to deficiencies either in the force field or in the experimental structure.

As for MD, it is also possible for the Tabu-Search to start the equilibration with the complete water shell. In comparison with the other possibilities this approach will be called *Tabu-Search complete*. A detailed investigation of the proposed algorithm as well as a comparison with the standard MD approach is given in section 5.4.

4.4 PathOpt - a global transition state search algorithm

PathOpt is an algorithm for investigating reaction pathways between a reactant and a product state. For complex systems, often several possible reaction ways leading over different transition states are possible. The discovery of such pathways is a crucial process in understanding the underlying reaction mechanisms and several methods are already known.[124,195,207,210,211,214,215,217,218,224]

The flowchart of the algorithm is given in Figure 4.10. An illustration of the methodology is shown in Figure 4.8. As usual for most double-ended search methods, an initial alignment of reactant and product state is essential for reliable results. In the present study, alignments were performed with VMD,[324] however, every alignment algorithm is adequate. These structures are used to create an initial path by a simple linear fit between them. This path is the same as the initial path for an NEB[202–204] calculation. The middle point of this path is used to create an (n-1) dimensional hyperplane perpendicular to the initial path. This hyperplane is searched by global optimization. In the present implementation, this was done by employing the Basin Hopping approach.[115] In future, the Tabu-Search based method developed and discussed in the present work should also be implemented.[52,153–155,320] As can be seen from Figure 4.8, each minimum on the perpendicular hyperplane corresponds to the trace of a saddle point of first order.

To ensure, that the random steps within BH remain in the perpendicular hyperplane, the normalized vector pointing from reactant to product placed at the middle point is used ($\hat{\tau}$). The cross product of each random point with $\hat{\tau}$ gives a random point lying in the hyperplane (see Equation 4.21).

$$r_{perp} = r_{rand} \times \hat{\tau} \tag{4.21}$$

The local optimizations in BH are performed with an L-BFGS algorithm[270,271] using a modified gradient (g_{proj}) which is projected onto the hyperplane (Equation 4.22). The projected gradients are visualized in Figure 4.9.

$$\begin{aligned} g &= \nabla(f) \\ g_{proj} &= g_{normal} - (g_{normal} \cdot \hat{\tau})\hat{\tau} \end{aligned} \tag{4.22}$$

Solutions of BH are accepted if the new point is either lower in energy than the previous solution or the Metropolis criterion is fulfilled (Equation 4.23).

$$\text{accept if:} \qquad r < \exp\left(-\frac{E_2 - E_1}{RT}\right) \tag{4.23}$$

The accepted points resulting from global optimization are optimized to their closest transition state using the Dimer-method.[198–200] The lowest imaginary frequency is used to initialize the Dimer-search. Sometimes, the point obtained after local optimization in the perpendicular hyperplane possesses more than one imaginary frequency. However, only one frequency is considerably different to zero. The other imaginary frequencies correspond to either rotational or translational degrees of freedom. The same is true when only imaginary frequencies close to zero are obtained. These frequencies are not considered for optimization to transition states. The minima connected via the resulting transition state are obtained by local optimization along the one imaginary frequency. The received path fragments are analyzed for their connection to either the initial or final state. This is done by testing if only one distinct transition state lies between one of the end points of the fragment and the initial or final state. The resulting paths are taken for further PathOpt iterations (preferably the longest path). The search is stopped if convergence criteria like maximal iteration number or reasonable number of paths are fulfilled. If this is the case, they are appended to the concerning path. Otherwise, the longest connected path is taken for further PathOpt iterations. A repetition of the described algorithm finally delivers complete pathways from reactant to product via different transition states and intermediate minima. Example applications are given in section 5.6.

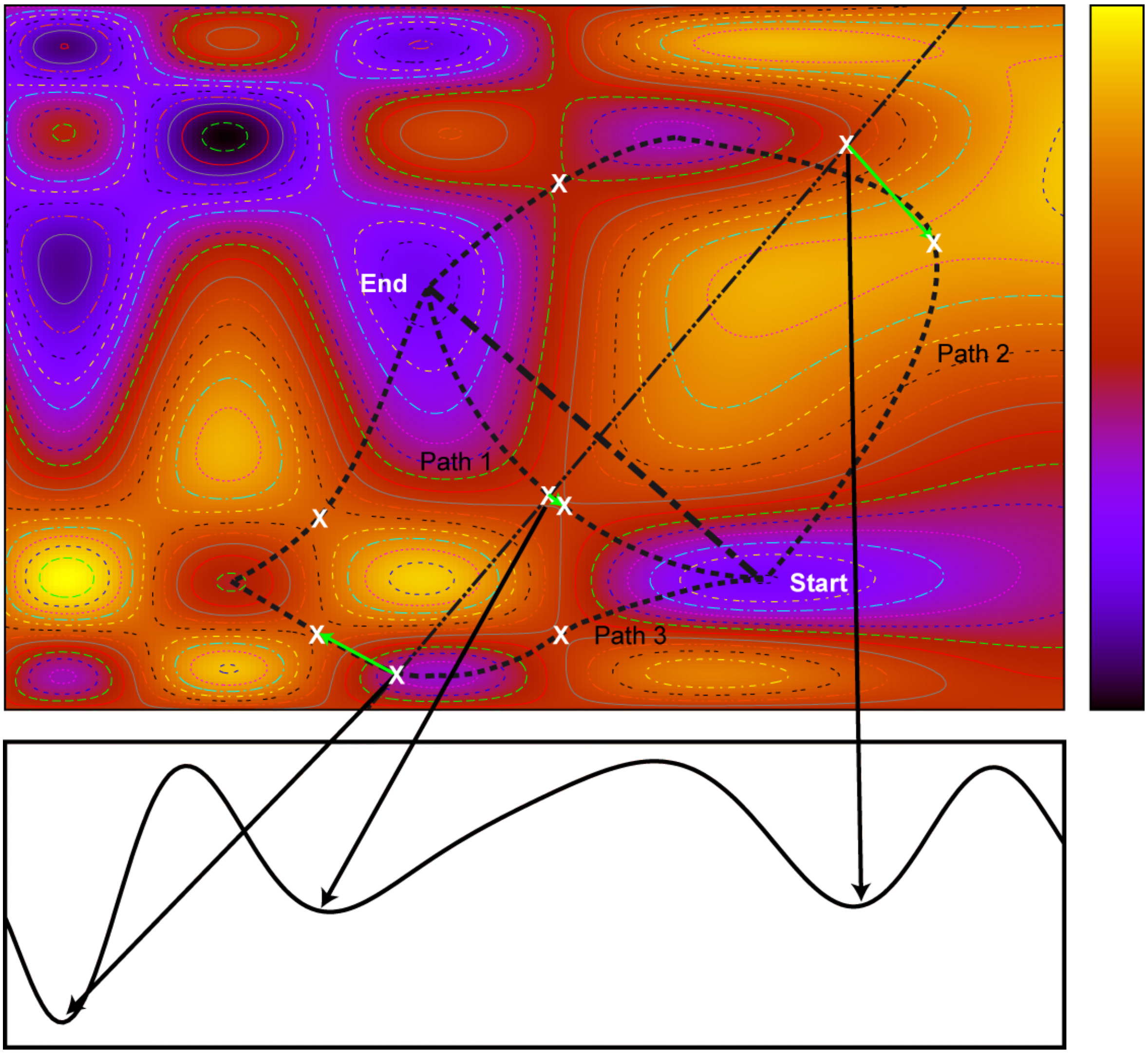


Figure 4.8: Illustration of the PathOpt algorithm: The search starts with an initial linear fit between the reactant and the product. Points leading to potential transition state are located by global optimization on the plane perpendicular to the initial path (i.e. a n-1 dimensional hyperplane). Accepted points are optimized to the closest transition state.

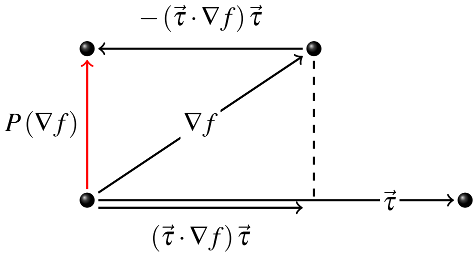

$-(\vec{\tau}\cdot\nabla f)\,\vec{\tau}$
$P(\nabla f)$
∇f
$\vec{\tau}$
$(\vec{\tau}\cdot\nabla f)\,\vec{\tau}$

Figure 4.9: Illustration of the projection of gradients onto the perpendicular plane according to Equation 4.22.

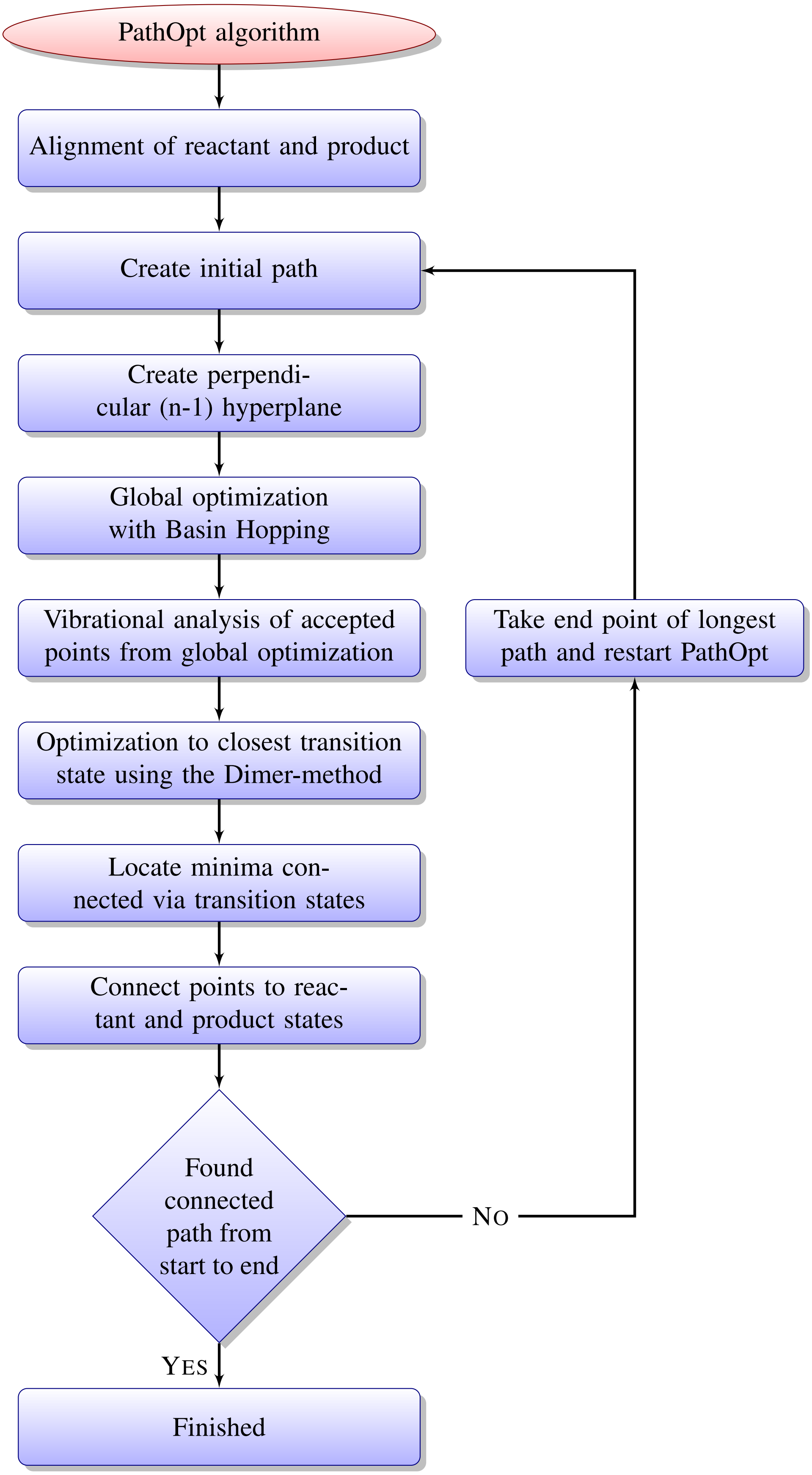

PathOpt algorithm
Alignment of reactant and product
Create initial path
Create perpendicular (n-1) hyperplane
Global optimization with Basin Hopping
Vibrational analysis of accepted points from global optimization
Optimization to closest transition state using the Dimer-method
Locate minima connected via transition states
Connect points to reactant and product states
Found connected path from start to end
No
Take end point of longest path and restart PathOpt
Yes
Finished

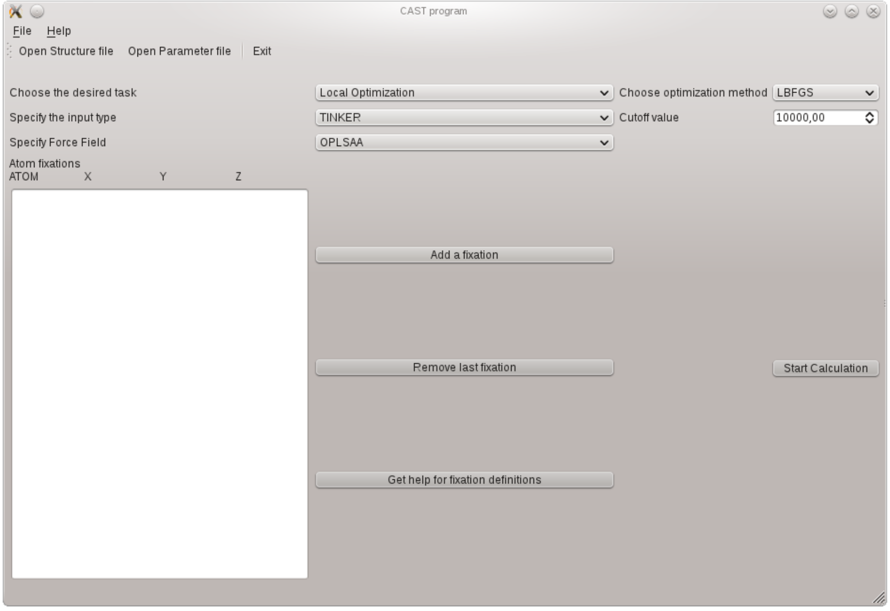
Figure 4.10: Flowchart of the PathOpt algorithm.

4.5 GUI-Development

Graphical User Interfaces (GUI) are very helpful to enhance the usability of a program. The GUIs designed in this work are written with QT4 and C++. QT4 is a platform independent library for designing graphical user interfaces. This has the advantage, that a GUI written under Linux can, in principle, also be used under Windows. Furthermore, the source code of the original command line-based program can be reused in the GUI-based program which simplifies the maintenance of both programs.

4.5.1 CAST

The CAST program is usually used via command line under Linux environment. To enhance the user-friendliness, a graphical user interface (GUI) was designed. A screenshot can be found in Figure 4.11. In principle, the GUI for CAST replaces the input files. The core source code of the program remains the same, it is simply implemented into the own class *CAST*. The main window contains several drop-down menus to choose the desired task, specify the input type, select a forcefield, or apply fixations during an optimization. After loading the input structure, the calculation can be started by pushing the "Start Calculation" button. If further input is required, the user is now asked to provide the necessary information. The output of the calculations is given in a separate window.



CAST program

Choose the desired task: Local Optimization

Specify the input type: TINKER

Specify Force Field: OPLSAA

Choose optimization method: LBFGS

Cutoff value: 10000,00

Atom fixations
ATOM X Y Z

Figure 4.11: Graphical User Interface for the CAST program.

Listing 4.5 shows the source code of the main function of a QT4 program. This function is very similar for most QT4-programs. The line "setlocale(LC_NUMERIC,"C");" is needed to override the local settings of the operating systems. Otherwise, on German systems the coordinates have to be given with comma instead of points. This would require a conversion of all standard coordinate file formats. In the main function, an instance of the MainWindow class is created and started (see Listing 4.6). Only if this object is stopped, the program returns to main and finishes the GUI. Different from usual class definitions in C++, QT4 header files further contain slots. Slots are used to manage the communication between e.g. a button in the GUI and a certain subroutine. The constructor of the MainWindow class creates the complete windows and sets up the connections between buttons, drop-down menus, or text fields with the concerning subroutines and creates an instance of the CAST class. The information provided in the GUI of the MainWindow is passed on to the CAST object. After the calculation with CAST is finished, the program returns to the MainWindow. There, a new calculation can be started. The program can be finished by pressing the "Exit" button.

Listing 4.5: main.cpp of the CAST-GUI.

```
#include <QtGui/QApplication>
#include <QLocale>
#include "mainwindow.h"

int main(int argc, char *argv[]) {
    QApplication a(argc, argv);
    setlocale(LC_NUMERIC,"C");

    MainWindow w;
    w.show();
    return a.exec();
}
```

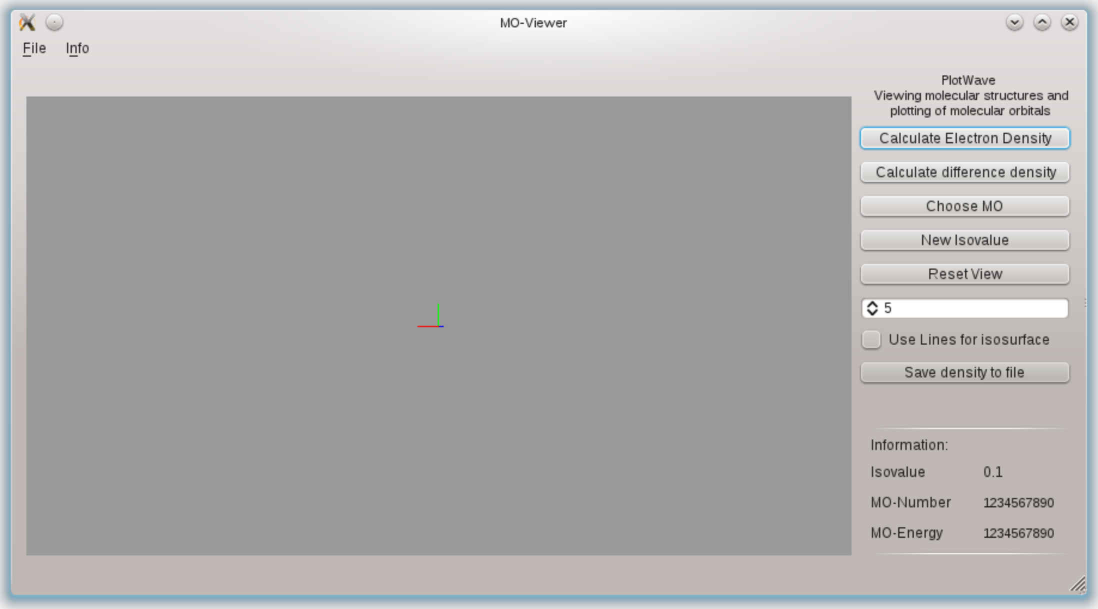
Listing 4.6: Header file of the MainWindow class of the CAST-GUI.

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QMainWindow>
#include "cast.h"
...
class MainWindow : public QMainWindow
{
    Q_OBJECT
public:
    MainWindow(QWidget *parent = 0);
    ...
private slots:
    void newFile();
    void openstructure();
    ...
private:
    void createActions();
    void createMenus();
    ...
    QMenu *fileMenu;
    ...
    QToolBar *fileToolBar;
    ...
    QAction *newAct;
    QAction *openAct;
    ...
    CAST *cast;          //object of CAST, central widget
};
#endif // MAINWINDOW_H
```

4.5.2 PlotWave

Program design PlotWave is designed as a molecular viewer to display tinker, Cartesian, and PDB files. Furthermore, standard output of molecular orbital (MO) calculations in Molden and TeraChem format is supported. The program can be run using both, Linux and Windows. The GUI is developed using QT4, while the visualization of atoms and densities is done using OpenGL. The remaining code is written in standard C++. The main aim of the program is the fast calculation of molecular orbitals and electron densities which can be easily put out to a file. A screenshot of the program can be found in Figure 4.12. The central widget (dark gray) contains the OpenGL widget which is used for data visualization. The buttons on the right allow the user to control the program. Rotating and moving of the molecule can easily be done using the mouse.
The molecular orbital which should be visualized is chosen from a drop-down menu by clicking on "Choose MO". The button "New Isovalue" allows for adjusting the numeric value for displaying the isosurface. The information about the current MO is given in the right bottom corner. After calculating the electron density by clicking "Calculate Electron Density", this information can also be saved to a file.

Calculation of molecular orbitals and electron densities The main aim of the viewer program is the calculation and visualization of molecular orbitals and electron densities. The calculations are divided and placed into several C++-classes. First, the input file has to be read and the basis sets have to be extracted and determined. With the obtained data, the set of primitive and contracted Gaussian functions needed for the basis sets is created. With the Cartesian coordinates of the molecule, a grid box is created containing the complete system. At each grid point, the values of the contracted Gaussian functions are calculated. When a particular orbital is to be plotted, the grid points are multiplied with the MO-coefficients obtained in the SCF-calculation. This improves the performance of the program as only one multiplication is necessary instead of recalculating the grid for the complete molecule.


MO-Viewer
File
Info
PlotWave
Viewing molecular structures and plotting of molecular orbitals
Calculate Electron Density
Calculate difference density
Choose MO
New Isovalue
Reset View
5
☐ Use Lines for isosurface
Save density to file

Information:

Isovalue	0.1
MO-Number	1234567890
MO-Energy	1234567890

Figure 4.12: GUI of the molecular viewer PlotWave.

The class *Primitives* contains the equations for primitive Gaussian functions of s-type and p-type (Φ_x^{PG})[325]:

$$g_s = \left(\frac{8\alpha^3}{\pi^3}\right)^{1/4} \exp^{-\alpha^2} \tag{4.24}$$

$$g_p = \left(\frac{128\alpha^5}{\pi^3}\right)^{1/4} x\exp^{-\alpha^2} \tag{4.25}$$

The class *Contracted* inherits from *Primitives* and constructs the contracted Gaussian functions. Equation 4.26 for example shows a contracted s-type function consisting of three primitive functions.

$$\Phi_{1s}^{CG} = \sum_{i=a}^{3} d_{1s_i} \Phi_{1s}^{PG}(r, \alpha_{1s_i}) \tag{4.26}$$

The class *Atomicbasis* constructs the basis sets using the needed entities of *Contracted* (in principle, basis sets consist of a sum of contracted Gaussian functions).

The class *Molorb* manages the calculation of molecular orbitals and electron density. The class can be controlled from outside the class using public functions and variables. These functions are used in the QT-framework.

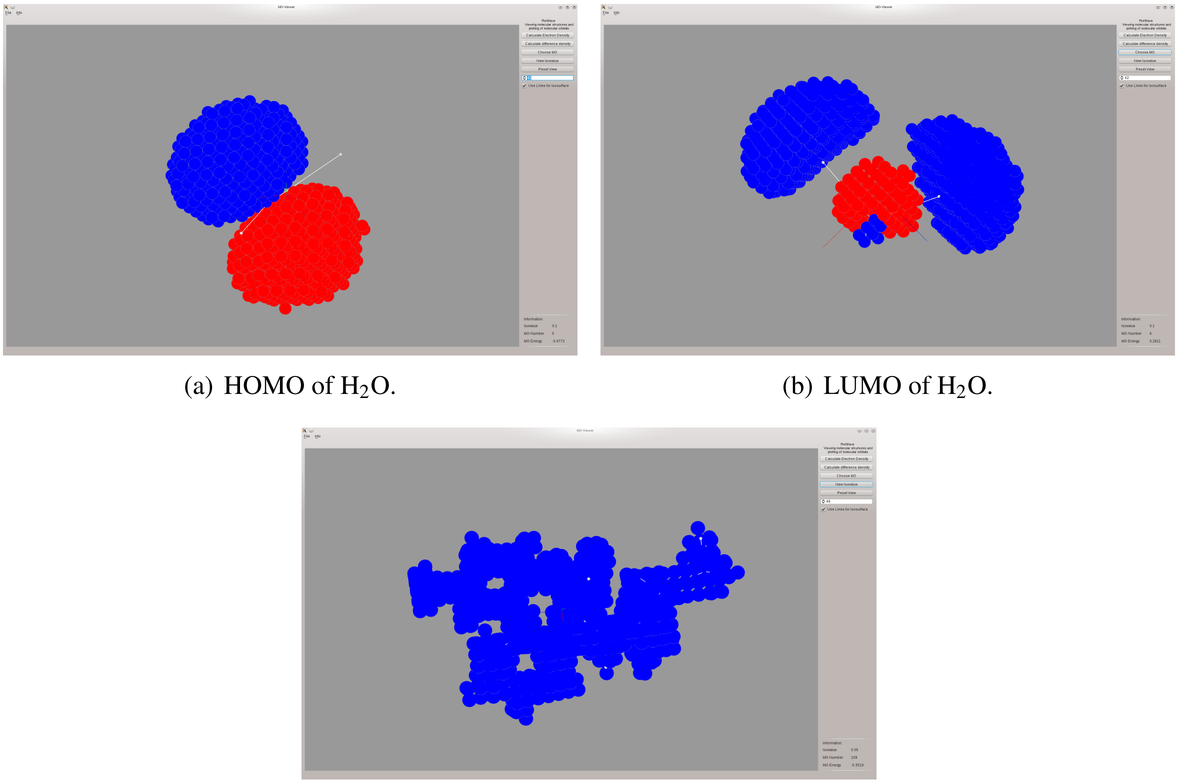
A molecular orbital is the sum of all contracted Gaussian functions multiplied with the corresponding MO-coefficient.[19]

$$\Psi_i = \sum_r c_{ri} \Phi_r \tag{4.27}$$

The electron density is the sum over squares of all occupied orbitals.[19]

$$\rho(r) = \sum_{k}^{N_{occ}} n_k \left|\Psi_k(r)\right|^2 = \sum_{k}^{N_{occ}} n_k \left|\sum_{i}^{basis} c_{ki}\Phi_i\right|^2 \tag{4.28}$$

Figure 4.13 shows some examples of the molecular viewer program.


MO-Viewer
File
PlotWave
Viewing molecular structures and plotting of molecular orbitals
Calculate Electron Density
Calculate difference density
Choose MO
New Isovalue
Reset View
42
Use Lines for isosurface

Information:
Isovalue 0.1
MO-Number 5
MO-Energy -0.4773

(a) HOMO of H_2O.

Information:
Isovalue 0.1
MO-Number 6
MO-Energy 0.2611

(b) LUMO of H_2O.

Information:
Isovalue 0.05
MO-Number 109
MO-Energy -0.3519

(c) Electron density of an exemplary molecule.

Figure 4.13: Examples of molecular orbital and electron density visualization with PlotWave.

Viewing output of PlotWave in the x-ray refinement program COOT A standard program for x-ray refinement is COOT from the CCP4 development group.[326] COOT can read coordinates from PDB-files and experimental input files containing the information from an x-ray experiment (so-called MTZ-file). The input and output files from COOT are written in binary format. This makes the analysis and modification very difficult. The MTZ file contains the information about phases and amplitudes after the refinement process. This data delivers the electron density after a Fourier transform.[327]

To compare the orientation of a globally optimized ligand within the active site of an enzyme with the experimental x-ray structure, one can compare the electron density of the ligand (obtained from a SCF calculations) with the experimental one.

As the data types are very different there are generally two ways for the comparison:

1. Take the experimental electron density, write down a density map or perform a Fourier

transform of given functions and compare the density within a program different to COOT.

2. Take the theoretically derived electron density, convert it into a COOT readable format and compare the densities in COOT.

To investigate the first method, COOT files have to be analyzed. As all files in COOT concerning densities are written in binary format, these files cannot be looked at directly. The exact structure of the files has to be known (which requires a lot of reverse-engineering as exact descriptions are scarce).
CCP4 provides a set of libraries written in the programming language C.[328] These libraries give access to the data structures used in COOT. First, the MTZ-file was analyzed and decomposed (using the *cmtzlib*) in all relevant parts (mainly the phase, amplitude and coordinates information). Then, the data was rewritten onto normal text files to be usable by other programs as well. For the given example, the density contains about 80000 functions (for each function a separate phase and amplitude). One has to perform a Fourier transform of these functions, sum up all points and then convert the coordinates from reciprocal space to Cartesian space.
Using the first method, a lot of file conversions are necessary and Fourier transformations have to be implemented. All these functionalities are already implemented in COOT. Therefore, the second way was also investigated. The electron density calculated by the molecular viewer described above was taken as an input. The CCP4 library *cmaplib* provides the function to write or modify a density map file readable by COOT. The calculated electron density was converted to MAP-format, which could successfully be read and opened by COOT. An example can be seen in Figure 4.14. The only problem left is the fact that x-ray experiments are based on crystals. Therefore, the electron density is periodic. The electron density of the ligand now has to be placed exactly at the same position as the ligand in the experimental structure.

4.6 Conclusions

This chapter describes the method and program developments of the present work. Most of the algorithms are implemented in the new Conformational Analysis and Search Tool (CAST). CAST provides access to several force fields as well as interfaces to semi-empirical and *ab initio* programs. By the implementation of various approaches for analysis, local optimization, transition state search, or global optimization, the program can be widely used. The Tabu-Search algorithm, implemented in CAST, was improved by the implementation of the Basin Hopping approach for diversification and the Dimer method for the modest ascent strategy. Thus, the performance was increased significantly and very complex optimization tasks can now be investigated. The Tabu-Search algorithm was used to establish a new

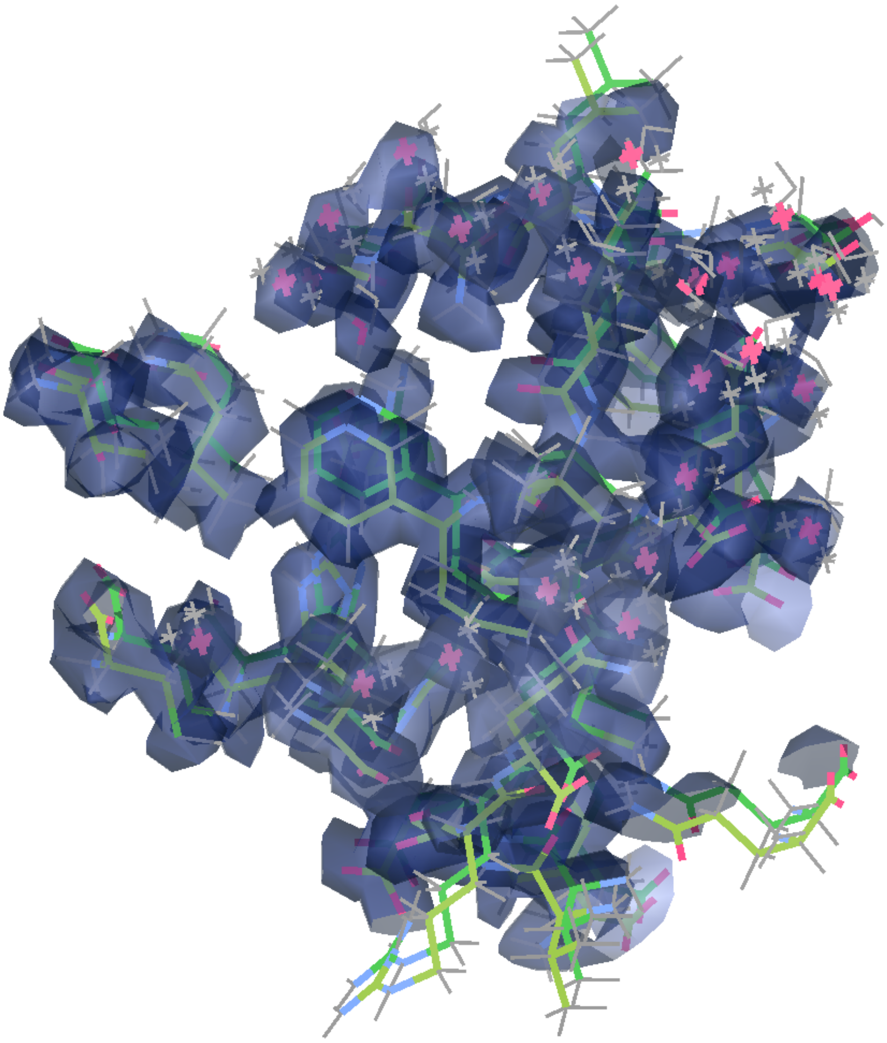


Figure 4.14: Calculated electron density converted into MAP format and visualized in COOT.

approach for the build up of solvent shells around a solute. The modest ascent strategy of Tabu-Search thereby allows for a step-wise solvation which enables a bias either to the (experimental) starting structure or the employed theoretical model. For investigations of reaction mechanisms of complex systems, a new approach called PathOpt was proposed. PathOpt searches the phase space between reactant and product for possible transition states which allows for the determination of different reaction pathways. Finally, a Graphical User Interface for the CAST program was presented, which was comprised of a small molecular viewer for the visualization of molecular orbital and electron density information.

5 Method application

5.1 Preliminary investigations: Optimization of Cluster systems

The global optimization of cluster systems is of paramount importance in computational chemistry. One example is the solvent shell around a molecule. Further examples are clusters of other solvent molecules or mixtures, or the optimization of aggregates and rare gas clusters.[7] Furthermore, many of these cluster systems are well investigated and comprise ideal benchmark systems to compare and evaluate the performance of new algorithms. Therefore, the performance of the Tabu-Search algorithms was investigated at hand of different Lennard-Jones (argon) clusters and water clusters.

5.1.1 Argon clusters

Lennard-Jones clusters, like for example clusters of the rare gas argon, are commonly used test systems for global optimization algorithms. An overview of the global minima of Lennard-Jones clusters of different sizes is given in 115 and the data base of J. P. K. Doye.[329] LJ-clusters usually posses very complex energy landscapes with many different minima and transition pathways.[279,330] One particular difficult example is the LJ_{38} cluster with a double-funnel energy landscape.[331] The first funnel ends in the global minimum while the second funnel ends in the second lowest minimum. Furthermore, the two minima are separated by a large energetic barrier, which explains the difficulty of the global optimization of this cluster. An optimization algorithm can easily be trapped in the second lowest minimum without finding the global one.
These aspects make LJ-clusters ideal benchmark candidates for global optimization algorithms as some minima are very hard to detect. To investigate the performance of the new Tabu-Search algorithm, clusters of different sizes have been investigated and optimized. The results can be found in Table 5.2. All global minima found with the Tabu-Search algorithm are in agreement with the literature.[115,329] This shows the strength of the new global optimization approach, since the optimization problem of LJ_{38} was also solved successfully. Figure 5.1 illustrates the three lowest minima of LJ_{38} obtained by the Tabu-Search optimization. The symmetry and order of these minima are in agreement with the minima suggested by Doye.[331] Note that the presented results are only meant for benchmarking the new algorithm. Therefore, structural aspects of the clusters are not discussed in detail.

5.1.2 Water clusters

Water, as the most important solvent in biological systems, plays a crucial role in many processes. More details are given in section 5.4. Therefore, it is essential to describe the structure of water clusters properly. To investigate the performance of the Tabu-Search algorithm for optimizing such cluster systems, different water clusters have been optimized. The

Table 5.1: Argon cluster optimized with the Tabu-Search algorithm.
[a] Total energy within the OPLS-AA force field
[b] Number of steps till first occurrence of global minimum

Cluster (Ar_x)	Symmetry	Energy [kcal/mol][a]	#steps[b]
4	T_d	-1.40340	1
6	O_h	-2.97335	23
8	C_s	-4.63625	158
10	C_{3v}	-6.64803	24
12	C_{5h}	-8.88062	3
13	I_h	-10.36804	97
20	C_{2v}	-18.05171	386
30	C_{2v}	-30.00623	24
38	O_h	-40.68186	300
40	C_s	-43.32994	35
60	C_s	-71.54427	242
80	C_s	-100.12875	561

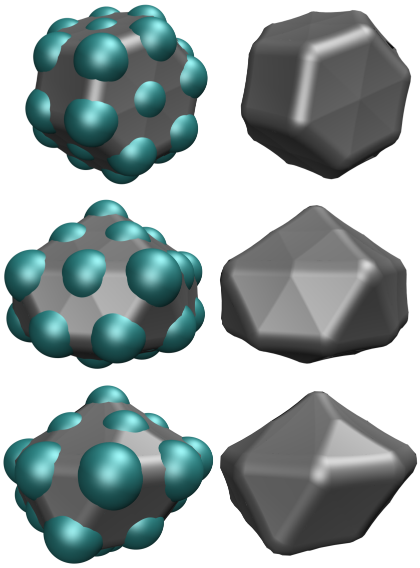


Figure 5.1: The three deepest minima of the Ar_{38} cluster, the deepest minimum is on top of the figure, the third deepest on the bottom. The global minimum is a fcc (face-centered cubic) truncated octahedron (point group O_h), the second and third deepest minima are incomplete Mackay icosahedra (Point group of 2.: C_{5v}, Point group of 3.: C_s).[331]

results are compared to optimizations with MCM proposed by Scheraga[120] implemented in Tinker 5.0. The searches were initialized with a random cluster configuration. Three water clusters have been investigated: $(H_2O)_{10}$, $(H_2O)_{20}$, and $(H_2O)_{50}$. The calculations have been performed with the OPLS-AA force field[232–235] and the TIP3P water potential[332] implemented in CAST.

The $(H_2O)_{10}$ cluster was further analyzed by global optimization with the TeraChem program using the interface described in section 4.1.4. The results for the force field optimizations are shown in Figure 5.2 and 5.3 which clearly show the better performance of Tabu-Search in comparison to MCM. For the Tabu-Search optimizations, 5000 main iterations have been used, while for MCM 10000 iterations were used. In each optimization run, the final results for $(H_2O)_{20}$ and $(H_2O)_{50}$ are lower in energy with Tabu-Search within a comparable computational time. For the $(H_2O)_{10}$, the Tabu-Search approach reliably delivers two of the most important arrangements.[333] MCM also located the global minimum structures, however, the second lowest structure was not found in these optimizations. It is often the case that not only the lowest structure, but all low lying configurations are important. This underlines an advantage of the Tabu-Search approach.

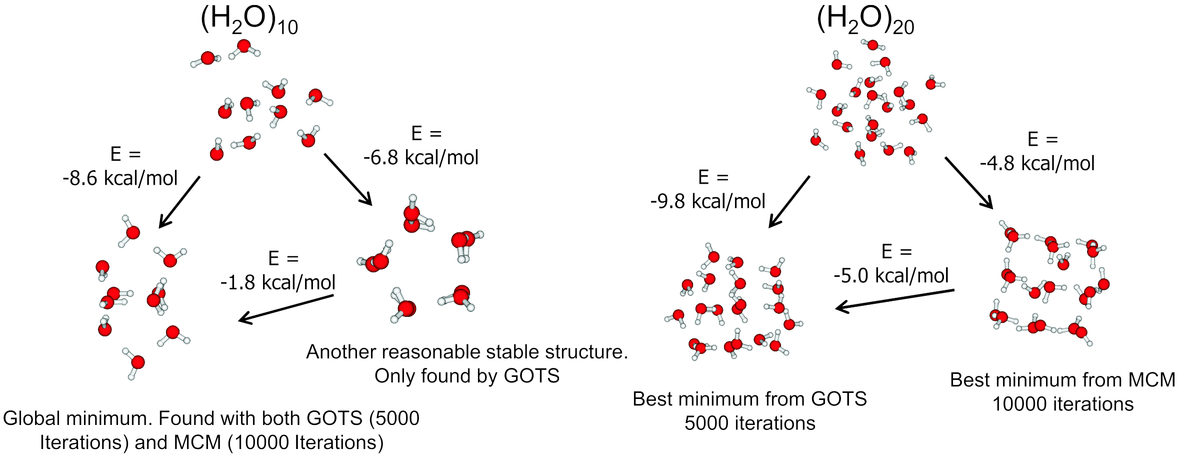

$(H_2O)_{10}$
E = -8.6 kcal/mol
E = -6.8 kcal/mol
E = -1.8 kcal/mol
Another reasonable stable structure. Only found by GOTS
Global minimum. Found with both GOTS (5000 Iterations) and MCM (10000 Iterations)
$(H_2O)_{20}$
E = -9.8 kcal/mol
E = -4.8 kcal/mol
E = -5.0 kcal/mol
Best minimum from GOTS 5000 iterations
Best minimum from MCM 10000 iterations

Figure 5.2: Results of the global optimizations of $(H_2O)_{10}$ and $(H_2O)_{20}$. The minimum higher in energy (right hand side of each cluster) is the one obtained by a MCM optimization with 10000 iteration.

Table 5.2: Results of the global optimizations of different water clusters with Tabu-Search and MCM

System	relative energy in kcal/mol	
	Tabu-Search	MCM
$(H_2O)_{10}$	0.0	0.0
$(H_2O)_{20}$	0.0	5.0
$(H_2O)_{50}$	0.0	6.7

The implemented interface to TeraChem further allows for a conformational search with *ab initio* calculations. Here, the difference between the OPLS-AA force field and B3LYP-D/6-31G++ is revealed, as the found minima possesses a different order (see Figure 5.11). More details are given in the next subsection.

The results of this section are mentioned to emphasize the applicability of Tabu-Search op-

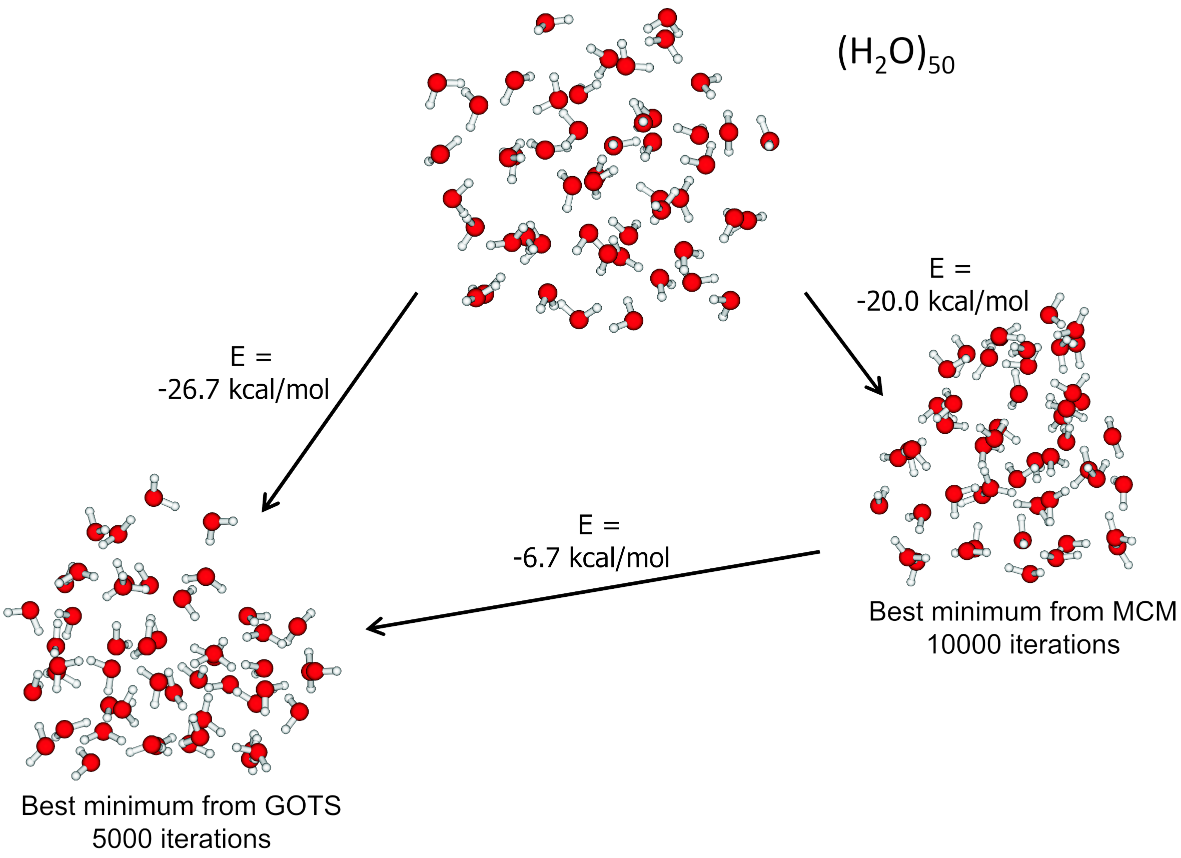

$(H_2O)_{50}$
E = -20.0 kcal/mol
E = -26.7 kcal/mol
E = -6.7 kcal/mol
Best minimum from MCM 10000 iterations
Best minimum from GOTS 5000 iterations

Figure 5.3: Results of the global optimizations of $(H_2O)_{50}$.

timization to cluster optimization and should clarify the performance as global optimization algorithm. Therefore, the results are not discussed in more detail.

5.2 Conformational Search

5.2.1 Efficiency of Tabu-Search based conformational search

Motivation The results of the following subsection are published in ref. 52. The aim of the study was the investigation of the usability of Tabu-Search based global optimization algorithms for conformational search (CS). The Gradient Only Tabu Search (GOTS) developed by Svetlana Stepanenko[154] had already been applied to conformational search problems,[155] but an extensive comparison to other CS-algorithms was lacking. Furthermore, the strengths and weaknesses of GOTS are discussed and modifications to the GOTS approach are introduced which dramatically improve the performance.

Computational details The original GOTS algorithm was applied to the conformational search of a set of test molecules (see Figure 5.4). The results were compared to conformational searches using Molecular Dynamics with minimization (MDM), Simulated Annealing (SA) and Monte Carlo with Minimization (MCM) also known as Basin Hopping (BH). All calculations were performed using the OPLS-AA force field as implemented in Tinker

5.0[37,250,305–308] using a truncated Newton-like optimization algorithm[305] for geometry optimization. For MD, SA, and MCM calculations the implemented algorithms of Tinker 5.0[37,250,305–308] were used. To be comparable to these optimization algorithms, the Tinker program packages was implemented into the GOTS approach (see 4.1 for details). Each calculation started from 30 different starting structures which were taken randomly from an a priori MD simulations (1 ns simulation time (NVT ensemble), 1 fs time step, 100 snap shots).

MD simulations were performed for 1 ns (NVT ensemble) with a 1 fs time step (1000000 steps in total). Every 10 ps, a snapshot was taken which was subsequently minimized. No heating or cooling protocol was applied, as the search space for the test molecules is relatively small.

The SA simulations implicitly include the heating and cooling procedures. The initial temperature was set to 1000 K. The equilibration was performed for 100 steps which was followed by cooling to 0 K using a linear decrease in the temperature with a factor (current step number)/(total number of steps). Every 10 ps (i.e. 10000 steps), a snapshot was taken which was subsequently minimized.

The MCM of BH simulations were performed using the implementation of Tinker 5.0. Each global optimization with MCM comprises 5000 MCM iterations and a maximal step size of 3.0 Å.

As already mentioned, the GOTS algorithm was combined with the Tinker program package to make the calculations comparable. Each GOTS calculations included 1000 *steepest descent* - *modest ascent* steps. When the search gets stuck in a given region, a Diversification Search (DS) was initialized. For more detail see section 4.

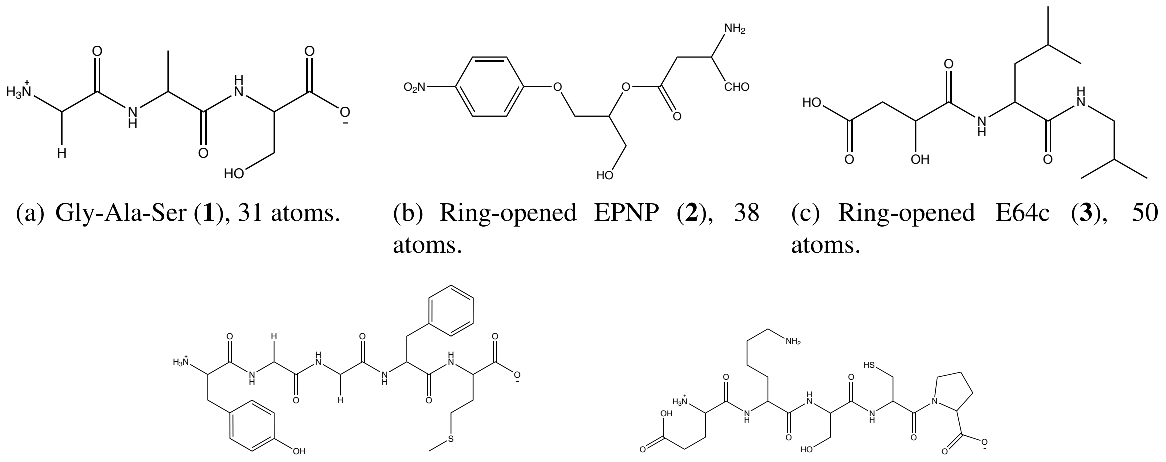
The first results revealed that the BH approach often performed better than the GOTS algorithm. Therefore, BH was used to improve the DS of GOTS, i.e. BH was implemented into the DS of GOTS to create a new Tabu-Search algorithm, named GOTS/BH. Each time a DS is initialized a short BH simulation (e.g. 200 steps) is performed. Details of the implementations can be found 4.2. For the present study, the BH search was performed in Cartesian coordinates with a maximal step size of 3.0 Å. The step size has to be adapted to the system under investigation. For bigger systems, a smaller step size might be favorable.

The efficiency of many CS-algorithms extremely depends on the starting structure used. Therefore, the StartOpt/RingSearch algorithm developed during my diploma thesis[46] was combined with the above mentioned CS-algorithms. StartOpt/RingSearch searches a given molecule for possible intra-molecular ring conformations build up through hydrogen bond donors and acceptors. A second series of calculations with the abbreviation StartOpt was performed. For each molecule and search algorithm, the StartOpt/RingSearch algorithm was applied once and the obtained structures were used as starting structures.

The last series included starting structure containing several intra-molecular ring conformations. These were obtained by subsequent application of StartOpt/RingSearch until no new

structure were found any more (up to three iterations). For the bigger test molecules ((**4**) and (**5**)), the number of ring structures was too large. Therefore, only the energetically best structures were used as new starting point of StartOpt/RingSearch.
It should also be mentioned, that the calculations within this study were performed with an early version of GOTS. In the meantime, several improvements concerning performance as well as search strategies are included (see section 4 for the implementation into CAST) and the computational times can be different now. Nevertheless, the results are presented as given in the publication.

Test systems Five molecules were taken as test systems. Three peptide systems (tripeptide Gly-Ala-Ser (**1**), pentapeptide Glu-Lys-Ser-Cys-Pro (**5**) and [Met^5]enkephalin (**4**)) and two organic ligands with biological activities (Ring-opened EPNP (**2**) and ring-opened E64c (**3**)). The size of the systems range from 31 atoms to 76 atoms. Structural formulas can be found in Figure 5.4. Of course, various other molecules were investigated using the Tabu-Search algorithm implemented in CAST. All of them are pointing to the high efficiency of Tabu-Search based conformational search. Nevertheless, the systems given above are taken as examples to demonstrate the performance of the new Tabu-Search-Basin-Hopping approach. The test set is well suited as a benchmark system because the molecules are typical bio-organic systems with many functional groups, like often investigated in molecular modeling.



(a) Gly-Ala-Ser (**1**), 31 atoms.

(b) Ring-opened EPNP (**2**), 38 atoms.

(c) Ring-opened E64c (**3**), 50 atoms.

(d) Met^5-enkephalin (**4**), 75 atoms.

(e) Glu-Lys-Ser-Cys-Pro (**5**), 76 atoms.

Figure 5.4: Test systems used for comparing the efficiency of Tabu-Search based conformational search to commonly used algorithms.

Results and Discussion Table 5.3 to 5.4 and A.1 to A.3 give the computational results for the simulations of the five test molecules. The results for the smallest molecule (**1**) in Table 5.3 show that the system is too small for acting as a general benchmark system. With only 31 atoms and 11 freely rotatable bonds (two of them are more or less rigid amide bonds), the system is relatively small. Looking at the results for the conformational searches, only MD and SA do not locate the global minimum (GM) with -168.4 kcal mol^{-1}. Nevertheless, the final results found by MD and SA is only 0.9 kcal mol^{-1} above the GM. Also the efficiency in locating the GM is already quite good for all methods. The use of starting structures provided by StartOpt/RingSearch do not alter the final minima but affects the performance obviously.

For MD and SA, the performance seems to be increased. For example, the percentage in locating the final minimum in MD is increased from 17% to 75% and the computational time is decreased from 1.2 to 0.1 CPU minutes. However, for GOTS and BH methods, the performance is decreased when starting from structures containing one single or two ring conformations. This is due to the inclusion of higher lying structures which are not good starting structures. Table 5.3 makes clear, that only the combination of GOTS and BH gives the best performance. When short BH simulations are used within the DS of GOTS, the efficiency in locating the GM is increased to 100% for this case, no matter which starting structure was used.

Table 5.3: Results for the tripeptide (**1**) containing 31 atoms. [a]Relative energy of the energetically lowest minim found in the given simulation with respect to the lowest minimum found in all simulations (E = -168.4 kcal mol^{-1}). Energies are given in kcal mol^{-1}. [b]Percentage of simulations runs which found the minimum depicted in column one. [c]Average number of steps (MCM and GOTS) or snap shot (MD and SA) needed to find the minimum in column one the first time. The averaging is only done for runs where the minimum was found. [d]Corresponding averaged CPU time in minutes.

Optimization method	E^{a}_{min}	#global[b] (%)	#steps[c]	CPU time[d]
MD	0.9	17	59	1.2
SA	0.9	13	10	0.2
BH	0.0	83	1613	1.2
GOTS	0.0	80	322	0.3
GOTS/BH	0.0	100	58	0.1
MD-StartOpt	0.9	75	3	0.1
SA-StartOpt	0.0	80	9	0.2
BH-StartOpt	0.0	60	1877	1.5
GOTS-StartOpt	0.0	55	250	0.2
GOTS-StartOpt/Mult	0.0	53	164	0.1
GOTS/BH-StartOpt/Mult	0.0	100	258	0.4

The other two smaller molecules (ring-opened (**2**) and (**3**), Table A.1 and A.2) show a similar trend. For (**2**), BH simulations are finding the GM in each run. The use of structures containing single ring structures again slightly increase the performance for MD and SA. However, these structures decrease the efficiency of GOTS. Also structures containing several ring conformations have poor effect. Only the combination of GOTS and BH improves the GOTS approach. (**2**) seems to represent a very easy conformational search problem. Only the pure

MD simulation does not locate the GM. All other simulations find the GM in different yields. The use of StartOpt/RingSearch structures does not give a significant improvement.

The results for the small systems (**1**), (**2**), and (**3**) outline the efficiency of Tabu-Search based global optimization. However, they also show that structures provided by StartOpt/RingSearch does not necessarily improve the performance for these systems as the search space is too small and the provided starting structures may include too many poor starting points for global optimization.

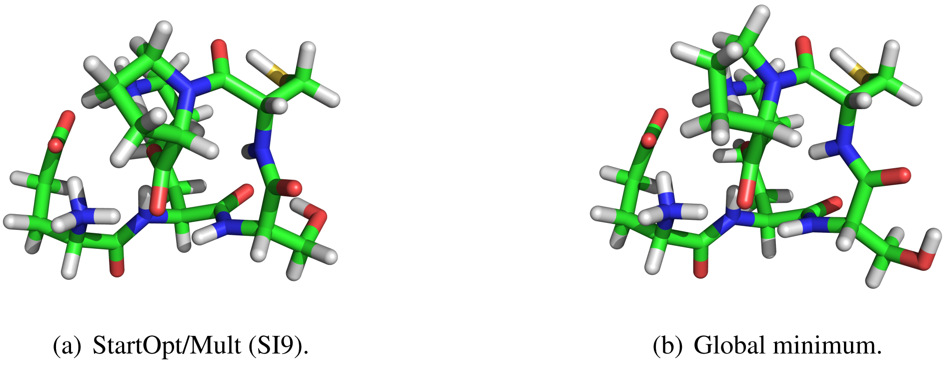
The pentapeptide (**5**), however, represents a very good test system for conformational search. Table 5.4 shows the results for this molecule. The molecule consists of 25 freely rotatable bonds (including four amide bonds). A pure MD initiated from a random MD-generated structure completely fails in locating the GM ($\sim$18 kcal mol^{-1} above the GM) indicating that the search space becomes too large for the very simply MD approach. The considerably lower minimum located by SA is still $\sim$ 6 kcal mol^{-1} above the GM. BH performs quite good by locating a minimum only 0.7 kcal mol^{-1} above the GM in 6% of the runs. The GOTS, in comparison, had a slightly worse performance. Again, the combinations of GOTS and BH outperform both single algorithms. It is the only approach which locates the GM (in 37% of the runs).

The use of StartOpt/RingSearch structures improves the performance of all used algorithms. Hence, in this case these structures seems to be helpful. For (**5**), the number of possible ring structures is too big to generate all combinations of three closed rings. Therefore, only the 10 energetically lowest conformations of the first generations were used to produce the second and the 10 best structures from the second generation to yield the final structures containing three intra-molecular ring structures. The 36 final structures lie between 7 and 23 kcal mol^{-1} above the GM. The seven best conformers are characterized in more detail in Table 5.5. The RMSD values for torsional angles are calculated using the formula given by Becker *et al.*[50]:

$$d_{ij} = \frac{1}{N} \sum_{k=1}^{N} \min \left[\left(\Theta_k^{(i)} - \Theta_k^{(j)} \right)^2, \left(2\pi - \Theta_k^{(i)} - \Theta_k^{(j)} \right)^2 \right] \tag{5.1}$$

The table also points out, from which starting structure the GM or the slightly above lying minimum is located.

Using these structures as starting structures for BH, GOTS, and GOTS/BH shows a significant improvement in the performance of the Tabu-Search methods. BH itself does not alter its performance to much. It only needs fewer steps until reaching its best results (0.7 kcal mol^1 above the GM). The normal GOTS, however, finds the GM in 9% of the runs when a StartOpt/Mult structure is used. The much more efficient GOTS/BH approach locates the GM in 68% of all runs when originated from a structure containing three intra-molecular ring conformations. The difference in the influence of the starting structures between pure BH and GOTS or GOTS/BH becomes clear when the search strategies are compared. BH is a purely stochastic algorithm. A better starting structure only introduces a deeper lower limit for the Metropolis criteria. In contrast, GOTS follows paths on the potential energy surface. Therefore, a structure closer to the GM can indeed accelerate the performance. The best structure obtained by threefold StartOpt application (SI9) as well as the global minimum are shown in Figure 5.5.



(a) StartOpt/Mult (SI9).

(b) Global minimum.

Figure 5.5: Structures for molecule (**5**).
5.5-a: Best structure obtained by threefold application of StartOpt (SI9, see Table 5.5).
5.5-b: Global minimum structure.

Unfortunately, the utility of the starting structure does not correlate with the potential energy or the RMSD values. This was also verified within the research project of Anastasia Weickert under my supervision,[334] where the influence of ring conformation on the global optimization of three polypeptides was examined exhaustively by creating a database containing all possible ring conformations with up to three closed rings. For example, when looking at a system with three close ring structures, the first ring closure can stabilize the system. However, it often appeared that the second ring closure destabilized the system before a dramatical stabilization by the third ring closure occurred. Therefore, up to now no systematic approach is available to create the most reasonable ring systems.

The neurotransmitter [Met^5]-enkephalin (molecule (**4**)) is only slightly smaller than (**5**) and was also used in many other investigations[45,120,173,195]. However, the results summarized in Table A.3 indicated that molecule (**4**) represents a much easier conformational search problem than molecule (**5**). Starting from random MD-generated structures, again MD and SA fails in locating the GM. BH and GOTS/BH performed very well in locating the GM in 57% and 87% of the simulation runs. Even the pure GOTS approach already located the GM, however, the GM was only located once. Like for the smaller molecules (**1**) to (**3**), the use of StartOpt/RingSearch structures slightly decreases the performance and success quota. This might be due to the steric hindrance of the aromatic residues Tyr-1 and Phe-4 leading to more unfavorable ring conformations. Furthermore, the global minimum structure only contains one ring conformation. All other conformations might lead to a destabilization.

Table 5.4: Results for the pentapeptide (**5**) containing 76 atoms. [a]Relative energy of the energetically lowest minim found in the given simulation with respect to the lowest minimum found in all simulations (E = -376.3 kcal mol^{-1}). Energies are given in kcal mol^{-1}. [b]Percentage of simulations runs which found the minimum depicted in column one. [c]Average number of steps (MCM and GOTS) or snap shot (MD and SA) needed to find the minimum in column one the first time. The averaging is only done for runs where the minimum was found. [d]Corresponding averaged CPU time in minutes.

Optimization method	E^a_{min}	#global[b] (%)	#steps[c]	CPU time[d]
MD	17.8	-	44	4.8
SA	6.2	-	46	5.1
BH	0.7	6	3140	13.2
GOTS	1.2	-	955	2.2
GOTS/BH	0.0	37	632	4.5
MD-StartOpt	4.7	-	64	7.0
SA-StartOpt	5.8	5	57	6.3
BH-StartOpt	0.7	13	2347	9.9
GOTS-StartOpt	0.7	-	694	1.6
BH-StartOpt/Mult	0.7	-	436	2.0
GOTS-StartOpt/Mult	0.7	8	128	0.3
	0.0	9	184	0.4
GOTS/BH-StartOpt/Mult	0.0	68	472	3.4

Table 5.5: Characterization of conformers of (**5**), which were obtained after threefold application of StartOpt/RingSearch. [a]RMSD value giving the difference between torsional angles. [b]Relative energy of the structure with respect to the GM (-376.3 kcal mol^{-1}). [c]Percentage of runs of the simulation which found the minimum laying 0.7 kcal mol^{-1} above the GM. [d]Percentage of runs of the simulation which found the GM.

			#semi[c]	#global[d]	
Structure	RMSD[a]	E_{min} [b]	GOTS	GOTS	GOTS/BH
F15	43.7	11.8	17	17	90
F5	54.0	15.1	-	-	40
F7	39.7	11.4	-	-	80
SA2	36.1	11.4	-	-	90
SA5	53.1	10.3	-	-	10
SB1	55.9	11.8	11	-	100
SI9	36.0	6.4	32	32	80
Total			9	7	68

Conclusions The results presented above clearly show the applicability of Tabu-Search based algorithms for conformational search. The comparison to MD, SA, and BH indicated, that GOTS is much more efficient than MD and SA. However, BH itself often performs better than GOTS. Therefore, short BH sequences were implemented into the DS of GOTS which proved to be very useful. The application of the new GOTS/BH approach to the test molecules (**1**) to (**5**) outperforms each of the single methods. The study also revealed the need for a careful check of the stereo centers, since BH simulations can easily switch them, especially when force field calculations are used. In newer versions of the Tabu-Search a subroutine is implemented for an automatic stereo checking, restricting the conformational search in switching a stereo center. This subroutine was not included in the study presented above.

The usefulness of five, six, or seven-membered ring structures produced by StartOpt/Ring-Search was shown for larger molecules, especially when structures containing multiple ring conformations are used.

The presented conformational search studies are only a small excerpt of possible applications. Improvements to the Conformational Analysis and Search Tool (CAST) and the Tabu-Search algorithm (see section 4 and following sections) make the new Tabu-Search algorithm applicable to much larger systems. Examples of further applications are shown below.

5.2.2 Conformational search of peptide ligand-receptor systems

Motivation This work is submitted to publication.[335] The work was done in cooperation with Stefan Niebling from the group of Sebastian Schlücker, University of Osnabrück.

Conformational search using force field methods on complex biomolecular systems is a key factor in understanding molecular and structural properties. The reliability of such investigations strongly depend on the efficiency of the conformational search algorithm as well as the accuracy of the employed force field. Therefore, in the following section two different approaches are compared; the Monte-Carlo multiple minimum/low mode sampling (MCMM/LM) using the OPLS2005 and Tabu-Search combined with Basin Hopping (TS/BH) employing an OPLS-AA implementation. Thus, their performance in locating energetically low lying structures and the efficiency in scanning the conformational phase space of non-covalently bonded complexes were investigated. As test systems, complexes of the artificial peptide receptor CBS-KKF with different tetra-peptide ligands were taken. The reliability and the accuracy of the two approaches were examined by re-optimizing all low-energy structures employing density functional theory with empirical dispersion correction in combination with triple zeta basis sets. Solvent effects were mimicked by a continuum solvent-model. In all four test systems, the TS/BH approach yielded structures which were much lower in energy after DFT optimization. Additionally, TS/BH provided many low lying structures which are not located by MCMM/LM.

Computational Details The MCMM/LM method implemented in MacroModel V9.9[133] consists of a combination of the Monte Carlo Multiple Minima (MCMM) algorithm in internal coordinates[27] and the Low-Mode search method (LMOD).[40,41,134] The LMOD algorithm is a simplification ("brute force" approach) of the mode-following concept.[40] Instead of locating a saddle point exactly, the method starts with a low mode eigenvector of the

Hessian matrix and follows this direction linearly. The search algorithm was combined with the OPLS2005 force field. The standard settings from MacroModel were applied.
All Tabu-Search calculations were performed with the CAST program described in this work. The steepest descent part was implemented by a L-BFGS algorithm[271–275] and for the modest ascent part a specially adapted Dimer method[198–200] was employed (section 4.2). For local optimizations, a 0.004 kJ mol^{-1} Å^{-1} euclidean norm of the gradient was used. To ensure that stereo centers were not inverted, all possible stereo centers were recognized at the beginning of the search and all moves inverting a center were rejected. The diversification search was implemented by short BH sequences with a maximum step size of 3.0 Å. In the following section, the combination of Tabu-Search and Basin Hopping is referred to as TS/BH. For each Tabu-Search simulation 3000 Steepest Descent - Modest Ascent iterations were employed. The TS/BH searches were performed in combination with the original OPLS-AA force field proposed by Jorgensen.[233,336,337] Non-bonded interactions are calculated without a cutoff radius.
All DFT calculations were performed with TurboMole (Version 6.3.1)[338] using the BP86 functional in combination with def2-TZVP[339,340] basis sets employing the RI approximation[341] and the D3 dispersion correction of Grimme.[342] Solvent effects were included by the COSMO model.[343]
MCMM/LM calculations with the MacroModel program and DFT calculations with TurboMole were performed by Stefan Niebling from the University of Osnabrück.

Test systems As a model system, non-covalent complexes between an artificial peptide receptor (CBS-KKF) and four different tetra-peptides ligands have been chosen (Figure 5.6). The peptide receptor CBS-KKF designed by Schmuck and coworkers is based on a guanidiniocarbonyl pyrrole,[344] which showed remarkably high binding constants in water, and a pronounced sequence- and diastereoselectivity.[345] It comprises of two parts. Firstly, the carboxylate binding site (CBS), a guanidiniocarbonyl pyrrole that efficiently binds carboxylates by a combination of electrostatic interactions and hydrogen bonding. Secondly, a tripeptide part (here L-Lys-L-Lys-L-Phe-NH_2, KKF), which further strengthens the complex by hydrogen bonds and electrostatic interactions. The tri-peptide part is responsible for the selectivity of the receptor towards a particular peptide. Due to its positively charged groups (CBS and two lysine side chains), the receptor CBS-KKF favors peptides which contain negatively charged residues.
The four peptide ligands include the diastereomer pair N-Ac-(D-Glu)$_3$-D-Ala (e$_3$a, ligand 200, (**6**)) and N-Ac-(D-Glu)$_3$-L-Ala (e$_3$A, ligand 201, (**7**)) and the tetra-peptides N-Ac-L-Ala-(D-Glu)$_3$ (Ae$_3$, ligand 202, (**8**)) and N-Ac-D-Ala-L-Ala-(D-Glu)$_2$ (aAe$_2$, ligand 203, (**9**)).[345] These systems are experimentally well characterized,[345,346] but theoretical studies about their structural properties are scarce.[347] All receptor/ligand complexes are ideal test systems since they are conformationally so flexible that a complete exploration of the

conformational space is impossible.

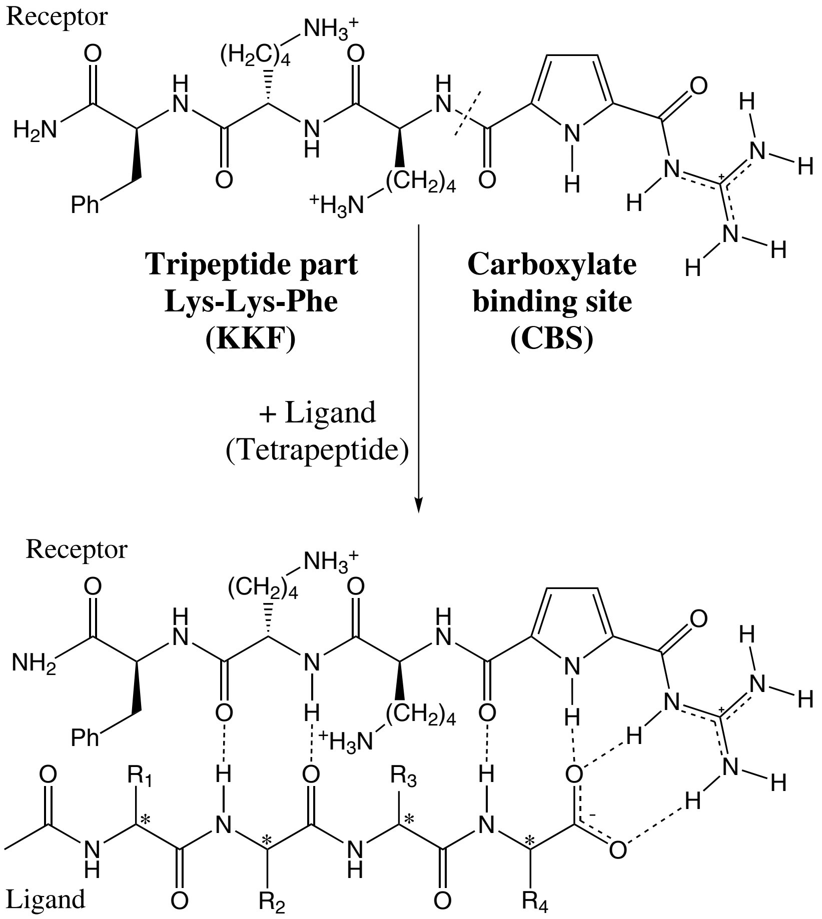
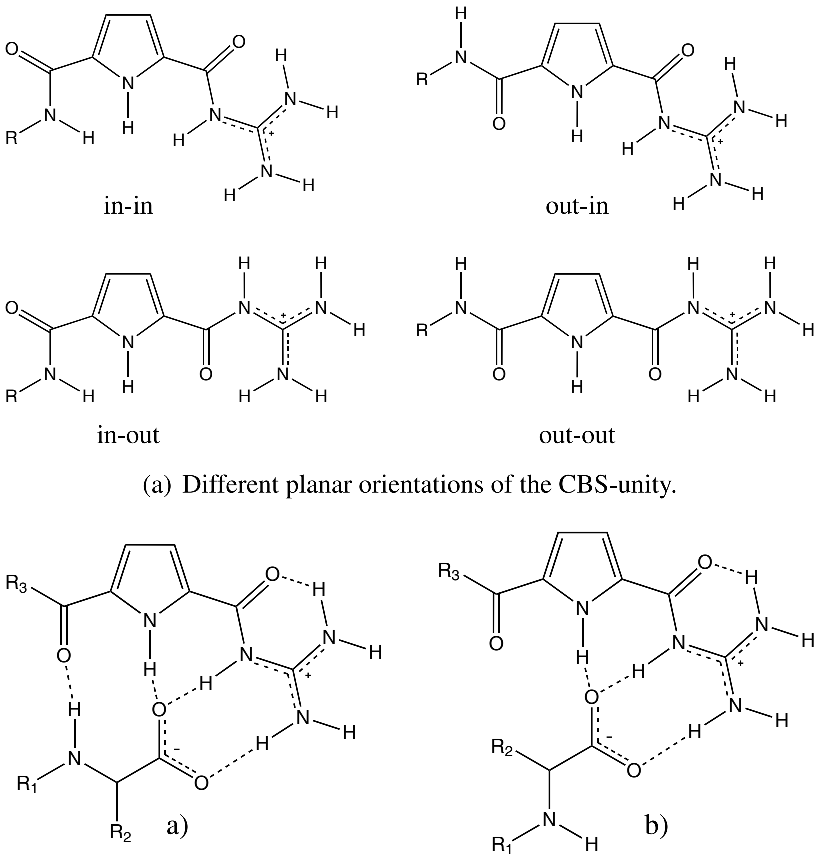


Figure 5.6: Ligand receptor complexes formed by CBS-KKF (top) and a tetra-peptide (bottom).

The receptor and the ligand have several different binding sites and orientations. To specify the structures the nomenclature introduced by Moiani et al.[347] was used. The four possible configurations at the CBS-unity of the CBS-KKF receptor result from a coplanar orientation of the pyrrole ring relative to the guanidiniocarbonyl and the amide which is adjacent to the pyrrole ring. The possible conformations are shown in Figure 5.7-a. If the respective NH is pointing in the same direction as the pyrrole-NH, it is referred to as "in" otherwise as "out". The labeling as well as deviations from planar arrangements are characterized with the help of the three dihedral angles in Figure 5.8. Low lying non-planar structures result from favorable interactions between the ligand carboxylates and the CBS unity and the lysine side chains of the receptor.

Beside the carboxylate group at the C-terminus (CT), each ligand possesses additional ones at the side chains (SC). For each, two planar orientations are possible, either with the residue pointing in direction of the receptor (a in Figure 5.7-b) or opposite to the receptor (b in Figure 5.7-b). The SC carboxylates are numbered from 1 to 4 (SC1 - SC4) according to the amino acid number of the tetra-peptide.

In order to cover a larger part of the conformational space, different starting structures were used for each receptor/ligand complex. The strongest single interaction between the CBS-KKF and the tetra-peptide ligands resulted from the salt bridge between the CBS and one of the carboxylate groups of the ligand. To account for the importance of these interactions all possible pairs were used as starting structures. The name of the starting structure consists of the investigated ligand (200 to 203), the carboxylate group which binds to the CBS-units (e.g. SC4) and the orientation of the carboxylate with respect to the receptor (a or b, see Figure 5.7-b). If more than one starting structure was generated for a given classification, they were numbered consecutively (e.g. 202_SC4a and 202_SC4a2). All starting structures are derived from an out-in conformation of the CBS and an all-trans conformation of the peptide backbone of the tri-peptide subunit.

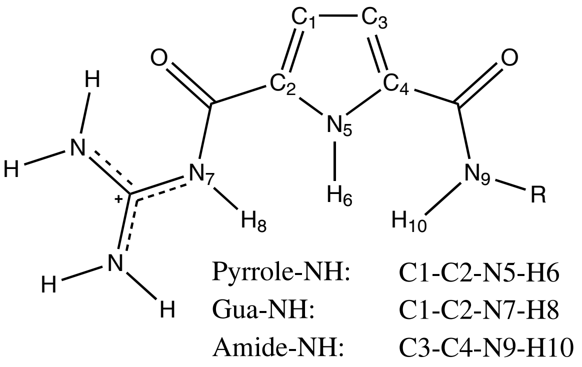


(a) Different planar orientations of the CBS-unity.

(b) Possible planar orientations of the carboxylate group.

Figure 5.7: Different possible conformations of the CBS-unity and the carboxylate group taken as starting structures.

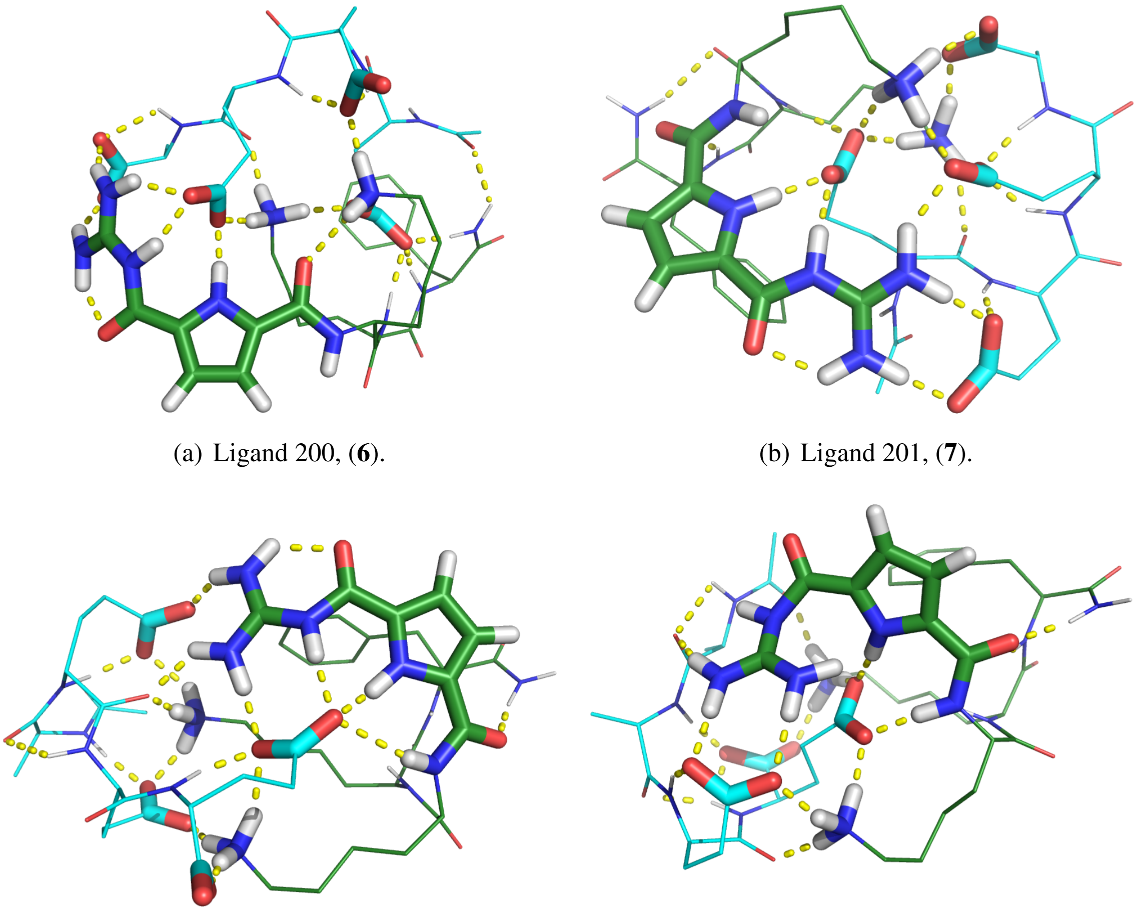
Results and discussion The results of the different conformational search approaches and DFT re-optimizations are summarized in Table 5.6 to 5.9. Relative energies based on force field and DFT computations (after geometry optimization) are included. Relative force field energies are given relative to the best result of MCMM/LM and TS/BH, respectively. DFT energies are given with respect to the lowest lying conformation from all searches. Based



Pyrrole-NH:	C1-C2-N5-H6
Gua-NH:	C1-C2-N7-H8
Amide-NH:	C3-C4-N9-H10

Figure 5.8: Atoms defining the dihedral angles used to specify the conformation according to Moiani. The atom numbers are only added for clarity and are different in each system. A dihedral angle of Gua-NH or Amide-NH between -180° and -90° or 90° and 180° is characterized as "in", an angle between -90° and 90° is referred to as an "out" conformation. The Pyrrole-NH illustrates the planarity of the pyrrole group.

on the DFT energy, in the following the lowest energy structures for each complex will be discussed (see Figure 5.9) .



(a) Ligand 200, (**6**).

(b) Ligand 201, (**7**).

(c) Ligand 202, (**8**).

(d) Ligand 203, (**9**).

Figure 5.9: Lowest structures found after re-optimization with DFT for each complex of the investigated ligands and the receptor CBS-KKF.

Table 5.6: Summary of the final structures computed for the complex of the CBS-KKF and the peptide e$_3$A (ligand 201). The results obtained with MCMM/LM-OPLS2005 are given on the left side while the findings received with TS/BH-OPLS-AA are shown on the right hand side. Relative force field energies are given relative to the best result of MCMM/LM and TS/BH, respectively. DFT energies are given with respect to the lowest lying conformation from all searches. [a] Name of ligand carboxylates involved in complexation, for more information see text. [b] Relative energy with respect to the lowest lying structure within the corresponding force field. Values in parentheses are re-optimized energies within the OPLS2005 (MacroModel) force field and given relative to the lowest structure of MCMM/LM. [c] Relative energy with respect to the lowest lying structure after re-optimization of the force field structures with BP86-D3/COSMO/def2-TZVP. [d] Dihedral angles Φ of the classification according to Moiani, illustration see Figure 5.8. * Conformations which which are not covered by Moiani (Figure 5.7-a). In these conformations the carbonyl and the adjacent NH-group are pointing in the same direction.

	MCMM/LM-OPLS2005						TS/BH-OPLS-AA					
				[a]Binding mode						[a]Binding mode		
Start	[b]ΔE^{MM} $[kJ/mol]$	[c]ΔE^{MM} $[kJ/mol]$	[a]Carb.	[d]Φ (Pyrrole)	[d]Φ (Amide)	[d]Φ (Gua)	[b]ΔE^{MM} $[kJ/mol]$	[c]ΔE^{MM} $[kJ/mol]$	[a]Carb.	[d]Φ (Pyrrole)	[d]Φ (Amide)	[d]Φ (Gua)
201_CTa	24	89	SC3	173	13 (out)	-170 (in)	32 (-20)	11	SC1, SC2	178	-6 (out)	-158 (in)
201_CTb	12	69	SC2,SC3	-178	132 (in)	151 (in)	0 (-11)	32	SC2, SC3, CT	-159	161 (in)	-142 (in)
201_CTb2	39	144	SC1,CT	-178	-3 (out)	177 (in)	32 (-20)	11	SC1, SC2	178	-7 (out)	-158 (in)
201_SC1a	30	62	SC1	168	-117 (in)	168 (in)	11 (-37)	2	SC1, CT	-179	147 (in)	103 (in)
201_SC1b	29	71	SC1	-177	-4 (out)	147 (in)	15 (-28)	0	SC1, SC2, SC3	-174	133 (in)	-163 (in)
201_SC2a	20	53	SC2	-178	-24 (out)	164 (in)	30 (-34)	9	SC2, SC3	-172	125 (in)	145 (in)
201_SC2b	14	24	SC2	-176	-3 (out)	-179 (in)	32 (19)	80	SC2, SC3	173	148 (in)	122 (in*)
201_SC3a	85	83	SC3	-173	25 (out)	-170 (in)	32 (-20)	19	SC1, SC2	164	52 (out*)	119 (in)
201_SC3b	0	14	CT	-177	8 (out)	-176 (in)	22 (-10)	82	SC3, CT	178	59 (out*)	-149 (in)

Table 5.7: Summary of the final structures computed for the complex of the CBS-KKF and the peptide e$_3$a (ligand 200). For more detail see Table 5.6.

	MCMM/LM-OPLS2005						TS/BH-OPLS-AA					
				^{a}Binding mode						^{a}Binding mode		
Start	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)
200_CTa	12	48	CT	-178	-16 (out)	163 (in)	24 (6)	29	SC3, CT	-174	134 (in)	153 (in)
200_CTb	31	55	SC3	-171	-8 (out)	-174 (in)	7 (-33)	7	SC1, SC3	174	135 (in)	-160 (in)
200_SC1a	84	149	SC1	174	11 (out)	-172 (in)	7 (-33)	7	SC1, SC3	174	137 (in)	-160 (in)
200_SC1b	52	125	SC1	180	143 (in)	-158 (in)	0 (37)	2	SC1, SC3, CT	167	165 (in)	112 (in)
200_SC2a	5	47	SC2, SC3	-173	11 (out)	174 (in)	9 (-18)	0	SC3, CT	166	13 (out)	124 (in)
200_SC2b	0	42	SC2	-169	-20 (out)	163 (in)	40 (28)	10	SC1, SC2	176	131 (in)	-166 (in)
200_SC3a	14	83	SC3, CT	-174	-134 (in)	-146 (in)	31 (24)	34	SC3, CT	172	-4 (out)	153 (in)
200_SC3b	54	100	SC1, SC3	178	-10 (out)	-156 (in)	7 (-33)	6	SC1, SC3	175	138 (in)	-163 (in)

Table 5.8: Summary of the final structures computed for the complex of the CBS-KKF and the peptide Ae_3 (ligand 202). For more detail see Table 5.6.
†The DFT geometry optimization of MCMM/LM structure 202_SC3b did not convergence (dE/dxyz=8e-3; criterion: 1e-3) after 400 cycles.

	MCMM/LM-OPLS2005						TS/BH-OPLS-AA					
				^{a}Binding mode						^{a}Binding mode		
Start	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)
202_CTa	75	115	CT	-180	-3 (out)	-177 (in)	16 (44)	68	SC3, CT	-164	-174 (in)	-114 (in)
202_CTb	89	122	CT	-177	-3 (out)	-175 (in)	13 (98)	45	SC2,SC3,SC4,CT	-171	111 (in)	116 (in)
202_SC2a	48	52	SC2	179	-4 (out)	174 (in)	11 (23)	19	SC2, SC3	-178	-4 (out)	172 (in)
202_SC2b	0	29	SC2	178	-29 (out)	165 (in)	0 (14)	0	SC2, SC4	172	140 (in)	-163 (in)
202_SC3a	45	66	SC3	173	-26 (out)	-176 (in)	7 (45)	53	SC2, SC3	-177	179 (in)	161 (in)
202_SC3b	104	148†	SC3	-172	-7 (out)	160 (in)	12 (79)	55	SC3, CT	176	59 (out*)	175 (in)
202_SC4a	84	133	CT	178	142 (in)	154 (in)	7 (38)	34	SC2, SC3, CT	178	153 (in)	146 (in)
202_SC4a2	74	90	SC4	179	-25 (out)	165 (in)	14 (107)	53	SC2, SC3	179	-41 (out)	-37 (out)
202_SC4b	48	34	CT	178	-13 (out)	178 (in)	8 (12)	48	SC2, SC3	178	-2 (out)	-161 (in)
202_SC4b2	20	64	CT	-179	-20 (out)	-174 (in)	8 (12)	47	SC2, SC3	178	-3 (out)	-161 (in)

Table 5.9: Summary of the final structures computed for the complex of the CBS-KKF and the peptide aAe$_2$ (ligand 203). For more detail see Table 5.6.

	MCMM/LM-OPLS2005						TS/BH-OPLS-AA					
				^{a}Binding mode						^{a}Binding mode		
Start	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)	$^b\Delta E^{MM}$ $[kJ/mol]$	$^c\Delta E^{MM}$ $[kJ/mol]$	^{a}Carb.	$^d\Phi$ (Pyrrole)	$^d\Phi$ (Amide)	$^d\Phi$ (Gua)
203_CTa	17	58	CT	-174	-18 (out)	153 (in)	33 (41)	54	SC3, CT	172	5 (out)	-168 (in)
203_CTb	15	50	CT	-179	-16 (out)	164 (in)	31 (58)	90	SC3, CT	177	49 (out*)	-148 (in)
203_SC3a	28	104	SC3	168	42 (out)	-148 (in)	0 (35)	0	SC3, SC4	-153	142 (in)	64 (out*)
203_SC3b	0	46	SC3	-175	-36 (out)	163 (in)	24 (24)	20	SC3, SC4, CT	-167	-179 (in)	-126 (in)
203_SC4a	17	42	CT	177	7 (out)	178 (in)	26 (53)	57	SC3	175	1 (out)	145 (in)
203_SC4a2	67	142	SC4	-174	-34 (out)	175 (in)	29 (28)	45	SC4	176	2 (out)	-159 (in)
203_SC4b	35	53	SC4	177	2 (out)	177 (in)	25 (53)	47	SC4, CT	163	6 (out)	127 (in)

Lowest energy structure for the complex CBS-KKF/e_3a (6) Figure 5.9-a shows the lowest energy structure of CBS-KKF with ligand 200. The structure results from the TS/BH search starting from the SC2a conformation. The CBS adopts an out-in conformation. The guanidinio part is tilted by 124° relative to the pyrrole plain. The side-chain carboxylate 3 is attached to the CBS center and further coordinated by the first lysine side-chain. Beside side-chain 3, the C-terminus is attached to the guanidinio part. All side-chain carboxylates are bridged by the lysine side-chains.

Lowest energy structure for the complex CBS-KKF/e_3A (7) The lowest energy structure for the complexes CBS-KKF/e_3A is shown in Figure 5.9-b (resulting from TS/BH starting from 201_SC1b). The CBS adopts a non-planar in-in conformation which binds three carboxylates, two of which are also hydrogen bonded to the first lysine side-chain. Three out of four carboxylates are clustered by hydrogen bonds to the two lysine side chains.

Lowest energy structure for the complex CBS-KKF/Ae_3 (8) The lowest energy structure for the complex with ligand 202 is shown in Figure 5.9-c (obtained by TS/BH starting from 202_SC2b). The CBS adopts an in-in conformation. Both carboxylates (side-chains 2 and 4) that are attached to the CBS are also hydrogen bonded to the lysine side-chains, which are connected to all the remaining carboxylates.

Lowest energy structure for the complex CBS-KKF/aAe_2 (9) The best result for CBS-KKF and aAe_2 (203) is shown in Figure 5.9-d (resulting from TS/BH starting from 203_SC3a). Compared to the starting structure the binding modality has changed drastically. Conformation of the guanidinio part relative to the pyrrole ring is classified as "out" since the guanidinio part is tilted by 64°. However, both the guanidiniocarbonyl NH and CO are pointing away from the pyrrole ring. Therefore, it is denoted as out*-conformation to distinguish it from the out-conformation as shown in Figure 5.7-a. This conformation enables the CBS to strongly coordinate two out of three ligand carboxylates. Each lysine side-chain is hydrogen bonded to two carboxylates thus bridging all three carboxylates.

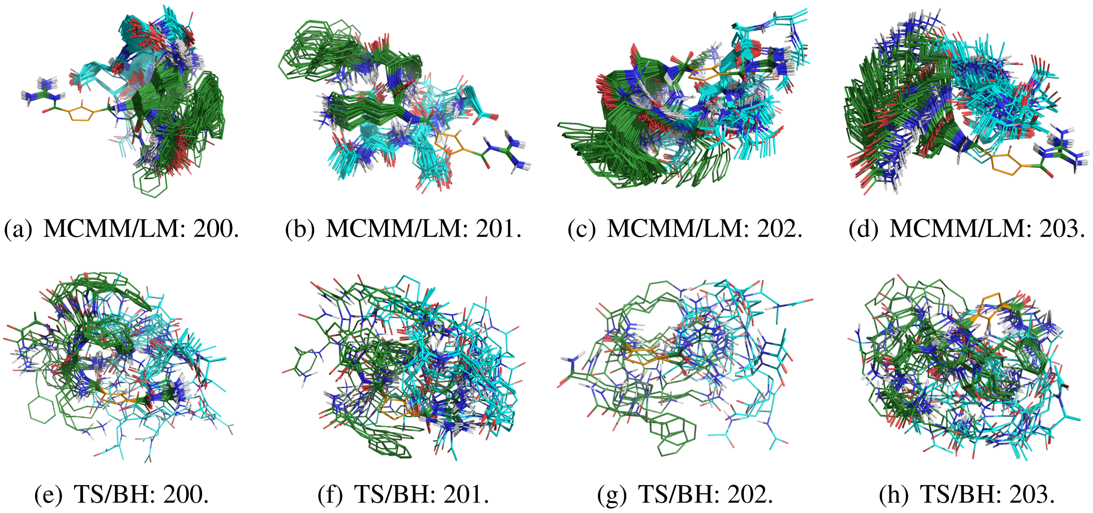
Comparison of TS/BH and MCMM/LM For each complex, several starting structures have been used for both conformational search methods (MCMM/LM-OPLS2005 and TS/BH-OPLS-AA). The force field energies are not appropriate to compare the reliability and efficiency of the two conformational search algorithms. Hence, the energetically lowest result of each search and starting structure was re-optimized using BP86-D3/def2-TZVP combined with COSMO leading to as many re-optimized structures as used starting structures. The relative DFT energies with respect to the lowest minimum for the four complexes are included in Table 5.6 to 5.9. In all cases, the lowest lying structure was found by the TS/BH-OPLS-AA approach, while the minima predicted by MCMM/LM-OPLS2005 are considerably higher in energy (ligand 201: > 14 kJ/mol, ligand 200: > 42 kJ/mol, ligand 202: > 29 kJ/mol, ligand 203: > 42 kJ/mol). Furthermore, the differences of the lowest and the highest energies of the DFT optimized structures are much smaller for TS/BH-OPLS-AA (34 kJ/mol, 82 kJ/mol, 68 kJ/mol and 90 kJ/mol for CBS-KKF/200 to 203) compared to the MCMM/LM-OPLS2005 approach (107 kJ/mol, 130 kJ/mol, 119 kJ/mol and 96 kJ/mol for CBS-KKF/200 to 203) and TS/BH delivered more low lying structures after DFT optimization.

Exemplarily for all receptor-ligand pairs, the results of CBS-KKF/201 (Table 5.6) are discussed in more detail. All other complexes show similar features so that only the differences are highlighted. The DFT results in Table 5.6 to 5.9 indicate that the TS/BH-OPLS-AA approach predicts lower lying conformers than the MCMM/LM-OPLS2005 method. Both methods were carried out with OPLS force fields but cross-checking (re-optimization of the Tabu-Search structures with MacroModel V9.9 and vice versa) revealed that both force fields differ considerably. To differentiate between the quality of the force field and the search algorithm the low lying structures suggested by TS/BH-OPLS-AA were re-optimized within the OPLS2005 force field using MacroModel. In Table 5.6 to 5.9 these values are given in parentheses in column $\Delta E(MM)$ of the TS/BH optimizations. The negative values found for the complexes of CBS-KKF with (**6**) and (**7**) show that various structures predicted by the TS/BH-OPLS-AA approach also lie below the global minimum of the MCMM/LM-OPLS2005 method if identical force fields are used. This underlines the advantage of the TS/BH, since MCMM/LM did not locate these structures. This is not the case for the complexes with (**8**) and (**9**) but for these complexes the lowest DFT structure resulted from TS/BH-OPLS-AA predictions as well.

These results indicate that the TS/BH approach scans wider parts of the phase space more exhaustively. This is proven by Figure 5.10 in which all accepted minima during the searches are superimposed. As an illustrative example, the search run was always taken in which the best structure was found. All structures are aligned at the pyrrole system of the CBS unit. The receptor is shown in green while the corresponding ligands are given in cyan. The much diffuser picture found for TS/BH-OPLS-AA calculations proves the broader scan of the phase space. A comparison of the start with the corresponding final structures provides further details. For MCMM/LM, the carboxylic group placed in the CBS at the start retained its position throughout the whole conformational search so that in the resulting structures often only one carboxylate is coordinated. In contrast, during the TS/BH-OPLS-AA optimizations, the carboxylate which coordinates to the CBS is often changed. Furthermore, TS/BH-OPLS-AA locates many structures in which several carboxylates are bound to the CBS. The complexation of several carboxylates becomes favorable if the planarity of the CBS is abrogated within the conformational search. This happens quite often in the TS/BH searches while in the MCMM/LM runs the CBS stays more or less planar. Furthermore, the CBS unit often flips from the starting out-in orientation to an in-in arrangement and sometimes also in an in-out or out-out orientation within the TS/BH simulations. In MCMM/LM, searches the out-in orientation is mostly retained. Besides the performance of the conformational search algorithm this might also be caused by an overestimation of the rotational barriers of this flip in the OPLS2005 force field.

The TS/BH also located structures in which the hydrogen of NH and the oxygen of CO of adjacent residues are pointing in the same direction (which is not covered in the classification according to Moiani[347], Figure 5.8). These special conformations are marked with an asterisk (*) in table Table 5.6 to 5.9. For the complexes with (**7**) and (**8**), these conformations only represent high lying local minima, but for (**9**) such a special orientation corresponds to the global minimum.

Furthermore, in the case of TS/BH-OPLS-AA, the involvement of the pyrrole-NH in the hydrogen bonding network resulted in several structures with non-planar pyrrole rings. Although the existence of such local minima is supported by DFT, the structures often lie high above the global minimum (e.g. for the structure resulting from 201_CTb, the energy difference is 32 kJ/mol). The distortion is driven by the formation of strong hydrogen bonds.



(a) MCMM/LM: 200. (b) MCMM/LM: 201. (c) MCMM/LM: 202. (d) MCMM/LM: 203.

(e) TS/BH: 200. (f) TS/BH: 201. (g) TS/BH: 202. (h) TS/BH: 203.

Figure 5.10: Superimposed structures accepted during the global optimization from the conformational searches with MCMM/LM and TS/BH which resulted in the lowest energy solution for the receptor-ligand complexes CBS-KKF-200, 201, 202, and 203. All structures are aligned with the pyrrole system of CBS (in orange). The receptor is shown in green, the corresponding ligand in cyan.

Conclusion In summary, the comparative study of the two conformational search algorithms revealed a better performance of TS/BH-OPLS-AA with respect to MCMM/LM-OPLS2005 for the investigated systems. TS/BH did not only deliver lower energy structures in a smaller energy range, as proven by the comparison of the DFT energies, but also seemed to cover the phase space more widely. Furthermore, TS/BH yielded special conformations at the binding site which have not been found before. The comparison of the four different receptor/ligand complexes shows this to be a general trend of the TS/BH approach. Therefore, this method is a very useful tool for the determination of low lying conformations of such systems.

5.3 *ab initio* Conformational search

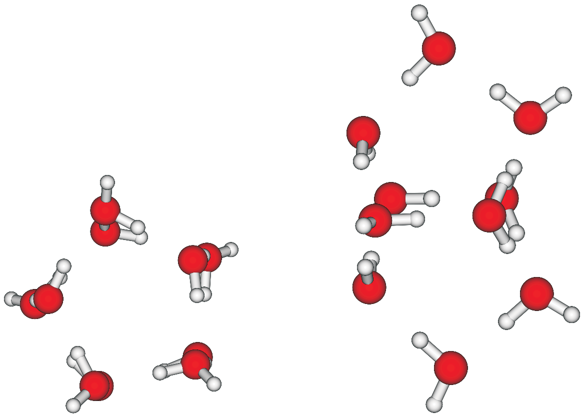
Motivation As discussed in section 5.2, the proper investigation of conformational search studies often include a pre-scan with force field methods and a refinement process with more elaborate methods to obtain accurate relative energies of the conformations. However, the refinement process has to be done very carefully as important structures can be very easily missed. Usual *ab initio* approaches are by far too demanding for a complete conformational search. However, recent improvements in hardware architecture, with the introduction of general purpose graphical processing units (GPU), offer new opportunities. The GPU-accelerated program TeraChem enables DFT calculation on GPUs for very big systems in reasonable time scales.[304] This further enables exhaustive studies using *ab initio* or DFT methods. For this study, the interface between the Tabu-Search algorithm and TeraChem already described in section 4.1.4 was used to investigate the usefulness of this approach at hand of several examples.

Computational Details The described calculations are either performed on a normal workstation (GTX280, TeslaC1060, AMD Phenom II X4, 3.2 GHz) or on a beowolf-GPU-Cluster (each node: 4xGTX480, 2x Intel Xeon 2.6 GHz, 96 GB RAM).
A developmental version of TeraChem (1.45) was used for *ab initio* calculations. All force field calculations were performed using CAST and the OPLS-AA force field parameters.[232–235] All calculations were performed using different DFT-functionals (BLYP or B3LYP) using the dispersion correction D3 by Grimme as implemented in TeraChem. Details of each calculation are given separately for every example. For global optimization with TeraChem, the interface described in section 4.1.4 was used.

5.3.1 $(H_2O)_{10}$

The global optimization of the $(H_2O)_{10}$ cluster is a problem which is well investigated in the literature. Searching the literature reveals several publications where the global minima of water cluster are discussed[125,333,348–351] including the global minimum of $(H_2O)_{10}$.[333,348] Performing the global optimization with OPLS-AA and the TIP3P water potential,[332] the two lowest lying minima could be obtained. Nevertheless, the energetic order of the minima is wrongly described in the force field.
When starting the global optimization from the same starting points, but using TeraChem with DFT (B3LYP-D/6-31G++), the correct global minimum structure is always obtained. These results clearly show that *ab initio* calculations will deliver more accurate results for

such cases. The final structures are shown in Figure 5.11 and the concerning relative energies are given in Table 5.10.



(a) Global minimum of DFT.

(b) Global minimum of OPLS-AA.

Figure 5.11: Global minima found for $(H_2O)_{10}$ with either B3LYP-D3/6-31++G (5.11-a) or OPLS-AA (5.11-b).

Table 5.10: Results of the global optimizations using Tabu-Search and the $(H_2O)_{10}$ cluster. OPLS-AA results were obtained by the force field implementation of CAST, B3LYP results are from the CAST-TeraChem interface. The global minimum for the corresponding method is always set to 0.0 kcal/mol. As it can be seen from the 0.00 kcal/mol energy structure, that the lowest minimum has changed when switching from OPLS-AA to B3LYP.

Structure	OPLS-AA/TIP3P	B3LYP-D3/6-31++G
Figure 5.11-a	0.32	0.00
Figure 5.11-b	0.00	1.26

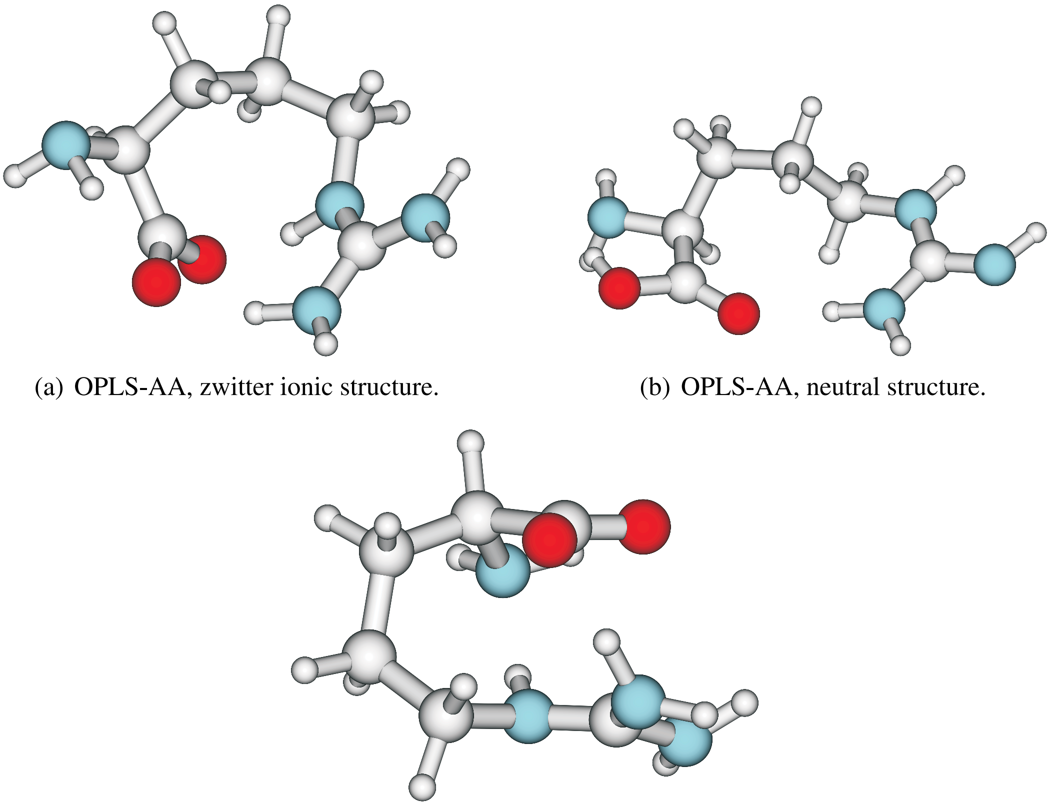
5.3.2 Arginine

Extensive conformational searches on the arginine molecule by Schlund *et al.*[352,353] showed that the global minimum structure of arginine contains a chair like conformation of the guanidinium and carboxylate residues. Usual conformational searches using force fields do not represent this structure correctly. Nevertheless, the chair like structure or at least a structure lying very close to this point can be obtained with CAST-TeraChem.

The global search using TeraChem started from the neutral form and finally resulted in the zwitter ionic form. For global optimization using OPLS-AA, the search was either started from neutral or zwitter ionic form. The calculations with TeraChem were performed with BLYP-D/6-31G.

In CAST-TeraChem, a chair like conformation of the zwitter ionic form of arginine was predicted as the global minimum. The structure obtained by CAST-TeraChem is already very similar to the structure obtained by refinement with MP2 and CCSD(T). However, the global minimum predicted with CCST(T) is in neutral form, which might be due to a wrong description in DFT.[353] The structures obtained by global optimization with the force field are located far away from this geometry. The final structures of each global optimization can be seen in Figure 5.12.

The example of the global optimization of arginine clearly illustrates the great advantage of the use of an *ab initio* conformational search method. The starting structure of the global optimization with Tabu-Search-TeraChem was in a neutral form. Nevertheless, during the global optimization a proton "jumped" from the carboxylic group to the guanidinium group yielding the zwitter ionic form, which is also more stable in the DFT approach. A search like this is only possible using *ab initio* methods. Furthermore, the interaction important for the chair-like conformation (Figure 5.12-c) is not properly included in force fields. The same results can be obtained by force field conformational search as shown by Sebastian Schlund[352,353], however, this includes an exhaustive conformational search and a refinement process of many different structures. Starting from the results obtained by an *ab initio* conformational search with a moderate theory (like DFT), the refinement with high-level



(a) OPLS-AA, zwitter ionic structure.

(b) OPLS-AA, neutral structure.

(c) CAST-TeraChem (BYLP-D/6-31g).

Figure 5.12: Final results of global optimization of argine, ΔE is the energy difference between starting point and final energy result in kcal/mol;
a) Global minimum of Tabu-Search-OPLS-AA starting from a zwitter ionic structure, $\Delta E = -47.0$
b) Global minimum of Tabu-Search-OPLS-AA starting from a neutral structure, $\Delta E = -5.3$
c) Global minimum of Tabu-Seach-TeraChem (BYLP-D/6-31g), $\Delta E = -16.7$

ab initio methods can be accelerated. The conformation is already very close to the global minimum in CCSD(T). Only slight rearrangements and the zwitter ionic state have to be changed.

5.3.3 [Met]^{5}enkephalin

For bigger systems like [Met]^{5}enkephalin, the investigations revealed a performance improvement by using a pre-optimization with a force field. After leaving a local minimum using a DFT method, the resulting structure is pre-optimized to release possible strains from the molecule, which can occur during the modest ascent strategy. The obtained structure is then directly optimized by DFT. The ranking of the structures within the global optimization is performed with DFT energies.

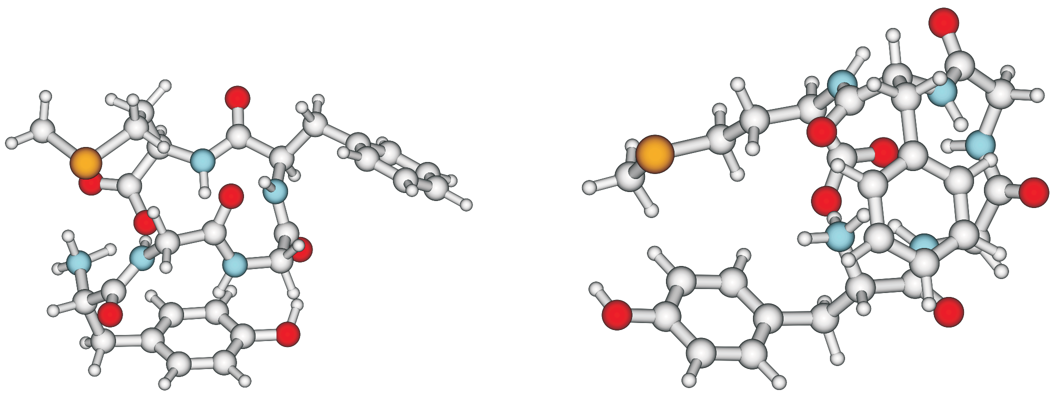
During the test calculations, the SCF calculations sometimes did not converge in the modest ascent part. Nevertheless, it was found to be more efficient to simply continue the search with the next optimization step instead of repeating or restarting the modest ascent. This is mainly due to the fact that most of the time the structure is already quite far away from the last minimum and can be used as new starting point.

Figure 5.13 shows the results of the global optimizations using CAST-TeraChem and CAST-OPLS-AA. Comparing the results obtained by CAST-OPLS-AA with the results obtained from CAST-TeraChem one can easily see, that the optimized structure in the force field has about 2 hydrogen bonds. The structure obtained by CAST-TeraChem possesses 4 hydrogen bonds. The energy difference between the starting point and the final result is a little bit higher for CAST-TeraChem as for CAST-OPLS-AA.

Re-optimizing the final structure of CAST-TeraChem within the OPLS-AA force field delivers a structure which is about 2.4 kcal/mol higher in energy than the final result from CAST-OPLS-AA. Vice versa, a re-optimization of the final structure of CAST-OPLS-AA with TeraChem (BLYP-D/6-31g) delivers a structure 4.7 kcal/mol higher in energy. This shows on the one hand, that the Tabu-Search algorithm properly delivers the global minimum within the employed theory. On the other hand, this shows that the important interactions in DFT seem to be neglected in the force field as the global minimum structure of DFT is higher in energy in the force field. This influences the final results from global optimizations dramatically. Therefore, in some cases it might be preferable to perform the global optimization directly within *ab initio* methods instead of a refinement process afterwards.

Conclusions The results shown above clearly show that the combination of CAST and TeraChem delivers valuable results. Of course, the computational effort will increase appreciably by using an *ab initio* routine instead of a simple force fields. Nevertheless, the advantage of *ab initio* methods for global optimization might be worth the increased effort.

Looking at the first example more closely, the CAST-TeraChem approach directly delivers the right global minimum structure. The force field does not describe the rank of the obtained



(a) Global minimum of DFT.

(b) Global minimum of OPLS-AA.

Figure 5.13: Global minima found for [Met]^{5}enkephalin with either BLYP-D/6-31g (5.13-a) or OPLS-AA (5.13-b).

minima correctly. This would require a refinement using *ab initio* methods. Of course, one never knows if the force field predictions are correct. Therefore, the refinement would be necessary in every case. The global optimization with *ab initio* methods is even more important for the example of arginine. Here, the right global minimum structure was not found at all by the force field approach.

5.4 Solvation of molecular systems

5.4.1 Global optimized solvent shell for the mini protein chignolin

Motivation This work is summarized in ref. 320. The proper description of explicit water shells is of enormous importance for all-atom calculations. To prove the usefulness of the new approach described in section 4.3, its efficiency is compared with standard molecular dynamics using the chignolin protein as a test candidate. The artificial mini-protein chignolin whose structure was revealed by NMR spectroscopy[354] represents an ideal test system to compare the efficiency of different approaches since the computed structures seem to depend strongly on the simulation protocol.[355,356] The orientation of the two residues Tyr-2 and Trp-9 is most problematic. In NMR, an edge-to-face (t-shaped) orientation of the aromatic rings of the residues is obtained. In contrast, MD simulations performed by Suenaga[355] and Satoh[356] on the basis of the Amber force field[236] mostly predicted more parallel configurations while the t-shaped conformation is rarely found. It could not be clarified, whether the theoretical or the NMR data were wrong. To shed light on this important issue we employed the OPLS-AA force field in combination with the TS-based approach to build up an appropriate water shell and to compute geometrical structures of the protein. The results are further analyzed by DFT computations.

Both algorithms describe the water shell similarly well, however, the new approach seems to deliver a solvation with an improved micro-solvation. Furthermore, the new approach enables a step-wise build-up of the solvent shell, so that the more important inner part can be prepared more carefully. It can furthermore generate solute structures which are either biased to the (experimental) starting structure or the underlying theoretical model, i.e. the employed force field.

Computational details All computations started from the β-hairpin structure of chignolin as given in the pdb-database (PDB-Code: 1UAO, amino acid sequence: Gly-Tyr-Asp-Pro-Glu-Thr-Gly-Thr-Trp-Gly).[354] MD simulations were performed with a time step of 1 fs in the NVT ensemble at 310 K. The system was heated in 10 K steps from an initial temperature of 10 K to the final temperature. During heating, the velocities were scaled to stay in reasonable ranges. The starting structure for MD (chignolin plus solvent shell) was prepared using the VMD program.[324] The protein molecule was placed in a pre-equilibrated sphere of 526 water molecules with a diameter of about 30 Å. During the simulations, spherical boundary potentials were applied to avoid evaporation of water molecules. Finally, 2000 snapshots (each 1 ps) were optimized using the L-BFGS algorithm implemented in CAST (see section 4.1). To check whether the water shell is sufficiently large to ensure converged properties of the protein, a bigger system (diameter about 38 Å and 1307 water molecules) was also simulated using NAMD[317] employing the Charmm27[239,357] and the OPLS-AA[232,336,358] force fields with a 1 fs time step in an NVT ensemble. After heating from 25 K to 310 K and running a 2 ns classical equilibration with frozen solute and spherical boundary conditions, a 5 ns production run was performed. All 1000 snap shots were minimized using the conjugate gradient algorithm implemented in NAMD.[317] The results are in good agreement with the MD simulations with CAST. Therefore, the focus will be set on the results obtained from the simulations of the smaller water sphere.

The TS calculations were performed using the TS-Dimer approach and the solvation algorithm described in section 4.2. Each global optimization run encompassed 1000 main iterations (*steepest descent - modest ascent* steps). The step size of the basin hopping diversification was adjusted to the system size and varied between 3.0 Å in the first to 0.7 Å in the last solvation step. For the step-wise solvation, 5 solvation steps were performed with 30, 115, 226, 467, and 526 water molecules.

All calculations with the CAST program were performed employing the OPLS-AA forcefield in combination with the TIP3P water potential[332] for surrounding water molecules. No cut-off radius was used for non-bonded interactions. For placement of new solvent molecules, the program VegaZZ[321] was used.

The obtained results were further analyzed by computations for a model system which comprises the phenol ring of Tyr-2 and the indole system of Trp-9 where one hydrogen atom was added at each ring to saturate the dangling bonds (Figure 5.14). Single point calculations and geometry optimizations with DFT were performed. For these calculations RI-DFT (B3LYP)[359–361] with D3 dispersion correction[342] was used in combination with

6-311G** basis set using the Turbomole 6.3.1 program.[338] COSMO[343] was used as a continuum solvent model (ε = 78.5).

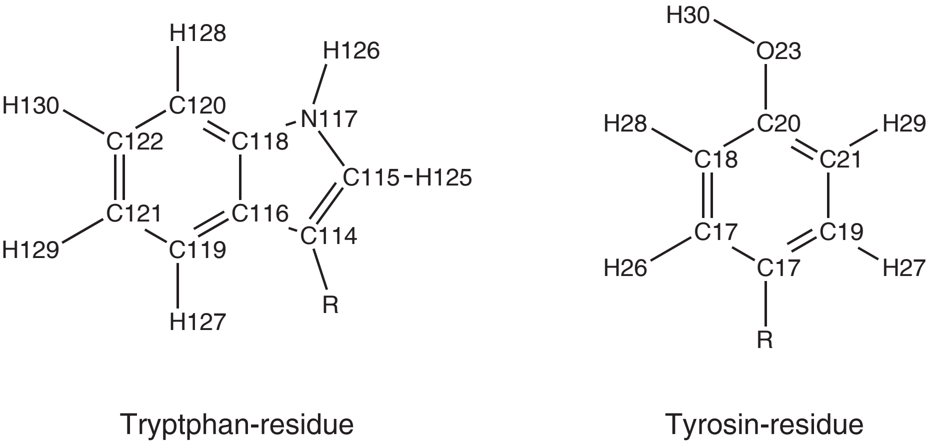


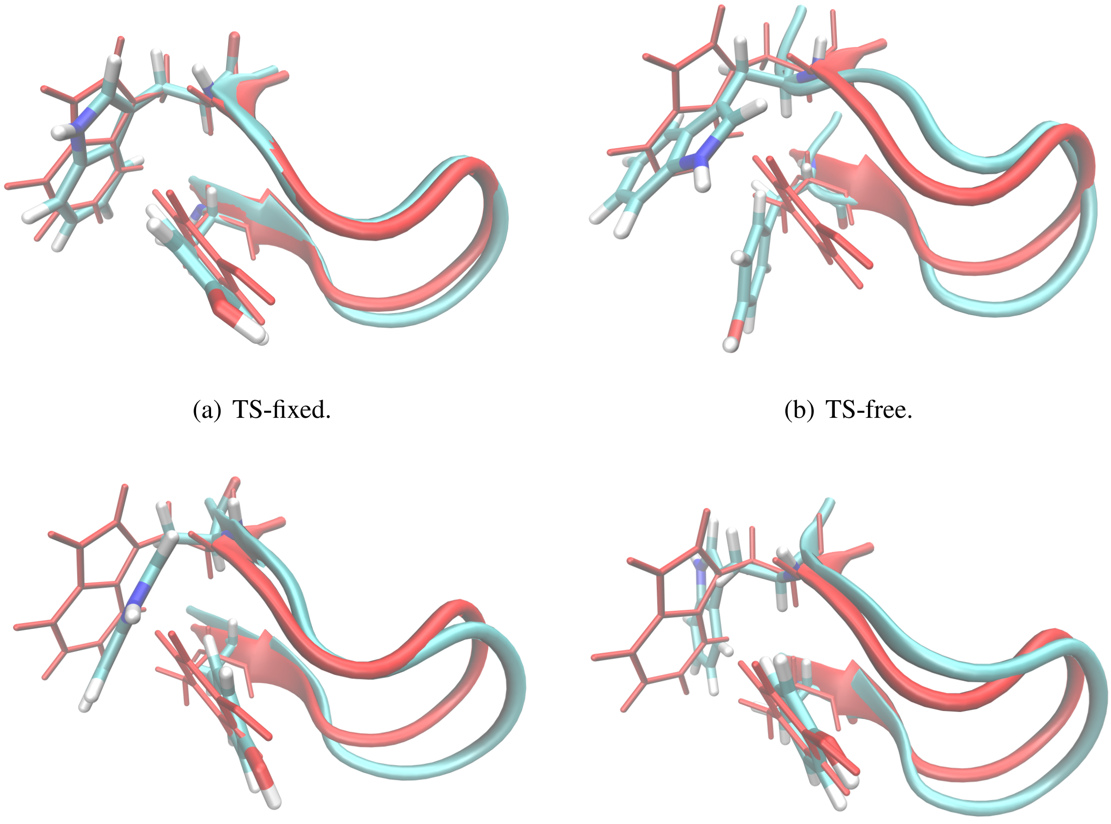
Figure 5.14: Tyr-2 and Trp-9 residues of the protein chignolin. For DFT calculations, R was to H. The corresponding atom-based RMSD values can be found in the appendix Table A.5.

Results and discussion Table 5.11 compares the various approaches in terms of the required CPU time and the energy of the energetically lowest lying geometries. While the MD-free calculation required about 174 CPU-h in total, the TS-fixed and TS-free approaches needed 150 and 93 h, respectively. Both approaches delivered arrangements which are considerable lower in energy than the MD-free approach (281 vs. 243 kJ/mol). The TS-complete strategy needed considerably more time than both other TS approaches but delivers similar results. Hence, this approach seems to be less efficient.

Table 5.11: Table for results of solvation of chignolin with 526 water molecules.
[a] Relative energy with respect to the best solution obtained by TS-fixed. For more information see text.

Method	CPU-time [h]	ΔE [kJ/mol][a]
MD-free	174	+ 281
TS-fixed	150	0
TS-free	93	+ 38
TS-complete	285	- 33

The root mean square deviations (RMSD) from the starting structure of the complete system (water + enzyme) in the MD simulation converged to a final value of about 17 Å, i.e. the radius of the complete system. This shows the good equilibration of the system indicating that the simulation time was sufficient. The total energy obtained from geometry optimization of selected frames has a standard deviation of 63 kJ/mol and a maximal fluctuation of about 502 kJ/mol. The RMSD values of the protein and the backbone are 2.5 Å and 1.0 Å, respectively, showing only modest general deviation of the secondary structure of the protein with



(a) TS-fixed.

(b) TS-free.

(c) MD-fixed.

(d) MD-free.

Figure 5.15: Energetically lowest predicted structures of the 1UAO protein (cyan). The arrangement of the Tyr-2 and Trp-9 residues is indicated. The reference NMR-structure is shown in red. The corresponding atom-based RMSD values relative to the NMR structure can be found in the appendix Table A.5. Atom numbering is done following Figure 5.14.

respect to the starting structure. The RMSD of the TS-free simulation closely resembles the MD-data. As expected, the TS-fixed optimization reveals much smaller RMSD values for the protein and the backbone (1.0 Å and 0.5 Å, respectively). This shows that the TS-fixed simulation indeed stayed much closer to the starting structure. Figures showing the RMSD of all systems as well as the total energy behavior during the optimizations are given in the appendix(see Figure A.1, A.2, and A.3).

All simulations agreed with the NMR experiment in an overall β-hairpin structure of chignolin and the orientation of most residues. Only very flexible and solvent-exposed side chains deviated. Especially, and in accordance with the literature,[355,356] considerable differences are found in the relative orientations of the Tyr-2 and Trp-9 residues (Figure 5.15, also see Table A.5 in the appendix for atom based RMSD values). The final structure obtained from the TS-fixed simulation (Figure 5.15-a) closely resembles the reference structure from NMR in which the Tyr-2 and Trp-9 adopt a T-shape configuration. In all other calculations, the two residues drifted apart and formed more parallel arrangements (Figure 5.15-b, 5.15-c, and 5.15-d). Such orientations were also predicted by simulations which employed the Amber force field.[355,356] The results of the two MD simulations, MD-free and MD-fixed, are very similar. As discussed above, the differences between the TS-fixed and TS-free simulations indicate that either the force field underestimates the stability of the T-shaped structure in comparison to the parallel one, or that the NMR data were erroneously interpreted.

To shed light on this issue the force field (OPLS-AA) computations were compared with B3LYP-D3 calculations for the model system depicted in Figure 5.14. The calculations started from the relative orientation of both residues as predicted by NMR, TS-free, and TS-fixed. For all structures, the DFT energies of the force field structure and the DFT-optimized structure were calculated. Likewise, the the OPLS-AA energies for the DFT and the force field- optimized systems were calculated. The results are shown in Table 5.12 and the trend is visualized in Figure 5.16. The DFT approach predicts the T-shaped orientation to be more stable (6 kJ/mol). The energies obtained by the force field of the same structure promote the parallel orientation as more stable (by about 5 kJ/mol). The same is true for the force field optimized structures although the energy differences are smaller. These results indicate deficiencies of the force field.

Table 5.12: Results of the calculations on the residues Tyr-2 and Trp-9. All energies are given in kJ/mol and are relative to the energy obtained with a single point calculation on the NMR-structure. Schematic view of the used system can be found in Figure 5.14. a) Predicted by NMR and TS-fixed. b) Predicted by TS-free, MD-free, and MD-fixed. c) B3LYP-D3/6-311G**, COSMO (ε = 78.5).

		Intermonomer orientation	
Method	Geometry from	T-shaped ^{a}	Parallel ^{b}
DFT ^{c}	OPLS-AA	-0.3	1.2
OPLS-AA	OPLS-AA	-4.8	-6.5
DFT ^{c}	DFT ^{c}	-5.2	0.4
OPLS-AA	DFT ^{c}	-0.6	-5.4

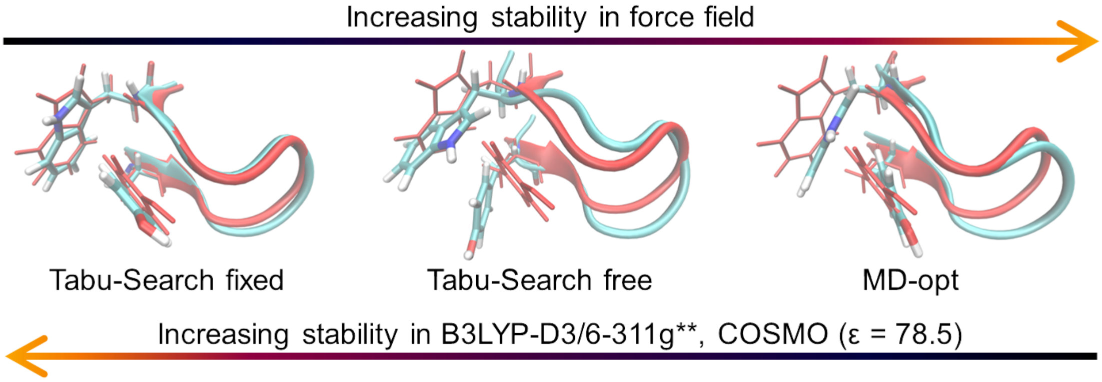
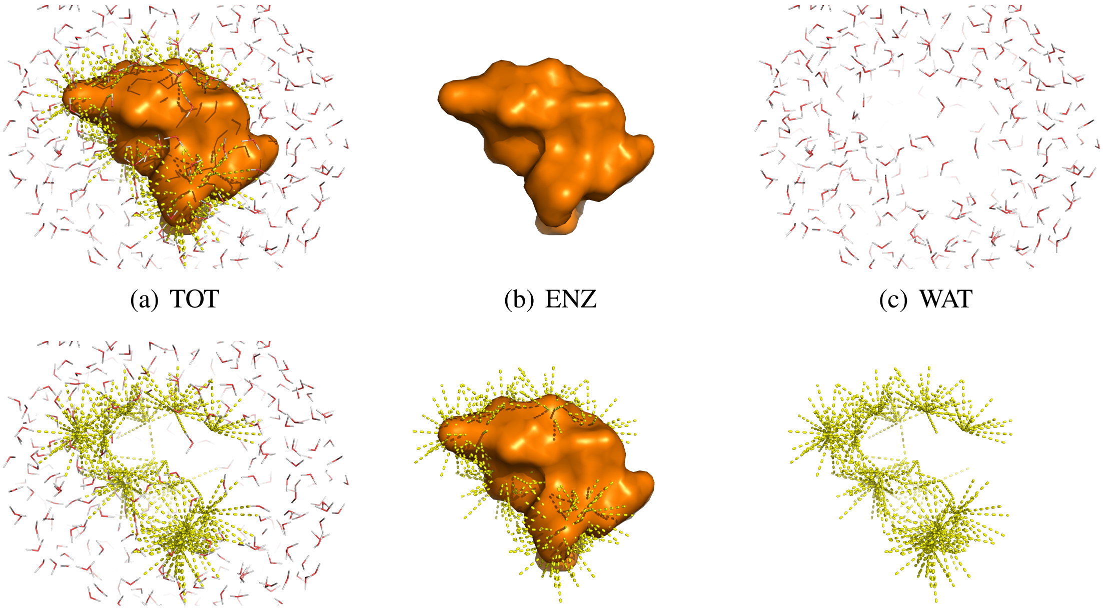

Increasing stability in force field
Tabu-Search fixed
Tabu-Search free
MD-opt
Increasing stability in B3LYP-D3/6-311g**, COSMO (ε = 78.5)

Figure 5.16: Stability of the Tyr-2 and Trp-9 interaction in OPLS-AA and DFT. The reference structure from NMR is given in red.

The different geometry predictions of MD and TS-free on the one hand and TS-fixed on the other could be caused by the latter simulations predicting non-meaningful water shells. To eliminate this possibility, Table 5.13 analyzes the various contributions to the total energy in more detail while Figure 5.18 compares the radial distribution function (RDF) predicted by MD-free, TS-fixed, and TS-free, respectively. For clarity the RDF of the MD simulation (with non-optimized frames) is also included. In Table 5.13, E(TOT) denotes the total energy of the complete system, while E(ENZ) and E(WAT) are the energies of the isolated enzyme and the water system obtained as single-point computations on the geometries obtained from the simulations. An illustration of the subsystems is given in Figure 5.17. Interesting insights are obtained from E(TOT) - E(ENZ) - E(WAT) which gives the interaction energy between the enzyme and the water shell, i.e. it reflects differences in the micro-solvation.

Table 5.13: Analysis of the energy contributions to the water-enzyme stabilization. For more information see text. All energies are given in kJ/mol.

Split energy systems	MD-free (lowest structure)	TS-fixed	TS-free
E(TOT)	-28206	-28487	-28453
E(ENZ)	-1361	-1005	-1306
E(WAT)	-23852	-23848	-24183
E(TOT) - E(ENZ) - E(WAT)	-2993	-3634	-2964


(a) TOT

(b) ENZ

(c) WAT

(d) TOT-ENZ

(e) TOT-WAT

(f) TOT-ENZ-WAT

Figure 5.17: Illustration of the subsystems used for calculating the energy contribution the the enzyme-water interaction.

As expected for a smaller peptide the energy of the water shell completely dominates the total energies (84-85 %). These differ by only about 1% and also the RDFs given in Figure

5.18 indicate high similarity. For distances larger than 4 Å, the RDF predicted by MD-free is more diffuse but the overall differences to the TS based methods are small. All simulations predict a first maximum at around 1.75 Å, while the second and the third distinct maxima are found at about 2.65 Å and 3.60 Å, respectively. These comparisons prove that the TS based approaches predict meaningful water shells.

Nevertheless, the TS-fixed simulation computes somewhat more structured water shells than the other simulations. Its first maximum appears at R = 1.65 Å and an additional shoulder is found at about R = 1.85 Å. An additional smaller maximum appears at R = 2.25 Å. More distinct patterns are also predicted for distances up to 8 Å. The TS-free simulation predicts, in accordance with the MD-free simulation, only one maximum which lies at R = 1.65 Å for TS-free and R = 1.75 Å for MD-free. For distances larger than 4 Å its RDF closely resembles the TS-fixed simulation. All these differences indicate that the TS-fixed simulation predicts a network in which the water molecules can occupy more strongly bound positions than in both other simulations. This may result from vacancies which remain open in the TS-fixed simulations due to the fixation of the protein atoms during the relaxation of the water molecules but are filled in both, TS-free and MD-free.

A similar interpretation is also appropriate for the interaction energies between solute and solvent (E(TOT) - E(ENZ) - E(WAT)). For the energetically lowest results of MD-free and TS-free these interactions contribute 10.6 and 10.4 % to the total energy, respectively. For the lowest structure of TS-fixed this contribution increases to 12.8 %. This proves that the TS-fixed approach yields water surroundings in which the micro-solvation interacts more strongly with the solute. The investigation of the behavior of the water-enzyme interaction energy in comparison to the water-water interaction energy during the progress of the optimization procedures delivers even deeper insights into the quality of the micro-solvation. Both interactions are illustrated in Figure 5.19. The energies are given relative to the first frame of the simulations. The range of the total energy values is in the range of the values given in Table 5.13. In the MD-free and the TS-free simulations, the energy of the water shell decreases during the progress of the simulation. However, the interaction of the enzyme with the water shell increases only in the beginning and equilibrates at a certain value. In case of TS-free, the enzyme-water interaction stays more or less constant and does not affect the optimization. In the MD-free simulation, however, the enzyme-water interaction is destabilized during the simulation. The fluctuation (standard deviation) of this energy contribution is about 569 kJ/mol. In TS-fixed calculations, the trend is completely the opposite. During the very first frames, the water-water interaction is stabilized by about 50 kJ/mol, while the water-enzyme interaction is more or less unaffected. However, in the proceeding optimization the water-water interaction is destabilized again in favor of an increased water-enzyme interaction. In the end, the two interactions converge to similar stabilization energies of about 100 kJ/mol resulting in a well-balanced and optimized micro-solvation.

In all calculations the interactions within the enzyme itself (Table 5.13, E(ENZ)) represent only about 5% of the total energy. The slightly smaller contribution found for the TS-fixed simulation results from the geometry of the protein being biased towards the NMR structure but not to the underlying force field.

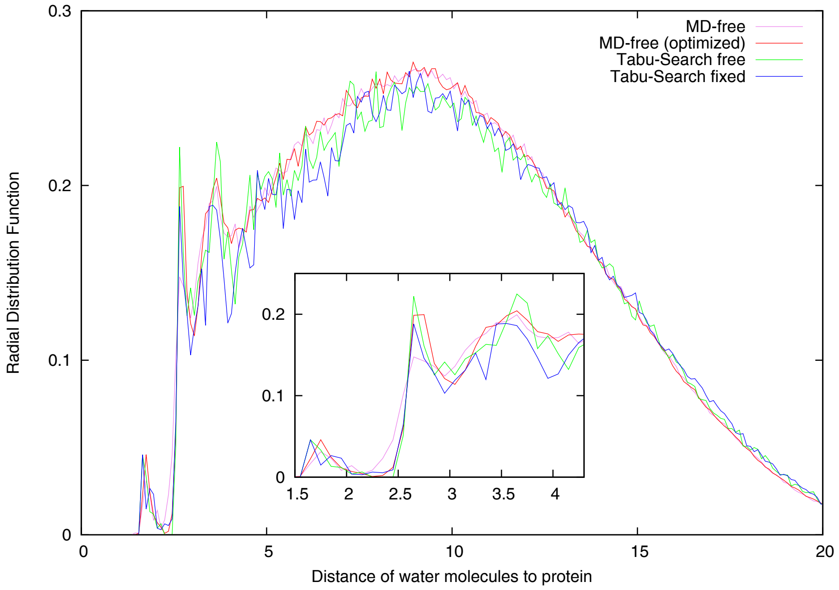
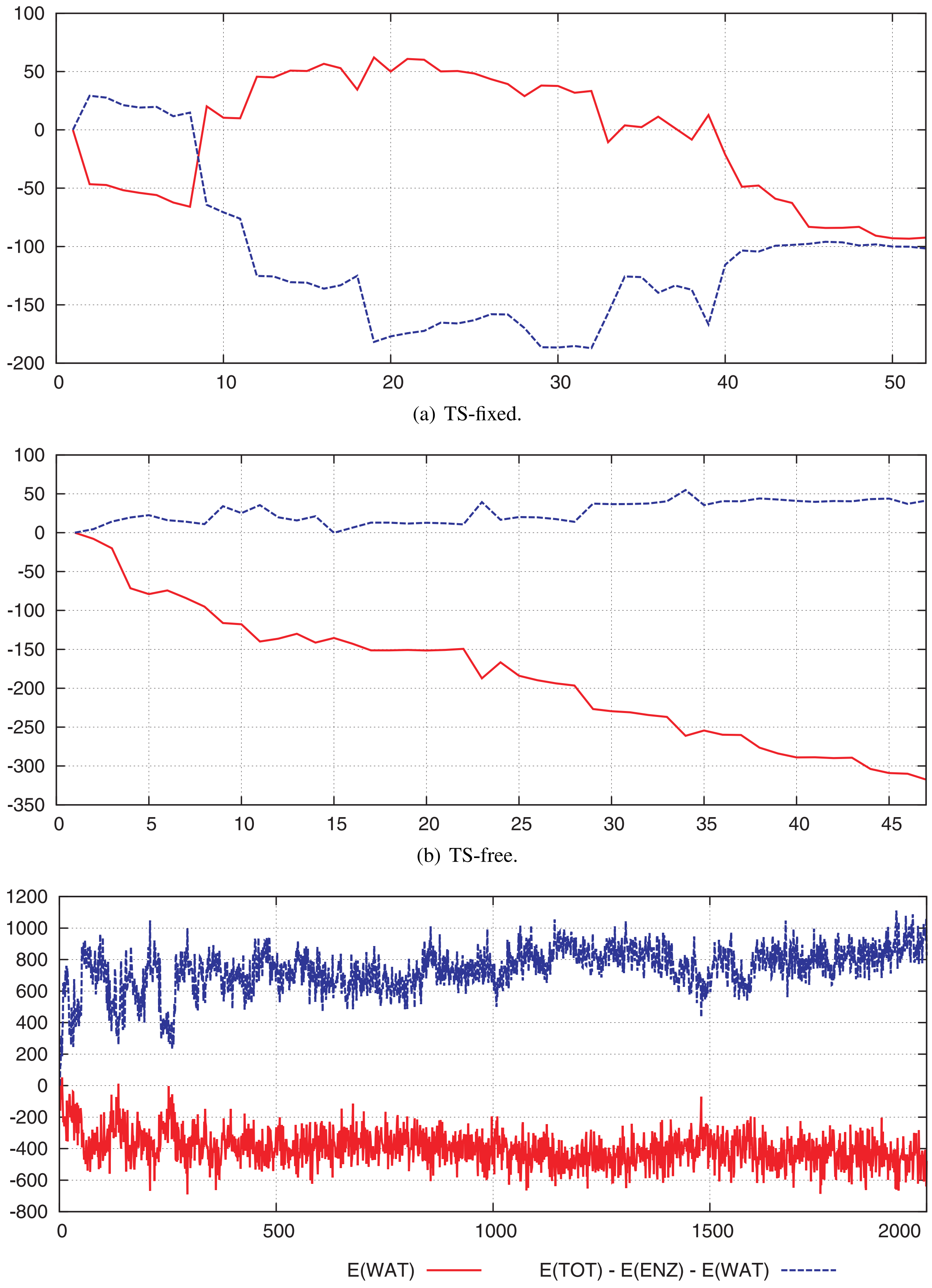


Figure 5.18: Radial distribution function (RDF) of water molecules around the chignolin protein. The 20 lowest structures of each simulation were taken for calculation of the RDF.

Conclusion The newly developed algorithm for solvation based on Tabu-Search optimization seems to be a very useful tool. The nature of the optimization algorithm allows for a step-wise built up of the solvent shell. This enables a more careful preparation of the more important inner part of the solvation shell (micro-solvation). By adjusting the optimization protocol, either a bias to the (experimental) starting structure or to the used force field can be employed. The investigation showed that the water shells predicted by the TS simulations provide an accurate description of the micro-solvation. The algorithm can be applied straightforwardly and is, therefore, an useful tool for the preparation of starting structures for subsequent QM/MM computations.



(a) TS-fixed.

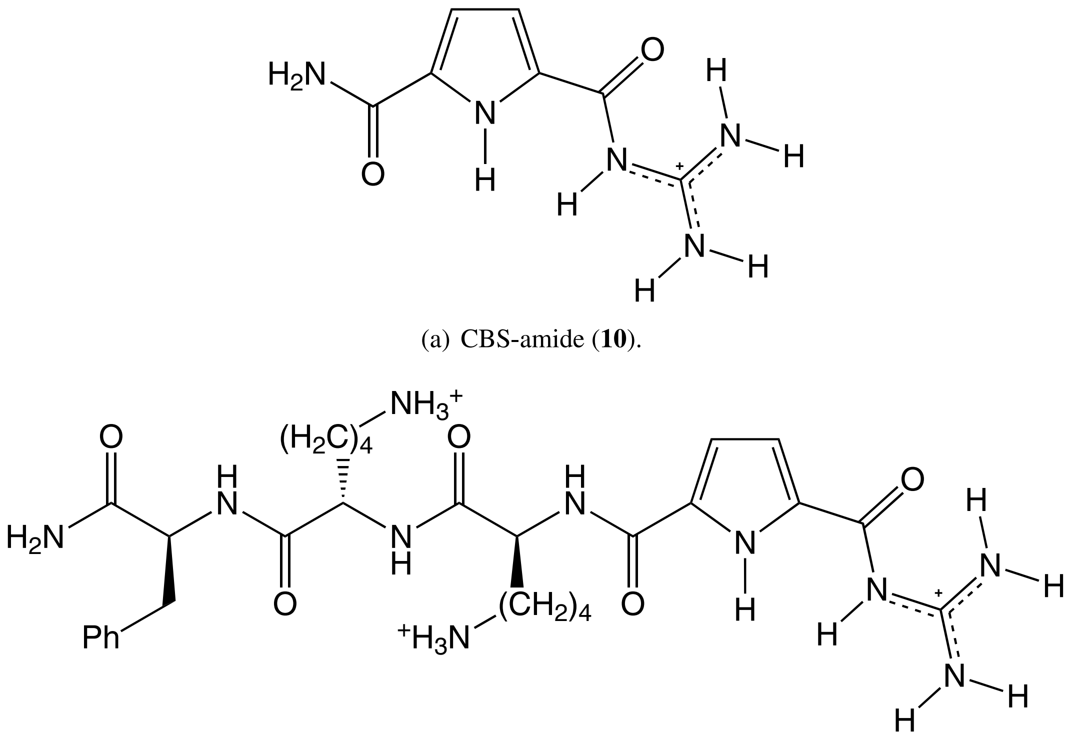
(b) TS-free.

(c) MD-free.

Figure 5.19: Progress of the energy contributions of the solvation shell for the different simulation protocols. The abscissa gives the frame number; the ordinate gives the energy values. Blue: E(TOT)-E(WAT)-E(ENZ). Red: E(WAT). For more information see text. All energies are given in kJ/mol and are relative to the first frame.

5.4.2 Solvation of CBS-systems

The proper description of the solvent shell and the micro-solvation is also of great importance when vibrational spectra of solvated systems are calculated and compared to experimental results. Many effects can be treated by implicit solvent models like PCM[362,363] or COSMO.[364,365] However, hydrogen bonds from the solute to the solvent cannot be handled by these methods. Therefore, an accurate treatment of an explicit solvent shell has to be taken into account which was already shown in an exhaustive study of Ghomi *et al.*[366–374] Within his PhD thesis, Stefan Niebling (University of Osnabrück) investigated the receptor CBS-KKF (**11**) and the CBS-amide (**10**) (already described in subsection 5.2.2, see Figure 5.20) experimentally by vibrational spectroscopy in aqueous solution. For the theoretical description, both, the number and the position of the solvent molecules is essential. Therefore, the solvent shells for the systems (**10**) and the (**11**) were build up with the Tabu-Search procedure described in section 4.3. Measurements and the calculation of vibrational spectra of these systems was done by Stefan Niebling (University of Osnabrück). For further detail, I refer to his thesis. At this point, I focus on the results of the conformational search studies.



(a) CBS-amide (**10**).

(b) CBS-KKF (**11**).

Figure 5.20: CBS-systems taken for building up the solvent shell with Tabu-Search.

CBS-amide Table 5.14 shows the final results of the global optimization of the CBS-amide (**10**) with different numbers of water molecules. To ensure, that all four possible conformations of the CBS-unity are obtained (in-in, in-out, out-in, out-out; see Figure 5.7-a for further detail) the central atoms of CBS-amide have been fixed during the global optimization. The energy values are given relative to the lowest results with the same number of water molecules. Throughout the solvation procedure, the out-in and out-out conformations are the most stable ones. The stability of the in-in conformation, which is the least stable arrangement in gas phase, strongly increases with the number of added water molecules. For 32 water molecules, the in-in conformation becomes even a little bit more stable than the in-out and out-out conformation. However, the larger solvent shell might have a huger influence on the total energy than the CBS-system. To compare the different systems, Stefan Niebling re-optimized the lowest structures using RI-PBE0-D3/def2-TZVP with the COSMO-solvent model (water). The results of these calculations are shown in Table 5.15. Comparing the relative force field and relative DFT energies, it can be seen, that those minima predicted as very stable in the force field also appear as stable in DFT. However, the energetic order of the minima has changed. A very interesting change in the relative energies can be seen by the inclusion of the COSMO model. Here, the in-in and out-in conformations are stabilized significantly.

To compare the structures to the experimental results, the infra red and resonance-raman spectra of these systems were calculated. The out-out conformation showed the best agreement with the experiment. Therefore, only the spectra of this conformation is shown for illustration of the performance of solvated structures obtained by Tabu-Search (Figure 5.21). The best agreements of the calculated spectra with the experiment are marked in green. A comparison of the calculated and the experimental spectra (done by Stefan Niebling) revealed no clear trend in the improvement of the spectra by adding more water molecules. However, it seems that the description of the spectra gets better with the first few water molecules, as most best agreements (green areas) appear for structures with a mediate amount of water molecules. Investigating the structures more carefully reveals, that as long as the first solvation shells is treated explicitly the description is improved. However, as soon as the second solvation shell is started the description gets worse. This is also in accordance to investigations of Ghomi *et al.*[372] They claimed, that the number of water molecules should be a minimal but sufficient number. Furthermore, they emphasized that the optimal number of water molecules can be the number of hydrogen-bond donors and acceptors. In case of the CBS-amide, 10 donors and acceptors are present. Therefore, a maximal number of 10 water molecules should be sufficient which correlates with the results obtained by our study. These water molecules can be placed accurately by the global optimization with Tabu-Search. As the first solvation shell seems to be most important, after global optimization of the complete water shell only the first solvation shell should be included for the calculation of the vibrational spectra (i.e. deletion of the remaining water molecules).

CBS-KKF The complete receptor (**11**) was also investigated and solvated. Table 5.16 shows the results of the solvation procedure. A comparison of the different solvation

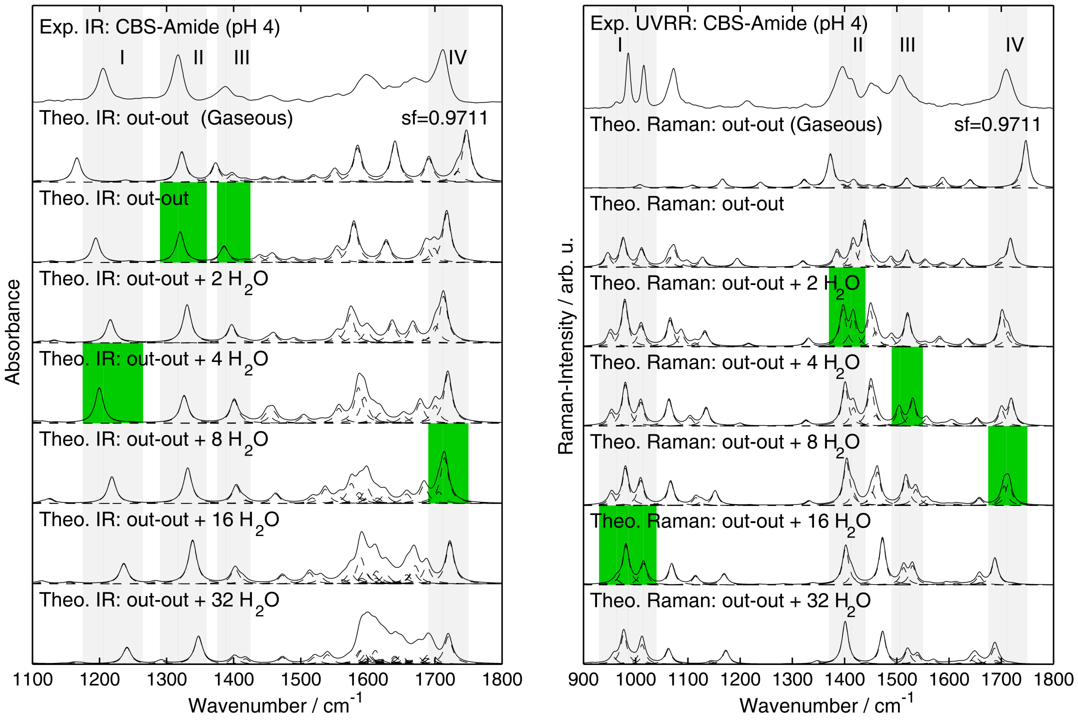
Table 5.14: Solvation of CBS-amide with different number of water molecules. The global optimizations were started from the four possible conformations shown in Figure 5.7-a. The conformation of the central CBS unit and the adjacent amide systems was fixed during the optimization to avoid conversion into different configurations. Energies are given in kJ/mol and relative to the lowest found result with the same number of water molecules.

#(water molecules)	in-in	out-in	in-out	out-out
0	82	13	38	0
2	59	8	17	0
4	67	0	25	13
8	13	0	4	4
16	17	4	4	0
32	17	0	33	33

Table 5.15: Relative energies (kJ/mol) of the re-optimized structures (RI-PBE0-D3/def-TZVP). The calculations are performed by Stefan Niebling, University of Osnabrück, and are included in his PhD-thesis. All but the first calculation (gaseous phase) were performed with the COSMO-solvent model (water).

#(water molecules)	in-in	out-in	in-out	out-out
0 (gaseous phase)	344	304	12	0
0 (COSMO)	26	13	11	0
2 (COSMO)	6	13	0	0
4 (COSMO)	10	2	0	3
8 (COSMO)	14	0	5	17
16 (COSMO)	17	19	0	17
32 (COSMO)	16	0	16	44

schemes, direct solvation and step-wise solvation, shows that the step-wise solvation delivers structures with a slightly lower energy which is in accordance to the study of the solvation of chignolin (subsection 5.4.1). The search always started from an out-in conformation at the CBS-unity. With 10 and 20 water molecules, the conformation switched to an out-out conformation in both solvation procedures. In the direct solvation (where all water molecules are added at once), the conformation did not switch with 30 and 40 water molecules. This implies, that the out-in conformation might be more stable in solution than the out-out conformation. The preserving of the out-out conformation in the step-wise solvation seems reasonable as the first 20 water molecules are already optimized for this situation. The following water molecules arrange around the new system. As rearrangements of the water molecules represent the more modest movements in comparison to a conversion of out-out to out-in, the Tabu-Search tends to optimize the solvent shell instead of optimizing the inner solute. This is also in accordance to the results from subsection 5.4.1.



(a) Infra red spectra of CBS-amide in out-out conformation.

(b) Resonance Raman spectra of CBS-amide in out-out conformation.

Figure 5.21: Vibrational spectra of CBS-amide in out-out conformation. The experimental spectra is shown on top. The calculated spectra with different amount of water molecules are shown below. The calculations have been performed with RI-PBE0-D3/def2-TZVP. Solvent effects (all calculations except gas phase calculation) have been included by the COSMO solvent model for water. Regions with the best agreement to the experiment are shown in green. All experiments and calculations of spectral data have been performed by Stefan Niebling, University of Osnabrück, who also provided the figures.

Table 5.16: Solvation of CBS-KKF with different number of water molecules. Energies are given in kcal/mol. [a]All water molecules are added at once and the system is optimized with Tabu-Search. [b]The step-wise approach described in section 4.3 was applied. The result of the previous solvation step was taken as start point for the next solvation.

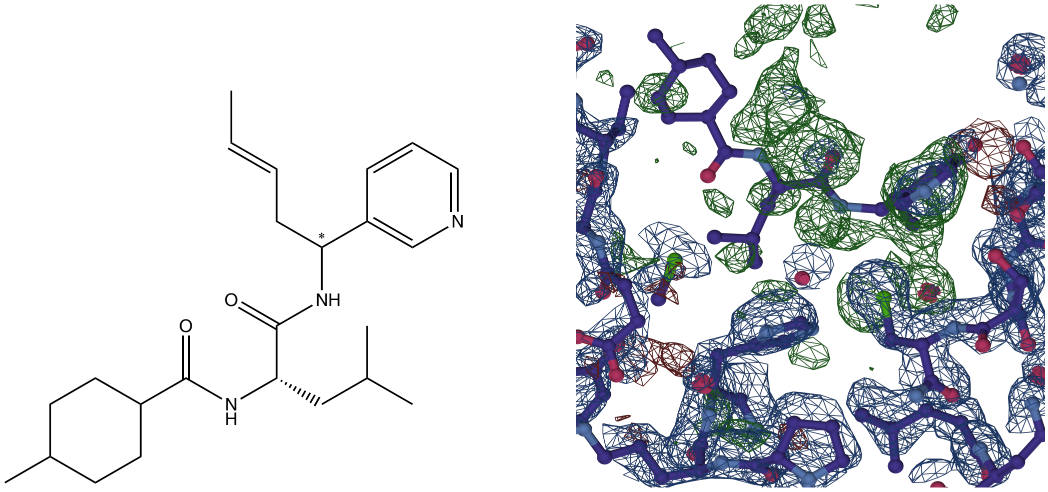
#(water molecules)	direct solvation[a]	conformation	step-wise[b]	conformation
10	-314	out-out	-	
20	-432	out-out	-434	out-out
30	-549	out-in	-552	out-out
40	-672	out-in	-679	out-out

5.5 Molecular Modeling and X-Ray refinement

The investigation and proper description of the orientation of a ligand in the active site of an enzyme is of utmost importance for the optimization of drug molecules and the exploration of reaction mechanisms. Furthermore, important information about experimental data can be obtained. One first application example of the methodology described in subsection 4.1.8 is the optimization of the ligand orientation within the active site of the SARS-CoV M^{pro} (Severe Acute Respiratory Syndrome-corona virus main protease) to provide a better description of the x-ray structure. Further applications deal with the optimization of non-covalent ligand-enzyme complexes of Rhodesain done in the bachelor thesis of Charlotte Brückner.[375]

5.5.1 Global optimization of TS174 in the active site of SARS-CoV M^{pro}

Motivation The inhibitor TS174 (**12**) (see Figure 5.22-a) was crystallized with the SARS-CoV M^{pro} (done by Uwe Dietzel, University of Wuerzburg). While the resolution of the enzyme is very good, the resolution of the ligand is much worse. An excerpt of the electron density can be found in Figure 5.22-b. From experimental results, it is proven that the ligand is binding to the enzyme. However, the exact orientation in the active site is unclear. During crystallization, the ligand was used as a racemate (see stereo center marked with an asterisks in Figure 5.22-a). Therefore, it is further not known which of the stereo isomers is bound. To obtain insights into this problematic and to explain the blurry electron density of the ligand, global optimizations based on Tabu-Search were performed. The results were also compared with MD-simulations of the concerning systems. The experimental data was provided by Uwe Dietzel (Kisker group, University of Wuerzburg) and the MD-simulations were performed by Alexander Paasche (Engels group, University of Wuerzburg).



(a) Inhibitor TS174, (**12**).

(b) Electron density of the active site of SARS-CoV M^{pro} with the inhibitor (**12**).

Figure 5.22: Inhibitor TS174 and its electron density in the active site of SARS-CoV M^{pro}.

Results and discussion The diffuse electron density leads to several structures with a reasonable refinement. Furthermore, the refined electron density looks very different depending on the orientation of the ligand. Therefore, it is very hard to impossible to determine the exact orientation of the ligand without further support by theoretical calculations. All in all, 12 structures with *S*-configuration and 5 with *R*-configuration have been proposed.

The Tabu-Search algorithm is able to create new and reasonable orientations within the active site. Therefore, all proposed structures were globally optimized using the Charmm27 force field leading to new orientations. The dynamic stability of an orientation can be investigated by MD-simulations. A smaller deviation from the starting structure during the simulations indicates a more stable structure.

Details about the MD-simulations can be taken from the work of Alexander Paasche. The set up of the MD simulation applied to each structure was:

- Conversion of (x-ray) structure data to Charmm forcefield compatible pdb structure.
- Addition of hydrogen atoms and definition of titratable groups, like histidines by pKa prediction (PropKa[376–379]).
- Solvation of the protein-inhibitor complex in a water sphere of 110 Å diameter.
- Energy minimization of the water shell and the protein-inhibitor complex.
- Slow heating of the system in a MD simulation to a temperature of 310 K with subsequent gradual release of constraints put on the protein structure.
- Equilibration of the system for a time of 1 ns.
- Productive simulation run for 5 ns.
- Analysis of acquired data.

For global optimizations with Tabu-Search, the procedure described in 4.1.8 was used. An 10 Å radius around the ligand was cut out from the complete system. Water molecules were neglected during the global optimization based on previous results obtained by L. Pason[316] and S. Brickel.[315] These results have shown that consideration of the explicit solvent shell mainly leads to a global optimization of the water molecules instead of global optimization of the ligand orientation.

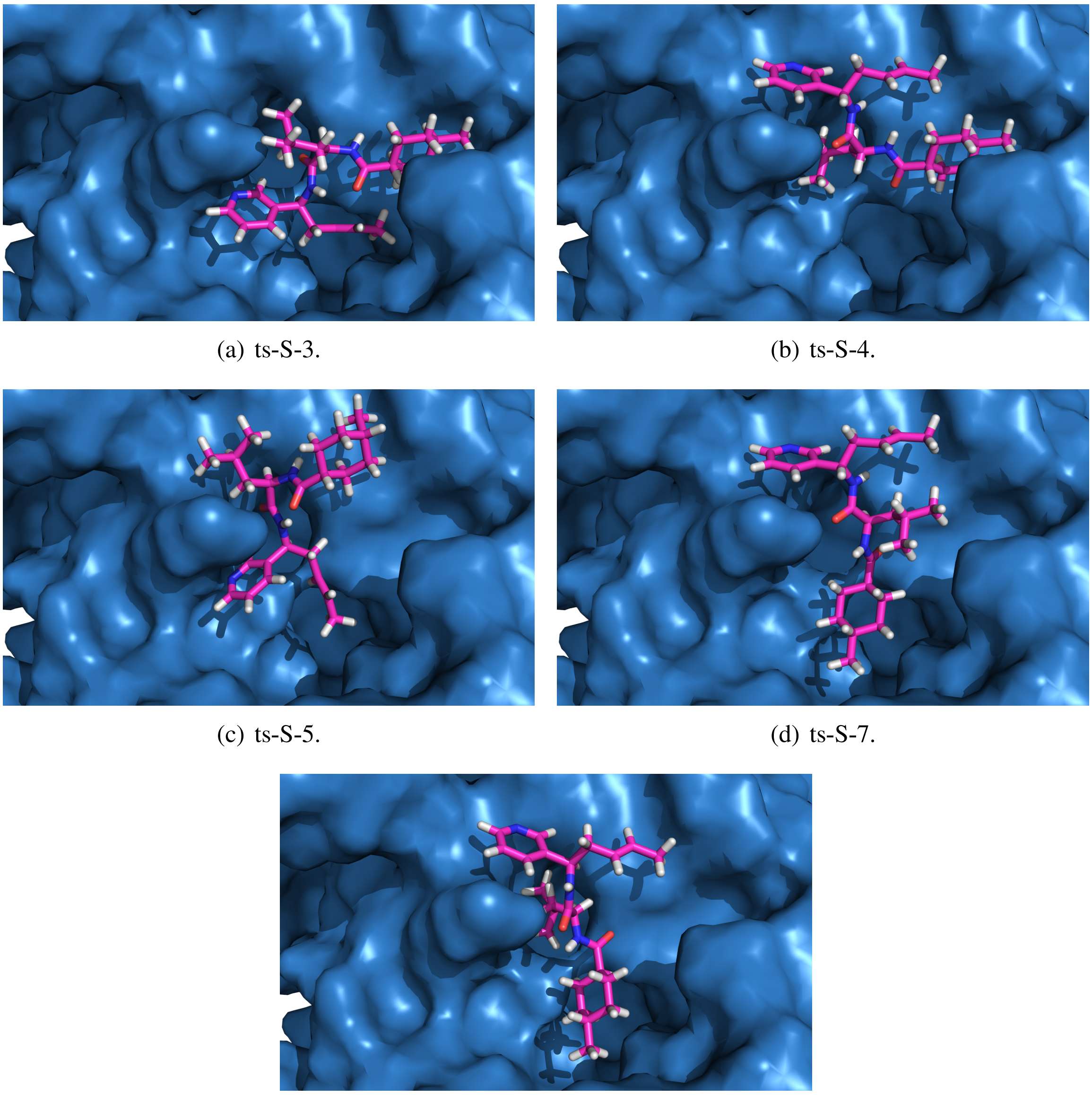
Table 5.17 shows the final energy results from the global optimizations. The values are given relative to the lowest result (S-7). The five best structures were taken and analyzed by a 5 ns MD-simulation like done for all other starting structures. The best results (gray in Table 5.17) were chosen by energy and diversity (i.e. too similar structure were not included twice).

Table A.6 shows the RMSD values from the MD-simulations of the starting structures, while Table 5.18 gives the RMSD values of the five best solutions from global optimization.

Table 5.19 shows the summarized results from the Tabu-Search calculations, the MD-simulations, and the experimental results. Here, the first column gives the relative energies of the Tabu-Search structures and the second column gives the RMSD values of the ligand during a 5 ns MD simulations. The third column gives a rough rating of the quality of the refined

Table 5.17: Final energy results of the Tabu-Search simulation starting from the given structure. Grey structure are further analyzed by MD-simulations. [a]Energies are in kcal/mol and relative to the lowest found result. [b]RMSD values of the ligand with respect to the lowest found conformation.

Structure	Energy[a]	RMSD[b]
S-1	3.7	7.56
S-2	1.6	7.74
S-3	2.0	7.25
S-4	0.1	4.55
S-5	1.6	7.76
S-6	2.0	7.24
S-7	0.0	0.00
S-8	0.1	4.52
S-9	11.5	7.18
S-10	0.2	3.84
S-11	8.5	8.14
S-12	7.6	7.03
R-1	6.6	7.11
R-2	3.3	3.01
R-3	1.5	2.72
R-4	5.1	7.44
R-5	12.9	8.51



(a) ts-S-3.

(b) ts-S-4.

(c) ts-S-5.

(d) ts-S-7.

(e) ts-R-3.

Figure 5.23: Structures obtained by Tabu-Search optimization.

electron density. It becomes obvious, that a low RMSD value in the MD-simulations correlates with a good refinement. The structures proposed by Tabu-Search seem to be reasonable as 3 out of five have a good refinement, which means relatively few difference density occurs. Furthermore, new structural arrangements can be created by Tabu-Search. The five structures lie in the active site very diversely. They can be seen in Figure 5.23. The energy difference between the found structures are very small (only about 2.0 kcal/mol). Having in mind, that all calculations have been performed within a force field with certain inaccuracies, all of them are in principle equally probable. Therefore, it is most likely that the diffuse electron density resulted from a superposition of several conformations and stereo isomers. Unfortunately, it is therefore not possible to determine an exact solution of the x-ray structure. Only suggestions of possible structures can be provided.

Table 5.18: Summary of RMSD values for SARS-CoV M^{pro} and inhibitor (**12**) with respect to first simulation frame or Tabu-Search (TS) structure. Averaged RMSD values* for MD simulations of all inhibitor poses. * Protein RMSD values take backbone atoms into account, inhibitor RMSD values all heavy atoms. (1st) = RMSD relative to first frame structure after 1ns equilibration. (TS) = RMSD value relative to minimized starting structure from Tabu-Search.

Structure	protein(1st)	protein(TS)	(**12**) (1st)	(**12**) (TS)
ts-S-3	1.57±0.19	1.50±0.16	1.26±0.31	1.57±0.27
ts-S-4	1.19±0.14	1.24±0.14	2.61±0.90	2.57±0.88
ts-S-5	1.41±0.22	1.40±0.19	5.63±1.54	5.59±1.57
ts-S-7	1.49±0.23	1.48±0.20	4.85±1.25	4.86±1.25
ts-R-3	1.25±0.14	1.26±0.13	1.08±0.26	1.38±0.25

Table 5.19: Summary of the results of Tabu-Search, MD-simulations, and experiment. ^{a} RMSD value from a 5 ns MD-simulation relative to the first snapshot (i Å). ^{b} Rating of the quality of the structure refinement with the experimental electron density.

Structure	$\Delta(E)$ Tabu-Search [kcal/mol]	RMSD of the ligand^a	Quality of refinement ^{b}
ts-S-3	2.0	1.26±0.31	very good
ts-S-4	0.1	2.61±0.90	good
ts-S-5	1.6	5.63±1.54	very bad
ts-S-7	0.0	4.85±1.25	bad
ts-R-3	1.5	1.08±0.26	middle

Despite the very good results one should keep in mind, that during the global optimizations no explicit solvent molecules or ions have been taken into account. These may have a strong influence on the orientation of the ligand. Future applications should also include such effects and it has to be checked for each case whether such effects are important or not.

5.6 Reaction pathways for argon clusters

As already described in subsection 5.1, Lennard-Jones (LJ) clusters represent ideal benchmark systems. Subsection 5.1 investigated the performance of Tabu-Search for global optimization of different LJ-clusters. In contrast, the following section discusses reaction pathways of selected clusters. The pathways are generated with different approaches to investigate their performance. In the following, the transitions of the Lennard-Jones clusters Ar_{12} and Ar_{13} (employing the OPLS-AA parameters of argon) are investigated. In the beginning, the conversion of the global minimum of Ar_{12} into another higher lying minimum was examined using the NEB method described in section 3.4.3 and 4.1.7. However, such systems are often very flexible and a lot of different pathways may exist. The NEB method can only locate the closest pathway. Therefore, the same system is later on used to benchmark the newly developed PathOpt algorithm (subsection 4.4). In addition, Ar_{13} is taken as one further example application.

5.6.1 Pathways obtained by the NEB method

Procedure The NEB search was initialized by an end and a start point using 30 images in total. After the NEB was relaxed, each point was locally optimized and clustered to the start point, the end point or new minima. The new minima were used for further NEB runs employing the same conditions. This procedure is therefore an iterative approach which tries to find the missing links between start and end points. The Ar_{12} system was taken as a test candidate.

Ar_{12} Using the procedure described above, the Ar_{12} cluster was investigated. The start point was set to the global minimum with an energy of -8.88 kcal/mol (within the OPLS-AA force field). The end point was the slightly higher lying minimum with an energy of -8.48 kcal/mol. Application to proper aligned structures yielded the path shown in Figure 5.24. Two argon atoms are moving simultaneously with one intermediate minimum.

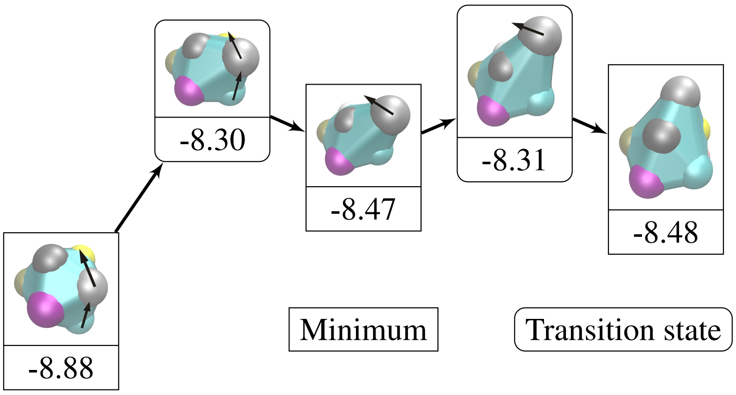


Figure 5.24: Transition path for Ar_{12} obtained by the NEB method.

5.6.2 Pathways obtained by the PathOpt algorithm

The performance of PathOpt algorithm described in subsection 4.4 was tested at hand of the Ar_{12} and the Ar_{13} clusters. PathOpt is based on a global optimization in a hyperplane standing perpendicular to the reaction coordinate. Minima of this reduced hyperplane represent traces to transition states between reactant and product state. Therefore, the procedure includes an optimization of a given point to the closest transition state which was employed by application of the Dimer-method. The initial dimer is created by either a random vector or a calculated vibrational mode. In case of the Ar_{12}, two different approaches have been tested: taking the largest imaginary frequency and taking the smallest imaginary frequency of the starting point for creating the dimer. However, the results have shown that the PathOpt algorithm usually delivers only one imaginary frequency which is significantly different to zero. All other imaginary frequencies (if existing) are close to zero and belong to either rotational or translational movements. Therefore, in the following always the largest imaginary frequency is used. Taking random modes for initializing the dimer also delivers first order saddle points, however, the lowest one is not always found.

Computational details All calculations (except frequency analysis) were performed with the CAST program. Local optimizations are performed with an L-BFGS algorithm using a gradient norm of 0.00001 kcal mol^{-1} Å^{-1}. For the Dimer-method, a gradient norm of 0.001 kcal mol^{-1} Å^{-1} was used. Force field calculations were performed taking the argon parameter of OPLS-AA.[233,336,337] Vibrational analysis was performed with the *vibrate* module of Tinker 5.1.[380] Structure alignments were done using the VMD program.[324] In both test cases, using the Ar_{12} and the Ar_{13} systems, a conversion of the global minimum to a slightly higher lying minimum was investigated. For each BH simulation, 200 optimization steps at 200 K have been performed.

Ar_{12} The Ar_{12} cluster was taken as a first benchmark system. One pathway already had been located with the NEB approach described in the previous subsection. The initial state was again set to the global minimum of Ar_{12} possessing an energy of -8.88 kcal mol^{-1} within the OPLS-AA force field. The final state was set to the minimum with an energy of -8.48 kcal mol^{-1}. For the first test, the search was restricted to one perpendicular plane in the middle of the initial path and only one PathOpt iteration. Optimization of the results of PathOpt delivered several transition states. The results with a classification of the transition are summarized in Table 5.20. After sampling the perpendicular hyperplane, optimization of all resulting points to the closest transition state, determination of the connected minima, and connecting the end points (Min1 and Min2 in Table 5.20) to the initial and final state (i.e. all in all one PathOpt iteration), three complete paths have been found which are shown in Figure 5.25 to 5.27.

The first path from Figure 5.25 only contained one intermediate minimum. The two argon

Table 5.20: Results of the PathOpt run for Ar_{12}. [a] Accepted point from global optimization in perpendicular hyperplane. [b] Connected minima from optimization along the imaginary frequency of the TS. [c] Transition state resulting from the point obtained by PathOpt. [d] Classification: "no connection": neither the initial nor the final state is connected; "rearrangement": same symmetry but different atom numbering; "fragment": fragment either connected to initial or final state, but second iteration would be needed; "TS too high": cluster destroyed in TS.

PathOpt point[a]	Min1[b]	TS[c]	Min2[b]	classification[d]
1	-8.37	**-8.34**	-8.47	delivered Path 2 (Figure 5.26), similar to NEB
2	-8.18	-7.96	-8.88	no connection
3	-8.88	-8.51	-8.88	rearrangement
4	-8.88	-8.63	-8.88	rearrangement
5	-8.88	-8.39	-8.88	rearrangement
6	-8.49	-8.17	-8.88	fragment
7	-8.88	-8.63	-8.88	rearrangement
8	-8.88	-8.63	-8.88	rearrangement
9	-8.88	**-7.96**	-8.19	delivered Path 1 (Figure 5.25)
10	-8.20	-8.19	-8.88	no connection
11	-8.37	-8.00	-8.24	fragment
12	-8.46	-8.16	-8.46	fragment
13	-8.88	-8.63	-8.88	rearrangement
14	-8.47	**-8.31**	-8.46	delivered Path 3 (Figure 5.27)
15	-8.48	-8.31	-8.47	no connection
16	-8.47	-8.31	-8.46	same as point 14
17	-8.47	-8.30	-8.46	fragment,
18	-8.47	-7.67	-8.47	TS too high
19	-8.48	-8.32	-8.48	no connection
20	-8.46	-8.30	-8.47	fragment
21	-8.47	-7.67	-8.47	TS too high
22	-8.88	-7.67	-8.47	TS too high
23	-8.46	-8.04	-8.48	fragment
24	-8.03	-7.88	-7.96	no connection
25	-8.47	-7.67	-8.47	TS too high
26	-8.47	-7.67	-8.47	TS too high

atoms with the largest displacement during the rearrangement (colored in silver and cyan) were moving separately. First, one argon atom takes its position as in the final state leading to the one intermediate minimum. The other argon atom followed via the next transition state. In comparison to the other paths as well as to the path obtained by the NEB approach, this path has a higher activation energy.

The second path from Figure 5.26 is very similar to the path found by the NEB approach. The final transition (minimum with -8.47 kcal/mol with its transition state at -8.31 kcal/mol) is identical with the transition in the NEB path. However, the minimum at -8.47 kcal/mol is reached via another transition with one intermediate minimum. The minima and transition states (minimum: -8.37 kcal/mol, transition states: -8.36 and -8.34 kcal/mol) were slightly lower in energy than the transition state from the NEB path (-8.30 kcal/mol).

The last path (Figure 5.27) comprises simultaneous movements of the argon atoms. The first step and the final step are identical to the ones in the second path. However, the transition

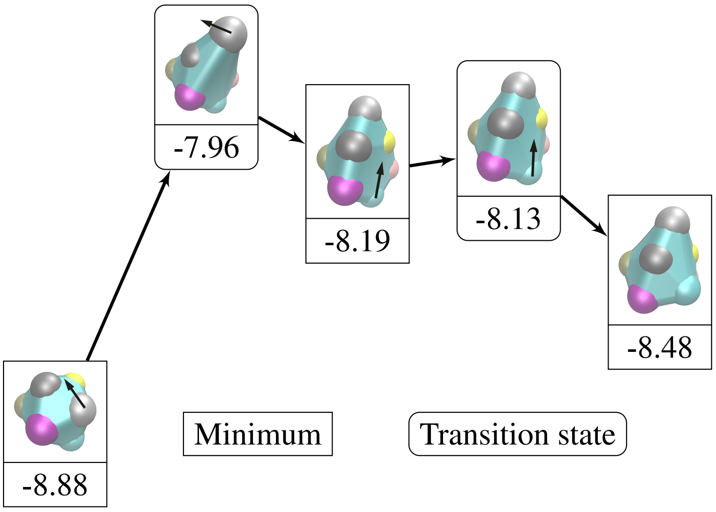


Figure 5.25: First path obtained for Ar_{12} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

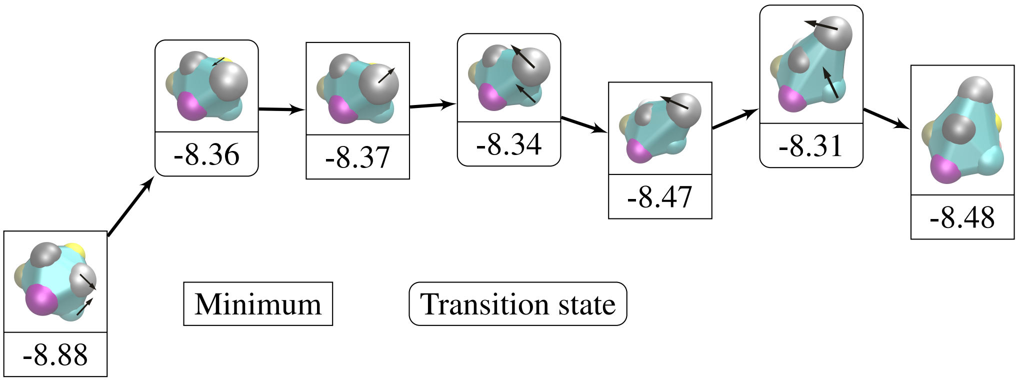


Figure 5.26: Second path obtained for Ar_{12} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

from the minimum with -8.37 kcal/mol to the minimum with -8.47 kcal/mol occurred via another intermediate minimum.

The solutions 3, 4, 5, 7, 8, and 13 (from Table 5.20) consisted of a rearrangement where Min1 and Min2 had the same symmetry but a different atom numbering. Solutions classified as

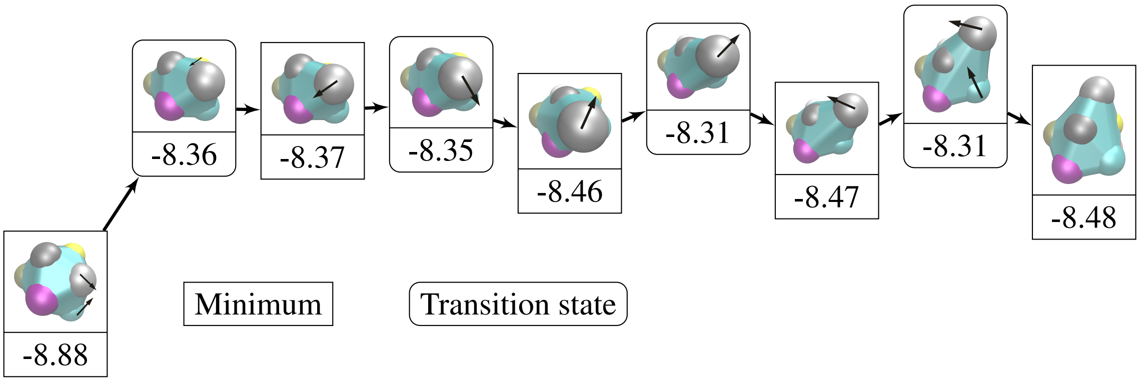


Figure 5.27: Third path obtained for Ar_{12} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

"no connection" delivered a fragment which was not connected to the initial or the final state whereas the solutions classified as "fragment" yielded a path already connected to either the initial or the final state. These points require further PathOpt iterations. During the global optimization, some points were obtained where the TS is too high in energy (i.e. the argon cluster is destroyed). These points could be excluded in future simulations.
A summary of the different paths found for Ar_{12} is given in Table 5.21.

Table 5.21: Results for the pathway investigations for Ar_{12} with the NEB approach and the PathOpt algorithm.

Path	intermediate minima	transition states
NEB Path (Figure 5.24)	1	2
Path 1 (Figure 5.25)	1	2
Path 2 (Figure 5.26)	2	3
Path 3 (Figure 5.27)	3	4

Ar_{13} The Ar_{13} seems to be a little bit more complicated than the Ar_{12} system. The results are summarized in Table 5.22. In contrast to Ar_{12}, the PathOpt search for Ar_{13} delivered mostly rotational transitions. To obtain a third path (Figure 5.28), a second iteration of PathOpt was necessary, between one already found fragment and the final state. However, two paths were already found by one iteration of PathOpt (Figure 5.29 and 5.30). One path was obtained by the procedure described in subsection 4.4 (Figure 5.29). The other path (Figure 5.30) was obtained by subsequently connecting minima, which were found after relaxation of a transition state found by PathOpt. The path shown in Figure 5.29 included a rearrangement of a cluster which already had the right symmetry of the final state, however, the atom numbering was in the wrong order. The third path (Figure 5.30) included two such rearrangements. First, a rearrangement of the global minimum in the same minimum with different atom numbering occurred. After a transition to the final state with the right structure but a wrong atom numbering, one atoms moves around to deliver the final state with the right ordering.

Table 5.22: Results of the PathOpt run for Ar_{13}. All energies in kcal mol^{-1}. [a] Accepted point from global optimization in perpendicular hyperplane. [b] Connected minima from optimization along the imaginary frequency of the TS. [c] Transition state resulting from the point obtained by PathOpt. [d] Classification: "delivered path": the point of PathOpt delivered on complete path; "rotation": rotational translation with an imaginary frequency close to zero.

PathOpt point[a]	Min1[b]	TS[c]	Min2[b]	classification[d]
1	-10.37	**-9.46**	-9.70	delivered Path 2 (Figure 5.29)
2	-9.19	-9.19	-9.19	rotation
3	-10.37	-10.37	-10.37	rotation, GM
4	-9.70	-9.49	-9.70	Min1=Min2
5	-9.70	-9.70	-9.70	rotation
6	-10.37	**-10.37**	-10.37	rotation, GM with different connectivity; delivered Path 3 (Figure 5.30) after connection to initial and final state. Found 14 times
20	-10.37	-10.37	-10.37	rotation, GM with different connectivity; found 7 times
27	-9.68	-9.68	-9.68	rotation
28	-9.27	-9.27	-9.27	rotation
29	-9.51	-9.51	-9.51	rotation
30	-9.50	-9.50	-9.50	rotation
31	-10.37	-10.37	-10.37	rotation
32	-9.38	-9.38	-9.38	rotation
33	-9.44	-9.44	-9.44	rotation
34	-10.37	-10.37	-10.37	rotation
35	-10.37	-10.37	-10.37	rotation, found 3 times
38	-10.37	-10.37	-10.37	rotation, found 9 times

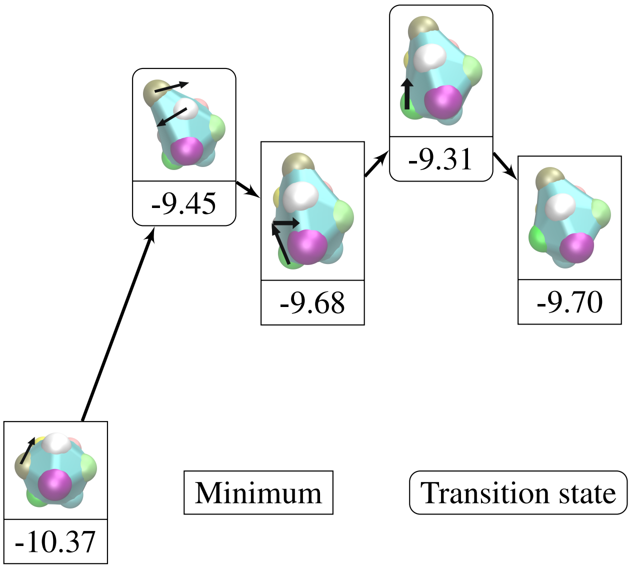


Figure 5.28: First path obtained for Ar_{13} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

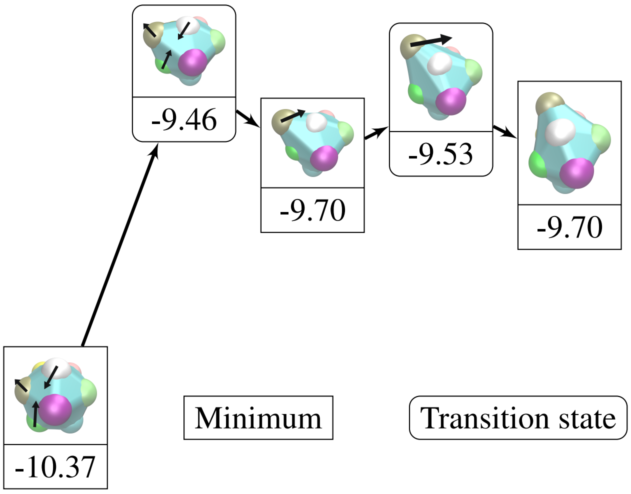


Figure 5.29: Second path obtained for Ar_{13} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

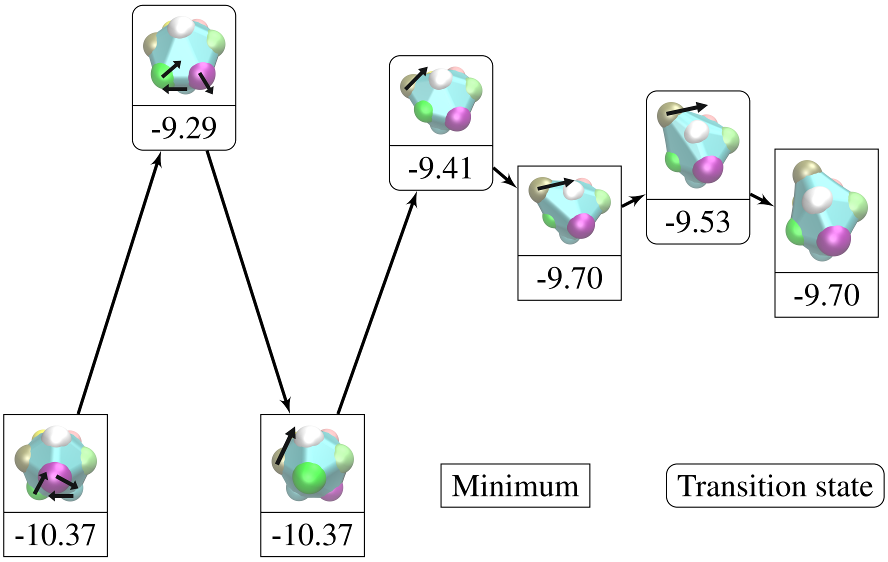


Figure 5.30: Third path obtained for Ar_{13} using the PathOpt algorithm. Argon atoms are colored due to their atomic indices. The largest displacements are indicated by arrows.

Conclusions and Outlook The results obtained for Ar_{12} using the PathOpt algorithm show the strengths of the new approach. With very few global optimization steps and only one perpendicular hyperplane, already three different pathways have been located. The Ar_{13} seems to be a little bit more complicated. Here, one single application of PathOpt delivered only one complete pathway. However, by further analysis of the obtained fragments further paths have been found. The applied methodology (taking two minima and connecting them via a linear fit and subsequent optimization to the closest transition state) is very similar to the discrete path sampling (DPS) approach proposed by Wales.[210] In the DPS approach, an initial data base consisting of all low lying minima is created. In contrast, the PathOpt algorithm directly samples the most interesting phase space (i.e. the space lying between reactant and product state). Therefore, the data base might be created more efficiently, which can also improve the DPS approach. PathOpt should be further improved by the utilization of several perpendicular hyperplanes. Thus, the phase space will be searched more accurately. As several structures are found which are quite similar, a clustering of minima before further analysis would be advantageous. This is also a key step in the DPS approach. The investigations also revealed that the output of PathOpt contains several transition points which belong to rotational or translational movements. However, these points can be recognized very easily and rejected during the global optimization. This further accelerates the method as less transition states have to be refined. Finally, the results of the previous chapters revealed that the Tabu-Search based algorithms developed in the present work posses a better performance than e.g. Basin Hopping alone. Therefore, the implementation of Tabu-Search into the PathOpt algorithm also might improve the method.

6 Summary

The visualization of energy functions is based on the possibility of separating different degrees of freedom. The most important approximation is the Born-Oppenheimer-approximation, which separates nucleus and electron movements. This allows the illustration of the potential energy as a function of the nuclei coordinates. In principle, the resulting multidimensional hypersurface represents the molecular formula of an arbitrary system. Minima of the surface correspond to stable points like isomers or conformers. They are important for predicting the stability of a state or thermodynamical properties. Stationary points of first order correspond to transition points. They describe, for example, phase transitions, chemical reaction, or conformational changes. Furthermore, the partition function connects the potential hypersurface to the free energy of the system. The aim of the present work is the development and application of new approaches for the efficient exploration of multidimensional hypersurfaces. Initially, the **C**onformational **A**nalysis and **S**earch **T**ool (CAST) - program was developed to create a basis for the new methods and algorithms. The development of CAST in object oriented C++ included, among other things, the implementation of a force field, different interfaces to external programs, analysis tools, and optimization libraries.

Descriptions of an energy landscape require knowledge about the most stable minima. The Gradient Only Tabu Search (GOTS) has been shown to be very efficient in the optimization of mathematical test functions. Therefore, GOTS was taken as a starting point. Tabu-Search is based on the *steepest descent* - *modest ascent* strategy. The *steepest descent* is used for finding local minima, while the *modest ascent* is taken for leaving a minimum quickly. Furthermore, Tabu-Search is combined with an adaptive memory design to avoid cycling or returning. The highly accurate exploration of the phase space by Tabu-Search is often too expensive for complex optimization problems. Therefore, an algorithm for diversification of the search is required. After exploration of the proximity of the search space, the algorithm would guide the search to new and hopefully promising parts of the phase space. First application of GOTS to conformational search revealed weaknesses in the diversification search and the *modest ascent* part. On the one hand, the original methodology for diversification is insufficiently diverse. The algorithm is considerably improved by combining the more local GOTS with the wider searching Basin Hopping (BH) approach improved not only the efficiency, but also the number of necessary steps until finding the global minimum. The second weak point is a too inaccurate and inefficient *modest ascent* strategy. Analysis of common transition state search algorithms lead to the adaption of the Dimer-method to the Tabu-Search approach. The Dimer-method only requires the first derivatives for locating the closest saddle point of first order. For conformational search, dihedral angles are usually the most flexible degrees of freedom. Therefore, only those are used in the Dimer-method for leaving a local minimum. Furthermore, the exact localization of the reaction pathway and the transition state is not necessary as the local minimum position should only be departed as fast as possible. This allows for larger step sizes during the Dimer-search. In the following optimization step, all coordinates are relaxed to remove possible strains in the system. The new Tabu-Search method with Dimer-search delivers more and improved minima. Furthermore, the approach is faster for larger systems. For a system with approximately 1200 atoms, an acceleration of 40 was measured. The new approach was compared to Molecular Dynamics with optimization (MD), Simulated Annealing (SA), and BH with the help of conformational search problems of bio-organic systems. In all cases, a better performance was found. A comparison to the Monte Carlo Multiple Minima / Low Mode Sampling (MCMM/LM) method proved the outstanding performance of the new Tabu-Search approach. The solvation of the chignolin protein further revealed the possibility of uncovering discrepancies between the employed theoretical model and the experimental starting structure. Ligand optimization for improvement of x-ray structures was one further new application field.

Besides the global optimization, the search for transition states and reaction pathways is also of paramount importance. These points describe different transitions of stable states. The Dimer- and the Nudged Elastic Band method have been implemented into the CAST program. However, they are only able to locate the closest transition state or reaction pathway, respectively. In complex systems, often several different transition pathways with many intermediate minima exist. Therefore, a new approach for the exploration of such cases was developed. The new approach is based on a global minimization of a hyperplane being perpendicular to the reaction coordinate. Minima of this reduced phase space belong to traces of transition states between reactant and product states on the unchanged hypersurface. Optimization to the closest transition state using the Dimer-method delivers paths lying between the initial and the final state. An iterative approach finally yields complex reaction pathways with many intermediate local minima. The new PathOpt algorithm was tested by means of rearrangement reactions of argon clusters showing very promising results.

The described implementation of PathOpt employed the Basin Hopping approach for global optimization. In the following step, the performance of the Tabu-Search algorithm in combination with PathOpt can be investigated. Furthermore, an outlook for the future is to investigate the influence of several perpendicular search planes on the performance of the algorithm.

7 Zusammenfassung

Die visuelle Darstellung von Energiefunktionen basiert auf der Möglichkeit, verschiedene Freiheitsgrade voneinander zu separieren. Die wichtigste Näherung ist dabei die Born-Oppenheimer-Näherung, welche die Kernbewegung von der Elektronenbewegung separiert. Sie erlaubt damit die Darstellung der potentiellen Energie als Funktion der Kernkoordinaten. Die daraus entstehende mehrdimensionale Hyperfläche entspricht im Prinzip der Summenformel eines beliebigen Systems. Minima der Fläche entsprechen stabilen Punkten wie Isomeren oder Konformeren. Diese sind wichtig für Aussagen über die Stabilität eines Zustandes oder die Thermodynamik. Stationäre Punkte erster Ordnung entsprechen Übergangsstrukturen und beschreiben zum Beispiel Phasenübergänge, chemische Reaktionen, aber auch Konformationsänderungen. Über die Zustandssumme ist die Hyperfläche zudem mit der freien Energie verknüpft. Das Ziel dieser Arbeit ist die Entwicklung und Anwendung neuer Methoden zur effizienten Untersuchung mehrdimensionaler Hyperflächen. Dabei wurde zunächst das **C**onformational **A**nalysis and **S**earch **T**ool (CAST) - Programm entwickelt, um eine Basis für die neuen Methoden und Algorithmen zu bilden. Die Entwicklung des CAST-Programms in objektorientiertem C++ beinhaltete unter anderem die Implementierung eines Kraftfeldes, verschiedene Schnittstellen zu externen Programmen, Analysealgorithmen und verschiedene Optimierungsmodule.

Um Aussagen über eine Energielandschaft treffen zu können, müssen zuerst die stabilsten Minima gefunden werden. Der Gradient Only Tabu Search (GOTS) hat sich als sehr effizient in der Optimierung von mathematischen Testfunktionen erwiesen. Daher wurde GOTS als Startpunkt für diese Arbeit verwendet. Tabu-Search basiert auf dem *steepest descent* - *modest ascent* Prinzip. Zum Finden neuer Minima wird der steilste Abstieg (*steepest descent*) verwendet, wohingegen ein Minimum auf dem Weg mit dem geringsten Anstieg (*modest ascent*) wieder Verlassen wird. Tabu-Search ist zudem mit einem lernfähigen Speicherdesign kombiniert, wodurch ein Zurück- und im Kreis laufen vermieden wird. Der Phasenraum wird von Tabu-Search sehr genau untersucht, was für komplexere Optimierungsprobleme aber zu aufwendig wird. Daher bedarf es eines Diversifizierungsschritts, welcher nach Absuchen eines Bereichs des Phasenraums, die Suche in neue, hoffentlich vielversprechende Bereiche bringt. Erste Anwendungen auf Konformationssuchen zeigten, dass GOTS Schwächen im Diversifizierungsschritt und der *modest ascent* Strategie besitzt. Zum einen ist die ursprünglich verwendete Methodik für die Diversifizierungssuche zu wenig divers. Eine Kombination des mehr lokalen GOTS mit der deutlich weiträumiger suchenden Basin Hopping (BH) Methode brachte eine erhebliche Verbesserung sowohl in der Effizienz als auch der Anzahl an nötigen Iterationen bis zum Finden des globalen Minimums. Der zweite Schwachpunkt besteht aus einer zu ungenauen und ineffizienten *modest ascent* Methode. Eine Analyse von gängigen Übergangszustand-Suchalgorithmen führte dazu, die Dimer-Methode für den Tabu-Search zu adaptieren. Diese benötigt lediglich die erste Ableitung, um damit zum Übergangszustand erster Ordnung zu konvergieren. Da Diederwinkel zu den am leichtesten veränderbaren Variablen innerhalb der Konformationssuche gehören, werden nur diese in der Dimer-Methode zum Verlassen eines Minimums verwendet. Zudem muss der Reaktionspfad und der Übergangszustand nicht exakt getroffen werden, da das Minimum nur möglichst schnell verlassen werden soll. Dies erlaubt größere Schrittweiten in der Dimer-Suche. Im nachfolgenden Optimierungsschritt werden alle Koordinaten relaxiert und dadurch eventuell auftretende Spannungen gelöst. Die neue Tabu-Search-Methode mit Dimer-Suche liefert mehr und deutlich verbesserte Minima. Zudem ist sie für größere Systeme deutlich schneller. Für ein System mit circa 1200 Atomen wurde eine Beschleunigung um den Faktor 40 erzielt. Die neue Methode wurde am Beispiel der Konformationssuche von bio-organischen Systemen mit Molekular Dynamik Simulationen mit Optimierung (MD), Simulated Annealing (SA) und BH verglichen, wobei sich in allen Fällen eine bessere Effizienz zeigte. Ein Vergleich zur Monte Carlo Multiple Minima / Low Mode Sampling (MCMM/LMOD) Methode anhand der Optimierung von peptidischen Ligand-Rezeptor-Komplexen belegte ebenfalls die hervorragende Effizienz des neuen Tabu-Search-Ansatzes. Die Solvatisierung des Chignolin-Proteins mit Tabu-Search deckte zudem die Möglichkeit auf, Differenzen zwischen der verwendeten theoretischen Methode und der experimentellen Startstruktur aufzudecken. Als weiterer neuer Anwendungsbereich wurde die Optimierung der Ligandorientierung zur Verbesserung von Röntgenstrukturen untersucht.

Neben der globalen Optimierung ist auch die Suche nach Übergangszuständen und Reaktionspfaden von größter Wichtigkeit. Diese beschreiben verschiedene Übergänge zwischen stabilen Zuständen. Die ebenfalls in das CAST-Programm implementierten Algorithmen, die Dimer- und die Nudged Elastic Band-Methode, können nur den nächstgelegenen Übergangszustand beziehungsweise Reaktionspfad finden. Bei komplexeren Systemen liegen aber oftmals mehrere Pfade mit vielen intermediären Minima vor. Um diese Systeme genauer untersuchen zu können, wurde ein neuer Ansatz entwickelt. Dieser basiert auf einer globalen Minimierung einer Hyperfläche, welche senkrecht zum Reaktionspfad steht. Die Minima dieses reduzierten Phasenraums sind auf der Gesamthyperfläche Spuren zu Übergangszuständen zwischen dem Edukt und dem Produkt-Zustand. Durch Optimierung dieser Punkte mittels der Dimer-Methode werden also Pfade gefunden, die zwischen Anfangs- und Endpunkt liegen. Ein iteratives Vorgehen liefert letztendlich komplexe Reaktionspfade, die über mehrere lokale Minima verlaufen. Der neue PathOpt-Algorithmus wurde anhand von Umlagerungsreaktionen von Argon-Clustern evaluiert, welche sehr vielversprechende Ergebnisse lieferten.

In der vorgestellten Implementierung von PathOpt wurde der Basin Hopping Ansatz für die globale Optimierung verwendet. Im nächsten Schritt kann untersucht werden, ob die Effizienz von PathOpt durch Verwendung von Tabu-Search weiter verbessert wird. Weiterhin wäre von Interesse, inwieweit die Verwendung mehrerer senkrechter Suchebenen Auswirkungen auf die Leistung des neuen Algorithmus zeigt.

References

1 M. Ventresca, *Comput. Oper. Res.* **2012,** *39,* 2763–2775.

2 A. Boubezoul, S. Paris, *Pattern Recogn.* **2012,** *45,* 3676–3686.

3 D. Tolkunov, A. Morozov, *Phys. Rev. Lett.* **2012,** *108,* 1–5.

4 D. Gront, S. Kmiecik, M. Blaszczyk, D. Ekonomiuk, A. Kolinski, *WIREs Comput. Mol. Sci.* **2012,** *2,* 479–493.

5 J. Lee, S. Gross, J. Lee, *Phys. Rev. E* **2012,** *85,* 1–5.

6 D. J. Wales, H. A. Scheraga, *Science* **1999,** *285,* 1368–1372.

7 J. D. Wales, *Energy Landscapes;* Cambridge University Press: United Kingdom, 2003.

8 G. P. Rangaiah, Stochastic Global Optimization. In *Adv. Process Sys. Eng.*, 1 ed.; G. P. Rangaiah, (Ed.), World Scientific: New Jersey, London, Singapore, Beijing, Shanghai, Hong Kong, Taipei, Chennai, 2010.

9 B. Hartke, *WIREs Comput. Mol. Sci.* **2011,** *1,* 879–887.

10 J. B. Robinson, *RAND Research Memorandum;* RAND Research Memorandum RM-303: Santa Monica, CA, 1949.

11 J. B. J. Kruskal, *P. Am. Math. Soc.* **1956,** *7,* 48–50.

12 J. D. Bryngelson, P. G. Wolynes, *P. Natl. Acad. Sci. USA* **1987,** *84,* 7524–7528.

13 S. S. Cho, P. Weinkam, P. G. Wolynes, *P. Natl. Acad. Sci. USA* **2008,** *105,* 118–123.

14 B. Alatas, *Expert Syst. Appl.* **2012,** *39,* 11080–11088.

15 K. D. Gibson, H. A. Scheraga, THE MULTIPLE-MINIMA PROBLEM IN PROTEIN FOLDING. In *Structure and Expression: Vol. I. From Proteins to Ribosome*; M. H. Sarma, R. H. Sarma, (Eds.), Adenine Press: New York, 1988.

16 H. A. Scheraga, *Int. J. Quantum Chem.* **1992,** *42,* 1529–1536.

17 T. Weise, *Global Optimization Algorithms - Theory and Application;* e-book: http://www.it-weise.de: 2009-06-26 ed.; 2009.

18 J. Brownlee, *Clever Algorithms: Nature-Inspired Programming Recipes;* Lulu Enterprises: 1 ed.; 2011.

19 F. Jensen, *Introduction to Computational Chemistry;* John Wiley and Sons: second ed.; 2006.

20 F. Glover, *Comput. Oper. Res.* **1986,** *13,* 533–549.

21 T. C. Schmidt, A. Paasche, C. Grebner, K. Ansorg, J. Becker, W. Lee, B. Engels, QM / MM Investigations Of Organic Chemistry Oriented Questions. In *Top. Curr. Chem.*; 2012.

22 Z. Q. Li, K. E. Laidig, V. Daggett, *J. Comput. Chem.* **1998,** *19,* 60–70.

23 S. Goedecker, "Minima hopping: Searching for the global minimum of the potential energy surface of complex molecular systems without invoking thermodynamics", Technical Report, 2004.

24 D. A. C. Beck, V. Daggett, *Methods* **2004,** *34,* 112–120.

25 S. R. Wilson, W. Cui, J. W. Moskowitz, K. E. Schmidt, *J. Comput. Chem.* **1991,** *12,* 342–349.

26 L. B. Morales, R. Gardunojuarez, D. Romero, *J. Biomol. Struct. Dyn.* **1991,** *8,* 721–735.

27 G. Chang, W. C. Guida, W. C. Still, *J. Am. Chem. Soc.* **1989,** *111,* 4379–4386.

28 H. Pohlheim, *Evolutionäre Algorithmen;* Springer-Verlag: Berlin, Heidelberg, New York, Barcelona, Hongkong, London, Mailand, Paris, Singapur, Tokio, 1999.

29 C. W. Reynolds, *Comp. Graph.* **1987,** *21,* 25–34.

30 J. Kennedy, R. Eberhart, Particle swarm optimization. In *Proceedings of ICNN'95*, Vol. 4; IEEE: 1995.

31 M. Dorigo, V. Maniezzo, A. Colorni, *IEEE T. Syst. Man. Cy. B* **1996,** *26,* 29–41.

32 M. Dorigo, L. M. Gambardella, *Biosystems* **1997,** *43,* 73–81.

33 M. Dorigo, G. D. Caro, L. M. Gambardella, *Artificial Life* **1999,** *5,* 137–172.

34 L. M. Gambardella, E. D. Taillard, M. Dorigo, *J. Oper. Res. Soc.* **1999,** *50,* 167–176.

35 D. Karaboga, "An idea based on Honey Bee Swarm for numerical optimization", Technical Report, Erices University, Engineering Faculty, Computer Engineering Department, Kayseri/Turkey, 2005.

36 L. Piela, J. Kostrowicki, H. A. Scheraga, *J. Phys. Chem.* **1989,** *93,* 3339–3346.

37 R. V. Pappu, R. K. Hart, J. W. Ponder, *J. Phys. Chem. B* **1998,** *102,* 9725–9742.

38 R. V. Pappu, G. R. Marshall, J. W. Ponder, *Nat. Struct. Biol.* **1999,** *6,* 50–5.

39 M. Saunders, K. N. Houk, Y. D. Wu, W. C. Still, M. Lipton, G. Chang, W. C. Guida, *J. Am. Chem. Soc.* **1990,** *112,* 1419–1427.

40 I. Kolossváry, W. C. Guida, *J. Am. Chem. Soc.* **1996,** *118,* 5011–5019.

41 I. Kolossváry, W. C. Guida, *J. Comput. Chem.* **1999,** *20,* 1671–1684.

42 F. Glover, *Decision Sci.* **1977,** *8,* 156–166.

43 F. Glover, *INFORMS J. Comput.* **1989,** *1,* 190–206.

44 F. Glover, *INFORMS J. Comput.* **1990,** *2,* 4–32.

45 J. Lee, H. A. Scheraga, S. Rackovsky, *J. Comput. Chem.* **1997,** *18,* 1222–1232.

46 C. Grebner, *New Approach in Conformational Search,* Diploma thesis, University of Wuerzburg, 2009.

47 J. Becker, *Algorithm for the assignment of protein structures as the starting point for the optimization of small peptides with unusual amino acid sequence,* Diploma thesis, University of Wuerzburg, 2010.

48 D. Weber, *The SolvAdd Program - Conformational sampling of hydration shells based in the formation of hydrogen bonds,* Diploma thesis, University of Wuerzburg, 2011.

49 C. J. Cramer, *Essentials of Computational Chemistry;* John Wiley and Sons Ltd.: West Sussex, Second edi ed.; 2004.

50 O. M. Becker, A. D. MacKerrel Jr., B. Roux, M. Watanabe, *Computational Biochemistry and Biophysics;* Marcel Dekker Inc.: New York, 2001.

51 A. R. Leach, *Molecular Modelling - Principle and Application;* Pearson Education Limited: second ed.; 2001.

52 C. Grebner, J. Becker, S. Stepanenko, B. Engels, *J. Comput. Chem.* **2011,** *32,* 2245–2253.

53 T. Huber, A. E. Torda, W. F. van Gunsteren, *J. Comput. Aided Mol. Des.* **1994,** *8,* 695–708.

54 C. Darwin, *On the origin of species by means of natural selection, or the Preservation of Favoured Races in the Struggle for Life;* John Murray: London, 1859.

55 W. Paszkowicz, K. D. Harris, R. L. Johnston, *Comp. Mater. Sci.* **2009,** *45,* ix–x.

56 R. Ismail, R. L. Johnston, *Phys. Chem. Chem. Phys.* **2010,** *12,* 8607–8619.

57 S. Nunez, R. L. Johnston, *J. Phys. Chem. C* **2010,** *114,* 13255–13266.

58 M. T. Oakley, D. J. Wales, R. L. Johnston, *J. Phys. Chem. B* **2011,** *115,* 11525–9.

59 D. T. Tran, R. L. Johnston, *P. Roy. Soc. A-Math. Phy.* **2011,** *467,* 2004–2019.

60 A. J. Logsdail, Z. Y. Li, R. L. Johnston, *J. Comput. Chem.* **2012,** *33,* 391–400.

61 S.-C. Su, C.-J. Lin, C.-K. Ting, *Proteome Sci.* **2011,** *9 Suppl 1,* S19.

62 P. Comte, S. Vassiliev, S. Houghten, D. Bruce, *Biosystems* **2011,** *105,* 263–70.

63 M. C. V. Benitez, R. S. Parpinelli, H. S. Lopes, *Concurr. Comp.-Pract. E.* **2012,** *24,* 635–646.

64 G. Sindhu, S. Sudha, Prediction of Protein Tertiary Structure Using Genetic Algorithm. In *Soft Computing Techniques in Vision Science*, Vol. 395; S. Patnaik, Y.-M. Yang, (Eds.), Springer Berlin Heidelberg: Berlin, Heidelberg, 2012.

65 X. Geng, J. Guan, Q. Dong, S. Zhou, *Int. J. Data Min. Bioin.* **2012,** *6,* 162–177.

66 Borg (Star Trek) - Wikipedia, the free encyclopedia; accessed 07/08/12 http://en.wikipedia.org/wiki/Borg_%2528Star_Trek%2529#Borg_Collective, 2012.

67 G. Beni, J. Wang, *Proceeding of NATO Advanced Workshop on Robots and Biological Systems* **1989,** *102,*.

68 Q. Bai, *Comp. Inf. Sci.* **1998,** *3,* 180–184.

69 H.-T. Yau, T.-H. Hung, C.-C. Hsieh, *Sensors* **2012,** *12,* 7468–7484.

70 A. Bieler, K. Altwegg, L. Hofer, A. Jäckel, A. Riedo, T. Sémon, P. Wahlström, P. Wurz, *J. Mass. Spectrom.* **2011,** *46,* 1143–1151.

71 J. Prasad, T. Souradeep, *Phys. Rev. D* **2012,** *85,* 1–13.

72 W. B. Park, N. Shin, K.-P. Hong, M. Pyo, K.-S. Sohn, *Adv. Func. Mater.* **2012,** *22,* 2258–2266.

73 S. Gajawada, D. Toshniwal, *Procedia Technology* **2012,** *4,* 360–364.

74 F. Marini, B. Walczak, *J. Chemometr.* **2011,** *25,* 366–374.

75 V. Namasivayam, J. Bajorath, *J. Chem. Inf. Model.* **2012,** *52,* 927–934.

76 V. Namasivayam, P. Iyer, J. Bajorath, *Chem. Biol. Drug Des.* **2012,** *79,* 22–9.

77 V. Namasivayam, P. Iyer, J. Bajorath, *J. Chem. Inf. Model.* **2011,** *51,* 1545–51.

78 J.-H. Wen, K.-J. Zhong, L.-J. Tang, J.-H. Jiang, H.-L. Wu, G.-L. Shen, R.-Q. Yu, *Talanta* **2011,** *84,* 13–8.

79 Z. Cheng, Y. Zhang, C. Zhou, *Chem. Biol. Drug Des.* **2011,** *78,* 948–59.

80 K. Tashkova, P. Korošec, J. Silc, L. Todorovski, S. Džeroski, *BMC Syst. Biol.* **2011,** *5,* 159.

81 A. Bonilla-Petriciolet, J. G. Segovia-Hernández, *Fluid Phase Equilibr.* **2010,** *289,* 110–121.

82 J. A. Lazzús, A. A. Pérez Ponce, L. O. Palma Chilla, *Fluid Phase Equilibr.* **2012,** *317,* 132–139.

83 H. Zhang, J. A. Fernández-Vargas, G. P. Rangaiah, A. Bonilla-Petriciolet, J. G. Segovia-Hernández, *Fluid Phase Equilibr.* **2011,** *310,* 129–141.

84 Y.-N. Liu, H. Dung, H. Zhang, G. Wang, Z. Li, H.-l. Chen, *Chem. Res. Chinese U.* **2011,** *27,* 108–112.

85 L.-Y. Chuang, H.-C. Huang, M.-C. Lin, C.-H. Yang, *PloS one* **2011,** *6,* e21036.

86 Y. Liu, W. Li, R. Ma, *Int. J. Biomath.* **2012,** *05,* 1250044.

87 M. Pippel, M. Scharfe, R. Meier, W. Sippl, *Nachr. Chem.* **2012,** *60,* 565–567.

88 File:Aco branches.svg - Wikipedia, the free encyclopedia; accessed 08/08/12 http://en.wikipedia.org/wiki/File:Aco_branches.svg, 2012.

89 L. Chen, H.-Y. Sun, S. Wang, *Inform. Sciences* **2012,** *199,* 31–42.

90 N. Sreelaja, G. Vijayalakshmi Pai, *Appl. Soft Comput.* **2012,** *12,* 2879–2895.

91 O. Korb, T. Stützle, T. E. Exner, PLANTS: Application of Ant Colony Optimization to Structure-Based Drug Design. In *Ant Colony Optimization and Swarm Intelligence, Lecture Notes in Computer Science, Volume 4150*, Vol. 4150; M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, T. Stützle, (Eds.), Springer Berlin Heidelberg: Berlin, Heidelberg, 2006.

92 O. Korb, T. Stützle, T. E. Exner, *Swarm Intell.* **2007,** *1,* 115–134.

93 O. Korb, T. Stützle, T. E. Exner, *J. Chem. Inf. Model.* **2009,** *49,* 84–96.

94 O. Korb, T. Stützle, T. E. Exner, *J. Chem. Inf. Model.* **2011,** *51,* 865–76.

95 M. Goodarzi, M. P. Freitas, Y. Vander Heyden, *Anal. Chim. Acta* **2011,** *705,* 166–73.

96 F. Hammann, C. Suenderhauf, J. Huwyler, *J. Chem. Inf. Model.* **2011,** *51,* 2690–6.

97 G. Chen, Y. Lu, *Chinese J. Chem.* **2011,** *29,* 2019–2026.

98 F. Abbasitabar, V. Zare-Shahabadi, *SAR QSAR Environ. Res.* **2012,** *23,* 1–15.

99 Z.-C. Li, X. Zhou, Z. Dai, X.-Y. Zou, *Anal. Chim. Acta* **2011,** *703,* 163–71.

100 F. Allegrini, A. C. Olivieri, *Anal. Chim. Acta* **2011,** *699,* 18–25.

101 M. S. Bergholt, W. Zheng, K. Lin, K. Y. Ho, M. Teh, K. G. Yeoh, J. B. Yan So, Z. Huang, *Int. J. Cancer* **2011,** *128,* 2673–80.

102 D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, *Artif. Intell. Rev.* **2012,** .

103 M. Gupta, G. Sharma, *Int. J. Soft Comp. Eng.* **2012,** *1,* 291–296.

104 R. Fonseca, M. Paluszewski, P. Winter, *J. Math. Model. Alg.* **2010,** *9,* 181–194.

105 C.-J. Lin, S.-C. Su, *Int. J. Innov. Comput. I.* **2012,** *8,* 2049–2064.

106 Y. Zhang, L. Wu, *Ad. Electr. Eng. Sys.* **2012,** *1,* 19–23.

107 C. Schiffmann, D. Sebastiani, *J. Chem. Theory Comput.* **2011,** *7,* 1307–1315.

108 A. Abraham, R. K. Jatoth, A. Rajasekhar, *J. Comput. Theor. Nanos.* **2012,** *9,* 249–257.

109 F. H. Stillinger, T. A. Weber, *J. Stat. Phys.* **1988,** *52,* 1429–1445.

110 J. Kostrowicki, L. Piela, *J. Optimiz. Theory App.* **1991,** *69,* 269–284.

111 J. Kostrowicki, L. Piela, B. J. Cherayil, H. A. Scheraga, *J. Phys. Chem.* **1991,** *95,* 4113–4119.

112 T. Head-Gordon, F. H. Stillinger, J. Arrecis, *P. Natl. Acad. Sci. USA* **1991,** *88,* 11076–80.

113 R. J. Wawak, M. M. Wimmer, H. A. Scheraga, *J. Phys. Chem.* **1992,** *96,* 5138–5145.

114 J. Pillardy, L. Piela, *J. Phys. Chem.* **1995,** *99,* 11805–11812.

115 D. J. Wales, J. P. K. Doye, *J. Phys. Chem. A* **1997,** *101,* 5111–5116.

116 R. J. Wawak, J. Pillardy, A. Liwo, K. D. Gibson, H. A. Scheraga, *J. Phys. Chem. A* **1998,** *102,* 2904–2918.

117 R. K. Hart, R. V. Pappu, J. W. Ponder, *J. Comput. Chem.* **2000,** *21,* 531–552.

118 M. Goldstein, E. Fredj, R. B. Gerber, *J. Comput. Chem.* **2011,** *32,* 1785–1800.

119 P. Bandyopadhyay, *Chem. Phys. Lett.* **2010,** *487,* 133–138.

120 Z. Li, H. A. Scheraga, *Proceedings of the National Academy of Sciences of the United States of America* **1987,** *84,* 6611–6615.

121 D. S. Stephenson, G. Binsch, *J. Magn. Reson.* **1980,** *37,* 395–407.

122 A. J. Markvardsen, W. I. F. David, *Acta Crystallogr. A* **2010,** *66,* 591–6.

123 A. Grossfield, J. W. Ponder, “Global Optimization via a Modified Potential Smoothing Kernel”, Technical Report, Washington University School of Medicine, Washington, 2002.

124 J. M. Carr, D. J. Wales, *J. Chem. Phys.* **2005,** *123,* 234901.

125 S. Kazachenko, A. J. Thakkar, *Chem. Phys. Lett.* **2009,** *476,* 120–124.

126 M. C. Prentiss, D. J. Wales, P. G. Wolynes, *J. Chem. Phys.* **2008,** *128,* 225106.

127 B. Strodel, J. W. L. Lee, C. S. Whittleston, D. J. Wales, *J. Am. Chem. Soc.* **2010,** *132,* 13300–13312.

128 B. Strodel, D. J. Wales, *J. Chem. Theory Comput.* **2008,** *4,* 657–672.

129 M. S. Bauer, B. Strodel, S. N. Fejer, E. F. Koslover, D. J. Wales, *J. Chem. Phys.* **2010,** *132,* 054101.

130 S. N. Fejer, D. Chakrabarti, D. J. Wales, *ACS Nano* **2010,** *4,* 219–228.

131 D. Chakrabarti, S. N. Fejer, D. J. Wales, *P. Natl. Acad. Sci. USA* **2009,** *106,* 20164–20167.

132 S. N. Fejer, T. R. James, J. Hernández-Rojas, D. J. Wales, *Phys. Chem. Chem. Phys.* **2009,** *11,* 2098–104.

133 F. Mohamadi, N. G. J. Richards, W. C. Guida, R. Liskamp, M. Lipton, C. Caufield, G. Chang, T. Hendrickson, W. C. Still, *J. Comput. Chem.* **1990,** *11,* 440–467.

134 I. Kolossváry, G. M. Keserû, *J. Comput. Chem.* **2001,** *22,* 21–30.

135 G. M. Keserû, I. Kolossváry, *J. Am. Chem. Soc.* **2001,** *123,* 12708–9.

136 C. Parish, R. Lombardi, K. Sinclair, E. Smith, A. Goldberg, M. Rappleye, M. Dure, *J. Mol. Graph. Model.* **2002,** *21,* 129–50.

137 C. A. Parish, M. Yarger, K. Sinclair, M. Dure, A. Goldberg, *J. Med. Chem.* **2004,** *47,* 4838–4850.

138 S. D. Hillson, E. Smith, M. Zeldin, C. A. Parish, *J. Phys. Chem. A* **2005,** *109,* 8371–8378.

139 A. Szczepanska, J. L. Espartero, A. J. Moreno-Vargas, A. T. Carmona, I. Robina, S. Remmert, C. A. Parish, *J. Org. Chem.* **2007,** *72,* 6776–6785.

140 S. Remmert, C. A. Parish, *J. Comput. Chem.* **2008,** *30,* 992–998.

141 S. Remmert, H. Hollis, C. A. Parish, *Bioorgan. Med. Chem.* **2009,** *17,* 1251–1258.

142 E. B. Wang, C. A. Parish, *J. Org. Chem.* **2010,** *75,* 1582–1588.

143 M. A. Castriciano, A. Romeo, N. Angelini, N. Micali, S. Guccione, L. M. Scolaro, *Photochem. Photobiol.* **2011,** *87,* 292–301.

144 A. P. R. Zabell, C. B. Post, *Proteins* **2002,** *307,* 295–307.

145 D. K. Menyhárd, G. M. Keseru, *J. Mol. Graph. Model.* **2006,** *25,* 363–72.

146 S. Hu, J. M. Pluth, F. A. Cucinotta, *J. Mol. Model.* **2012,** *18,* 2163–2174.

147 M. K. Holloway, P. Hunt, G. B. McGaughey, *Drug Develop. Res.* **2009,** *70,* 70–93.

148 A. A. Samma, C. K. Johnson, S. Song, S. Alvarez, M. Zimmer, *J. Phys. Chem. B* **2010,** *114,* 15362–15369.

149 B. Li, R. Shahid, P. Peshkepija, M. Zimmer, *Chem. Phys.* **2012,** *392,* 143–148.

150 F. Glover, S. Hanafi, *Discrete Appl. Math.* **2002,** *119,* 3–36.

151 J. Cheng, R. Fournier, *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)* **2004,** *112,* 7–15.

152 Z. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly, R. Marti, *INFORMS J. Comput.* **2007,** *19,* 328–340.

153 S. Stepanenko, B. Engels, *J. Comput. Chem.* **2007,** *28,* 601–611.

154 S. Stepanenko, B. Engels, *J. Comput. Chem.* **2008,** *29,* 768–780.

155 S. Stepanenko, B. Engels, *J. Phys. Chem. A* **2009,** *113,* 11699–705.

156 F. Glover, A Template For Scatter Search And Path Relinking. In *Artificial Evolution, Lecture Notes in Computer Science , Vol. 1363*; J. Hao, E. Lutton, E. Ronald, M. Schoenauer, D. Snyers, (Eds.), Springer: 1998.

157 M. Laguna, R. Marti, *Scatter Search - Methodology and Implementations in C;* Kluwer Academic Publishers: Boston, 2003.

158 M. Laguna, R. Marti, Scatter Search. In *Metaheuristic Procedures for Training Neural Networks*, first ed.; E. Alba, R. Marti, (Eds.), Springer Science+Business Media, LLC: New York, 2006.

159 R. Marti, M. Laguna, F. Glover, *Eur. J. Oper. Res.* **2006,** *169,* 359–372.

160 J. Molina, M. Laguna, R. Marti, R. Caballero, *INFORMS J. Comput.* **2007,** *19,* 91–100.

161 R. Baños, C. Gil, J. Reca, J. Martínez, *Comput. Optim. Appl.* **2007,** *42,* 421–441.

162 R. Caballero, M. Laguna, R. Martí, *J. Oper. Res. Soc.* **2009,** *62,* 2034–2046.

163 J. A. Egea, E. Balsa-Canto, G. Garci, J. R. Banga, *Ind. Eng. Chem. Res.* **2009,** *48,* 4388–4401.

164 M. G. C. Resende, C. C. Ribeiro, F. Glover, R. Marti, Scatter Search and Path-Relinking: Fundamentals, Advances, and Applications - Springer. In *Handbook of Metaheuristics*, 2nd ed.; M. Gendreau, J.-Y. Potvin, (Eds.), Springer Science+Business Media, LLC: New York, 2010.

165 J. A. Egea, E. Vazquez, J. R. Banga, R. Marti, *J. Global Opt.* **2007,** *43,* 175–190.

166 O. Cordon, S. Damas, J. Santamaria, R. Marti, *INFORMS J. Comput.* **2008,** *20,* 55–68.

167 R. A. Russell, W.-C. Chiang, *Eur. J. Oper. Res.* **2006,** *169,* 606–622.

168 M. Vásquez, H. A. Scheraga, *Biopolymers* **1985,** *24,* 1437–47.

169 K. D. Gibson, H. A. Scheraga, *J. Comput. Chem.* **1987,** *8,* 826–834.

170 A. A. Rabow, H. A. Scheraga, *Protein Sci.* **1996,** *5,* 1800–1815.

171 J. Lee, H. A. Scheraga, S. Rackovsky, *Biopolymers* **1998,** *46,* 103–115.

172 Y. Lee, A. Liwo, H. A. Scheraga, *P. Natl. Acad. Sci. USA* **1999,** *96,* 2025–2030.

173 J. Lee, H. A. Scheraga, *Int. J. Quantum Chem.* **1999,** *75,* 255–265.

174 J. Lee, D. R. Ripoll, C. Czaplewski, J. Pillardy, W. J. Wedemeyer, H. A. Scheraga, *J. Phys. Chem. B* **2001,** *105,* 7291–7298.

175 J. Lee, I.-H. Lee, J. Lee, *Phys. Rev. Lett.* **2003,** *91,* 1–4.

176 S.-Y. Kim, S. J. Lee, J. Lee, *J. Chem. Phys.* **2003,** *119,* 10274.

177 J. Lee, S.-Y. Kim, K. Joo, I. Kim, J. Lee, *Proteins* **2004,** *56,* 704–14.

178 J. Lee, *J. Korean. Phys. Soc.* **2004,** *45,* 1450–1454.

179 S.-Y. Kim, S. Lee, J. Lee, *Phys. Rev. E* **2005,** *72,* 1–6.

180 K. Lee, C. Czaplewski, S.-Y. Kim, J. Lee, *J. Comput. Chem.* **2005,** *26,* 78–87.

181 M. K. Song, S.-Y. Kim, J. Lee, *Biophys. J.* **2005,** *115,* 201–7.

182 K. Lee, J. Sim, J. Lee, *Proteins* **2005,** *60,* 257–62.

183 J. Lee, K. Joo, S.-Y. Kim, J. Lee, *J. Comput. Chem.* **2008,** *29,* 2479–2484.

184 K. Joo, J. Lee, I. Kim, S. J. Lee, J. Lee, *Biophys. J.* **2008,** *95,* 4813–9.

185 K. Joo, J. Lee, J.-H. Seo, K. Lee, B.-G. Kim, J. Lee, *Proteins* **2009,** *75,* 1010–23.

186 J. Lee, J. Lee, T. N. Sasaki, M. Sasai, C. Seok, J. Lee, *Proteins* **2011,** *79,* 2403–17.

187 H. Park, J. Ko, K. Joo, J. Lee, C. Seok, J. Lee, *Proteins* **2011,** *79,* 2725–34.

188 W.-H. Shin, L. Heo, J. Lee, J. Ko, C. Seok, J. Lee, *J. Comput. Chem.* **2011,** 3226–3232.

189 J. Lee, J. Lee, K. Joo, J. Lee, *Biophys. J.* **2011,** *100,* 217a.

190 G.-R. Lee, W.-H. Shin, H.-B. Park, S.-M. Shin, C.-O. Seok, *B. Kor. Chem. Soc.* **2012,** *33,* 770–774.

191 D. R. Ripoll, H. A. Scheraga, *Biopolymers* **1988,** *27,* 1283–303.

192 D. R. Ripoll, H. A. Scheraga, *J. Protein Chem.* **1989,** *8,* 263–87.

193 CASP9, CASP9 ABSTRACT BOOK Critical Assessment of Techniques for Protein Structure Prediction Ninth Meeting. In *CASP9 - Abstract Book*; 2010.

194 H. B. Schlegel, *WIREs Comput. Mol. Sci.* **2011,** *1,* 780–809.

195 D. A. Evans, D. J. Wales, *J. Chem. Phys.* **2003,** *119,* 9947–9955.

196 C. J. Cerjan, W. H. Miller, *J. Chem. Phys.* **1981,** *75,* 2800–2806.

197 D. J. Wales, *J. Chem. Soc. Faraday T.* **1990,** *86,* 3505–3517.

198 G. Henkelman, H. Jónsson, *J. Chem. Phys.* **1999,** *111,* 7010–7022.

199 A. Heyden, A. T. Bell, F. J. Keil, *J. Chem. Phys.* **2005,** *123,* 224101.

200 J. Kästner, P. Sherwood, *J. Chem. Phys.* **2008,** *128,* 014106.

201 B. Peters, A. Heyden, A. T. Bell, A. Chakraborty, *J. Chem. Phys.* **2004,** *120,* 7877–7886.

202 H. Jónsson, G. Mills, K. W. Jacobsen, Nudged elastic band method for finding minimum energy paths of transitions. In *Classical and Quantum Dynamics in Condensed Phase Simulations - Proceedings of the International School of Physics*; B. J. Berne, G. Ciccotti, D. F. Coker, (Eds.), World Scientific Publishing Co. Pte. Ltd.: Singapore, 1998.

203 G. Henkelman, H. Jónsson, *J. Chem. Phys.* **2000,** *113,* 9978–9985.

204 G. Henkelman, B. P. Uberuaga, H. Jónsson, *J. Chem. Phys.* **2000,** *113,* 9901–9904.

205 S. A. Trygubenko, D. J. Wales, *J. Chem. Phys.* **2004,** *120,* 2082–2094.

206 I. F. Galván, M. J. Field, *J. Comput. Chem.* **2008,** *29,* 139–143.

207 J. M. Carr, S. A. Trygubenko, D. J. Wales, *J. Chem. Phys.* **2005,** *122,* 234903.

208 L. P. Kadanoff, *Phys. Today* **2001,** *54,* 34–39.

209 C. Dellago, P. G. Bolhuis, D. Chandler, *J. Chem. Phys.* **1998,** *108,* 9236–9245.

210 D. J. Wales, *Mol. Phys.* **2002,** *100,* 3285–3305.

211 D. J. Wales, *Mol. Phys.* **2004,** *102,* 891–908.

212 R. A. Mata, H.-J. Werner, S. Thiel, W. Thiel, *J. Chem. Phys.* **2008,** *128,* 025104.

213 J. M. Dieterich, H.-J. Werner, R. A. Mata, S. Metz, W. Thiel, *J. Chem. Phys.* **2010,** *132,* 035101.

214 A. Laio, M. Parrinello, *P. Natl. Acad. Sci. USA* **2002,** *99,* 12562–12566.

215 D. Zagorac, J. C. Schön, M. Jansen, *J. Phys. Chem. C* **2012,** *116,* 16726–16739.

216 C. Dellago, P. G. Bolhuis, F. S. Csajka, D. Chandler, *J. Chem. Phys.* **1998,** *108,* 1964–1977.

217 C. Dellago, P. G. Bolhuis, P. L. Geissler, *Adv. Chem. Phys.* **2002,** *123,* 1–84.

218 P. G. Bolhuis, D. Chandler, C. Dellago, P. L. Geissler, *Annu. Rev. Phys. Chem.* **2002,** *53,* 291–318.

219 L. J. Munro, D. J. Wales, *Phys. Rev. B* **1999,** *59,* 3969–3980.

220 Y. Kumeda, D. J. Wales, L. J. Munro, *Chem. Phys. Lett.* **2001,** *341,* 185–194.

221 J. M. Carr, D. J. Wales, *J. Phys. Chem. B* **2008,** *112,* 8760–8769.

222 M. C. Prentiss, D. J. Wales, P. G. Wolynes, *PLoS Comput. Biol.* **2010,** *6,* e1000835.

223 D. Passerone, M. Parrinello, *Phys. Rev. Lett.* **2001,** *87,* 1–4.

224 M. Iannuzzi, A. Laio, M. Parrinello, *Phys. Rev. Lett.* **2003,** *90,* 238302.

225 H. Grubmüller, *Phys. Rev. B* **1995,** *52,* 2893–2906.

226 A. Voter, *Phys. Rev. B* **1998,** *57,* 985–988.

227 C. Peng, L. Zhang, T. Head-Gordon, *Biophys. J.* **2010,** *98,* 2356–2364.

228 C.-Y. Lu, D. E. Makarov, G. Henkelman, *J. Chem. Phys.* **2010,** *133,* 201101.

229 P. Faccioli, A. Lonardi, H. Orland, *J. Chem. Phys.* **2010,** *133,* 045104.

230 M. A. Rohrdanz, W. Zheng, M. Maggioni, C. Clementi, *J. Chem. Phys.* **2011,** *134,* 124116.

231 Z. D. Pozun, K. Hansen, D. Sheppard, M. Rupp, K.-R. Müller, G. Henkelman, *J. Chem. Phys.* **2012,** *136,* 174101.

232 W. L. Jorgensen, J. Tirado-Rives, *Abstr. Pap. Am. Chem. S.* **1998,** *216,* 043–COMP.

233 N. A. McDonald, W. L. Jorgensen, *J. Phys. Chem. C* **1998,** *102,* 8049–8059.

234 G. A. Kaminski, R. A. Friesner, J. Tirado-rives, W. L. Jorgensen, *J. Phys. Chem. C* **2001,** *105,* 6474–6487.

235 W. Damm, T. A. Halgren, R. B. Murphy, A. M. Smondyrev, R. A. Friesner, W. L. Jorgensen, *Abstr. Pap. Am. Chem. S.* **2002,** *224,* 009–COMP.

236 W. D. Cornell, P. Cieplak, K. M. Merz, J. W. Caldwell, P. A. Kollman, *J. Am. Chem. Soc.* **1995,** *177,* 5179–5197.

237 V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, C. Simmerling, *Proteins* **2006,** *65,* 712–25.

238 J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, D. A. Case, *J. Comput. Chem.* **2004,** *25,* 1157–1174.

239 A. D. MacKerell, *et al. J. Phys. Chem. B* **1998,** *102,* 3586–3616.

240 A. D. MacKerell, *Abstr. Pap. Am. Chem. S.* **1998,** *216,* U696–U696.

241 A. D. MacKerell, N. K. Banavali, *J. Comput. Chem.* **2000,** *21,* 105–120.

242 N. Foloppe, A. D. MacKerell, *J. Comput. Chem.* **2000,** *21,* 86–104.

243 N. L. Allinger, Y. H. Yuh, J.-H. Lii, *J. Am. Chem. Soc.* **1989,** *11a,* 8551–8566.

244 J. H. Lii, N. L. Allinger, *J. Am. Chem. Soc.* **1989,** *111,* 8566–8575.

245 J.-H. Lii, N. L. Allinger, *J. Am. Chem. Soc.* **1989,** *111,* 8576–8582.

246 J.-H. Lii, N. L. Allinger, *J. Comput. Chem.* **1991,** *12,* 186–199.

247 J.-H. Lii, N. L. Allinger, *J. Comput. Chem.* **1998,** *19,* 1001–1016.

248 J. W. Ponder, D. A. Case, *Adv. Protein Chem.* **2003,** *66,* 27–85.

249 J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. a. DiStasio, M. Head-Gordon, G. N. I. Clark, M. E. Johnson, T. Head-Gordon, *J. Phys. Chem. B* **2010,** *114,* 2549–64.

250 P. Y. Ren, J. W. Ponder, *J. Phys. Chem. B* **2003,** *107,* 5933–5947.

251 N. Carolina, C. Hill, J. P. M. Postma, *Biopolymers* **1984,** *23,* 1513–1518.

252 K.-H. Ott, B. Meyers, *J. Comput. Chem.* **1996,** *17,* 1068–1084.

253 L. D. Schuler, X. Daura, W. F. van Gunsteren, *J. Comput. Chem.* **2001,** *22,* 1205–1218.

254 S. J. Marrink, A. H. de Vries, A. E. Mark, *J. Phys. Chem. B* **2004,** *108,* 750–760.

255 S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, A. H. de Vries, *J. Phys. Chem. B* **2007,** *111,* 7812–24.

256 L. Monticelli, S. K. Kandasamy, X. Periole, R. G. Larson, D. P. Tieleman, S.-J. Marrink, *J. Chem. Theory Comput.* **2008,** *4,* 819–834.

257 N. L. Allinger, *J. Am. Chem. Soc.* **1977,** *99,* 8127–8134.

258 J.-H. Lii, S. Gallion, C. Bender, H. Wikstrom, N. L. Allinger, K. M. Flurchick, *J. Comput. Chem.* **1989,** *10,* 503–513.

259 T. A. Halgren, *J. Am. Chem. Soc.* **1992,** *114,* 7827–7843.

260 T. A. Halgren, *J. Comput. Chem.* **1996,** *17,* 490–519.

261 T. A. Halgren, *J. Comput. Chem.* **1996,** *17,* 520–552.

262 T. A. Halgren, *J. Comput. Chem.* **1996,** *17,* 553–586.

263 T. A. Halgren, R. B. Nachbar, *J. Comput. Chem.* **1996,** *17,* 587–615.

264 T. A. Halgren, *J. Comput. Chem.* **1996,** *17,* 616–641.

265 M. Tafipolsky, B. Engels, *J. Chem. Theory Comput.* **2011,** *7,* 1791–1803.

266 accessed 27/09/12 http://www.ks.uiuc.edu/Research/namd/2.6/ug/ug.html, 2012.

267 S. R. Niketic, K. Rasmussen, The Consistent Force Field. In *Lect. N. Chem.; 3*; Springer-Verlag: Berlin, Heidelberg, New York, 1977.

268 I. N. Bronstein, K. A. Semendjajew, G. Musiol, H. Mühlig, *Taschenbuch der Mathematik;* Wissenschaftlicher Verlag Harri Deutsch: Frankfurt am Main, 7 ed.; 2008.

269 W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical Recipes;* Cambridge University Press: New York, 3rd ed.; 2007.

270 D. C. Liu, J. Nocedal, *Math. Programm.* **1989,** *45,* 503–528.

271 J. Nocedal, *Math. Comput.* **1980,** *35,* 773–782.

272 C. Broyden, *J. Inst. Math. Appl.* **1970,** *6,* 76–90.

273 R. Fletcher, *Comput. J.* **1970,** *13,* 317–322.

274 D. Goldfarb, *Math. Comput.* **1970,** *24,* 23–26.

275 D. F. Shanno, *Math. Comput.* **1970,** *24,* 647–656.

276 P. Taylor, D. J. Wales, *Mol. Phys.* **1991,** *74,* 1–25.

277 D. J. Wales, *J. Chem. Soc. Faraday T.* **1992,** *88,* 653–657.

278 D. J. Wales, *J. Chem. Soc. Faraday T.* **1993,** *89,* 1305.

279 D. J. Wales, *J. Chem. Phys.* **1994,** *101,* 3750.

280 D. J. Wales, T. R. Walsh, *J. Chem. Phys.* **1996,** *105,* 6957.

281 A. F. Voter, *J. Chem. Phys.* **1997,** *106,* 4665.

282 A. Voter, *Phys. Rev. Lett.* **1997,** *78,* 3908–3911.

283 C. Shang, Z.-P. Liu, *J. Chem. Theory Comput.* **2010,** *6,* 1136–1144.

284 E. Weinan, W. Ren, E. Vanden-Eijnden, *Phys. Rev. B* **2002,** *66,* 5–8.

285 accessed 23/10/12 http://theory.cm.utexas.edu/henkelman/research/saddle/neb.jpg, 2012.

286 G. Henkelman, G. Johannesson, H. Jonsson, Chapter 10 Methods for Finding Saddle Points and Minimum Energy Paths. In *Theoretical Methods in Condensed Phase Chemistry - Progress in Theoretical Chemistry and Physics, Vol.5*; S. D. Schwartz, (Ed.), Kluwer Academic Publishers: 2002.

287 D. Sheppard, R. Terrell, G. Henkelman, *J. Chem. Phys.* **2008,** *128,* 134106.

288 S. Stepanenko, *Global Optimization Methods based on Tabu Search,* PhD thesis, University of Wuerzburg, 2008.

289 F. Glover, *Comput. Oper. Res.* **1986,** *13,* 533–549.

290 R. Battiti, V. Sommarive, *Computers & Mathematics with Applications* **1994,** *28,* 1–8.

291 R. Battiti, G. Tecchiolli, *Ann. Oper. Res.* **1996,** *63,* 153–188.

292 A.-R. Hedar, M. Fukushima, *Eur. J. Oper. Res.* **2006,** *170,* 329–349.

293 P. Siarry, G. Berthiau, *Int. J. Numer. Meth. Eng.* **1997,** *40,* 2449–2457.

294 R. Chelouah, P. Siarry, *Eur. J. Oper. Res.* **2000,** *123,* 256–270.

295 Q. Shen, W.-M. Shi, W. Kong, *Artif. Intell. Med.* **2010,** *49,* 61–66.

296 X. Zhang, T. Wang, H. Luo, J. Y. Yang, Y. Deng, J. Tang, M. Q. Yang, *BMC Syst. Biol.* **2010,** *4 Suppl 1,* S6.

297 I. Dotu, M. Cebria, P. V. Hentenryck, P. Clote, *IEEE ACM T. Comput. Bi.* **2011,** *8,* 1620–1632.

298 M. Rusu, W. Wriggers, *J. Struct. Biol.* **2012,** *177,* 410–9.

299 M. Rusu, Z. Starosolski, M. Wahle, A. Rigort, W. Wriggers, *J. Struct. Biol.* **2012,** *178,* 121–128.

300 D. Weber, *Softwareentwicklung zur Berechnung von Kraftfeldenergien,* F-Bericht thesis, University of Wuerzburg, 2010.

301 Seeker; accessed 20/09/12 http://www.linuxinsight.com/how_fast_is_your_disk.html, 2012.

302 MOPAC2009, James J. P. Stewart, Stewart Computational Chemistry, Version 9.096L web: http://OpenMOPAC.net.

303 M. Elstner, D. Porezag, G. Jungnickel, J. Elsner, M. Haugk, S. Suhai, G. Seifert, *Phys. Rev. B* **1998,** *58,* 7260–7268.

304 I. S. Ufimtsev, T. J. Martinez, *J. Chem. Theory Comput.* **2009,** *5,* 2619–2628.

305 J. W. Ponder, F. M. Richards, *J. Comput. Chem.* **1987,** *8,* 1016–1024.

306 M. E. Hodsdon, J. W. Ponder, D. P. Cistola, *J. Mol. Biol.* **1996,** *264,* 585–602.

307 C. E. Kundrot, J. W. Ponder, F. M. Richards, *J. Comput. Chem.* **1991,** *12,* 402–409.

308 P. Y. Ren, J. W. Ponder, *J. Comput. Chem.* **2002,** *23,* 1497–1506.

309 The Open Babel Package, version 2.2.2, http://openbabel.sourceforge.net/.

310 R. Guha, M. T. Howard, G. R. Hutchison, P. Murray-Rust, H. Rzepa, C. Steinbeck, J. Wegner, E. L. Willighagen, *Journal of chemical information and modeling* **2006,** *46,* 991–8.

311 Open Babel: API Documentation; accessed 17/10/12 http://openbabel.org/api/2.3.0/, 2012.

312 N. Clayden, J. Greeves, S. Warren, P. Wothers, *Organic Chemistry;* Oxford University Press: 2004.

313 ALGLIB; accessed 25/09/12 http://www.alglib.net/, 2012.

314 libLBFGS: L-BFGS library written in C; accessed 25/09/12 http://www.chokkan.org/software/liblbfgs/index.html, 2012.

315 S. Brickel, *Tabu search based global optimization of an enzyme-inhibitor complex,* Bachelor thesis, University of Wuerzburg, 2011.

316 L. P. Pason, *Entwicklung von Algorithmen zur globalen Optimierung eines Enzym-Ligand-Komplexes,* Bachelor thesis, University of Wuerzburg, 2011.

317 J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kalé, K. Schulten, *J. Comput. Chem.* **2005,** *26,* 1781–802.

318 G. Klebe, *Wirkstoffdesign;* Spektrum Akademischer Verlag: Heidelberg, Second ed.; 2009.

319 P. Echenique, J. L. Alonso, *J. Comput. Chem.* **2006,** *27,* 1076–87.

320 C. Grebner, J. Kästner, W. Thiel, B. Engels, *J. Chem. Theory Comput.* **2012,** http://dx.doi.org/10.1021/ct300898d.

321 Vega ZZ http://www.vegazz.net, 2009.

322 B. Lee, F. M. Richards, *J. Mol. Biol.* **1971,** *55,* 379–400.

323 A. Shrake, J. A. Rupley, *J. Mol. Biol.* **1973,** *79,* 351–71.

324 W. Humphrey, A. Dalke, K. Schulten, *J. Mol. Graphics* **1996,** *14,* 33–38.

325 A. Szabo, N. S. Ostlund, *Modern Quantum Chemistry;* Dover Publications: New York, First ed.; 1996.

326 P. Emsley, B. Lohkamp, W. G. Scott, K. Cowtan, *Acta Crystallogr. D* **2010,** *66,* 486–501.

327 W. Mass, *Kristallstrukturbestimmung;* Teubner: Stuttgart, Second ed.; 1996.

328 M. D. Winn, *et al. Acta Crystallogr. D* **2011,** *67,* 235–42.

329 Lennard-Jones Clusters http://physchem.ox.ac.uk/˜doye/jon/structures/LJ.html, 2012.

330 J. P. K. Doye, D. J. Wales, M. A. Miller, *J. Chem. Phys.* **1998,** *109,* 8143–8153.

331 J. P. K. Doye, M. A. Miller, D. J. Wales, *Chem. Phys.* **1999,** *110,*.

332 W. L. Jorgensen, *J. Am. Chem. Soc.* **1981,** *103,* 335–340.

333 S. Maheshwary, N. Patel, N. Sathyamurthy, A. D. Kulkarni, S. R. Gadre, *J. Phys. Chem. A* **2001,** *105,* 10525–10537.

334 A. Weickert, *Screening of the influence of ring structures for global optimization,* F-Bericht thesis, University of Wuerzburg, 2010.

335 C. Grebner, S. Niebling, C. Schmuck, S. Schlücker, B. Engels, *J. Phys. Chem. B* **2012,** *submitted,*.

336 W. L. Jorgensen, D. S. Maxwell, J. Tirado-Rives, *J. Am. Chem. Soc.* **1996,** *118,* 11225–11236.

337 W. L. Jorgensen, N. A. McDonald, *J. Mol. Struc.-Theochem* **1998,** *424,* 145–155.

338 *TURBOMOLE V6.3.1 2011, a development of University of Karlsruhe (TH) and Forschungszentrum Karlsruhe GmbH, 1989-2007, TURBOMOLE GmbH since 2007; http://www.turbomole.com;*.

339 F. Weigend, *Phys. Chem. Chem. Phys.* **2006,** *8,* 1057–1065.

340 F. Weigend, R. Ahlrichs, *Phys. Chem. Chem. Phys.* **2005,** *7,* 3297–3305.

341 O. Vahtras, J. Almlof, M. W. Feyereisen, *Chem. Phys. Lett.* **1993,** *213,* 514–518.

342 S. Grimme, J. Antony, S. Ehrlich, H. Krieg, *J. Chem. Phys.* **2010,** *132,* 154104.

343 A. Schäfer, A. Klamt, D. Sattel, J. C. W. Lohrenz, F. Eckert, *Phys. Chem. Chem. Phys.* **2000,** *2,* 2187–2193.

344 C. Schmuck, *Coord. Chem. Rev.* **2006,** *250,* 3053–3067.

345 C. Schmuck, P. Wich, *Angew. Chem. Int. Ed.* **2006,** *45,*.

346 C. Schmuck, M. Heil, *Chem. Eur. J.* **2006,** *12,* 1339–1348.

347 D. Moiani, C. Cavallotti, A. Famulari, C. Schmuck, *Chem.-Eur. J.* **2008,** *14,* 5207–5219.

348 P. N. Day, R. Pachter, M. S. Gordon, G. N. Merrill, *J. Chem. Phys.* **2000,** *112,* 2063–2073.

349 J. K. Kazimirski, V. Buch, *J. Phys. Chem. A* **2003,** *107,* 9762–9775.

350 H. Kabrede, *Chem. Phys. Lett.* **2006,** *430,* 336–339.

351 T. James, D. J. Wales, J. H. Rojas, *J. Chem. Phys.* **2007,** *126,* 054506.

352 S. Schlund, C. Schmuck, B. Engels, *Chemistry* **2007,** *13,* 6644–53.

353 S. Schlund, R. Müller, C. Grassmann, B. Engels, *J. Comput. Chem.* **2007,** *0,* 0.

354 S. Honda, K. Yamasaki, Y. Sawada, H. Morii, *Structure* **2004,** *12,* 1507–18.

355 A. Suenaga, T. Narumi, N. Futatsugi, R. Yanai, Y. Ohno, N. Okimoto, M. Taiji, *Chem. Asian J.* **2007,** *2,* 591–8.

356 D. Satoh, K. Shimizu, S. Nakamura, T. Terada, *FEBS Lett.* **2006,** *580,* 3422–6.

357 A. D. MacKerell, M. Feig, C. L. Brooks, *J. Comput. Chem.* **2004,** *25,* 1400–15.

358 J. Tiradorives, W. L. Jorgensen, *Abstr. Pap. Am. Chem. S.* **1992,** *204,* 43–COMP.

359 K. Eichkorn, M. Htiser, R. Ahlrichs, K. Eichkorn, O. Treutler, H. Marco, R. Ahlrichs, *Chem. Phys. Lett.* **1995,** *242,* 652–660.

360 K. Eichkorn, F. Weigend, O. Treutler, R. Ahlrichs, *Theor. Chem. Acc.* **1997,** *97,* 119–124.

361 M. Sierka, A. Hogekamp, R. Ahlrichs, *J. Chem. Phys.* **2003,** *118,* 9136.

362 M. Cossi, N. Rega, G. Scalmani, V. Barone, *J. Comput. Chem.* **2003,** *24,* 669–681.

363 J. Tomasi, B. Mennucci, R. Cammi, *Chem. Rev.* **2005,** *105,* 2999–3093.

364 A. Klamt, G. Schuurmann, *J. Chem. Soc. Perk. T. 2* **1993,** 799–805.

365 M. S. Lee, F. R. Salsbury, M. A. Olson, *J. Comput. Chem.* **2004,** *25,* 1967–1978.

366 N. Derbel, B. Hernández, F. Pflüger, J. Liquier, F. Geinguenaud, N. Jaïdane, Z. B. Lakhdar, M. Ghomi, *J. Phys. Chem. B* **2007,** *111,* 1470–7.

367 G. Guiffo-Soh, B. Hernández, Y.-M. Coïc, F.-Z. Boukhalfa-Heniche, M. Ghomi, *J. Phys. Chem. B* **2007,** *111,* 12563–72.

368 G. Guiffo-Soh, B. Hernández, Y.-M. Coïc, F.-Z. Boukhalfa-Heniche, G. Fadda, M. Ghomi, *J. Phys. Chem. B* **2008,** *112,* 1282–9.

369 B. Hernández, F. Pflüger, M. Nsangou, M. Ghomi, *J. Phys. Chem. B* **2009,** *113,* 3169–78.

370 B. Hernández, C. Carelli, Y.-M. Coïc, J. De Coninck, M. Ghomi, *J. Phys. Chem. B* **2009,** *113,* 12796–803.

371 B. Hernández, F. Pflüger, N. Derbel, D. joel Coninck, M. Ghomi, *J. Phys. Chem. B* **2010,** *114,* 1077–1088.

372 F. Pflüger, B. Hernández, M. Ghomi, *J. Phys. Chem. B* **2010,** *114,* 9072–83.

373 B. Hernández, F. Pflüger, A. Adenier, S. G. Kruglik, M. Ghomi, *J. Phys. Chem. B* **2010,** *114,* 15319–30.

374 B. Hernández, F. Pflüger, A. Adenier, M. Nsangou, S. G. Kruglik, M. Ghomi, *J. Chem. Phys.* **2011,** *135,* 055101.

375 C. Brückner, *Globale Optimierung nicht kovalenter Enzy-Ligand-Komplexe von Rhodesain,* Bachelor thesis, University of Wuerzburg, 2012.

376 C. R. Sondergaard, M. H. M. Olsson, M. Rostkowski, J. H. Jensen, *J. Chem. Theory Comput.* **2011,** *7,* 2284–2295.

377 H. Li, A. D. Robertson, J. H. Jensen, *Proteins* **2005,** *61,* 704–21.

378 D. C. Bas, D. M. Rogers, J. H. Jensen, *Proteins* **2008,** *73,* 765–83.

379 M. H. M. Olsson, C. R. Sø ndergaard, M. Rostkowski, J. H. Jensen, *J. Chem. Theory Comput.* **2011,** *7,* 525–537.

380 TINKER Molecular Modeling Package, v 5.1; http://dasher.wustl.edu/ffe/ 2010.

A Appendix

Table A.1: Results for ring-opened (**2**) containing 38 atoms. [a]Relative energy of the energetically lowest minim found in the given simulation with respect to the lowest minimum found in all simulations (E = -2.8 kcal mol^{-1}). Energies are given in kcal mol^{-1}. [b]Percentage of simulations runs which found the minimum depicted in column one. [c]Average number of steps (MCM and GOTS) or snap shot (MD and SA) needed to find the minimum in column one the first time. The averaging is only done for runs where the minimum was found. [d]Corresponding averaged CPU time in minutes.

Optimization method	E^a_{min}	#global^b (%)	#steps^c	CPU time^d
MD	0.5	90	23	0.7
SA	0.5	63	38	1.1
BH	0.0	100	420	0.8
GOTS	0.0	33	319	0.7
GOTS/BH	0.0	67	224	0.9
MD-StartOpt	0.5	92	20	0.6
SA-StartOpt	0.4	31	53	1.6
BH-StartOpt	0.0	100	1858	3.7
GOTS-StartOpt	0.4	15	131	0.3
GOTS-StartOpt/Mult	0.0	11	201	0.4
GOTS/BH-StartOpt/Mult	0.0	99	192	0.7

Table A.2: Results for ring-opened (**3**) containing 50 atoms. [a]Relative energy of the energetically lowest minim found in the given simulation with respect to the lowest minimum found in all simulations (E = -68.8 kcal mol^{-1}). Energies are given in kcal mol^{-1}. [b]Percentage of simulations runs which found the minimum depicted in column one. [c]Average number of steps (MCM and GOTS) or snap shot (MD and SA) needed to find the minimum in column one the first time. The averaging is only done for runs where the minimum was found. [d]Corresponding averaged CPU time in minutes.

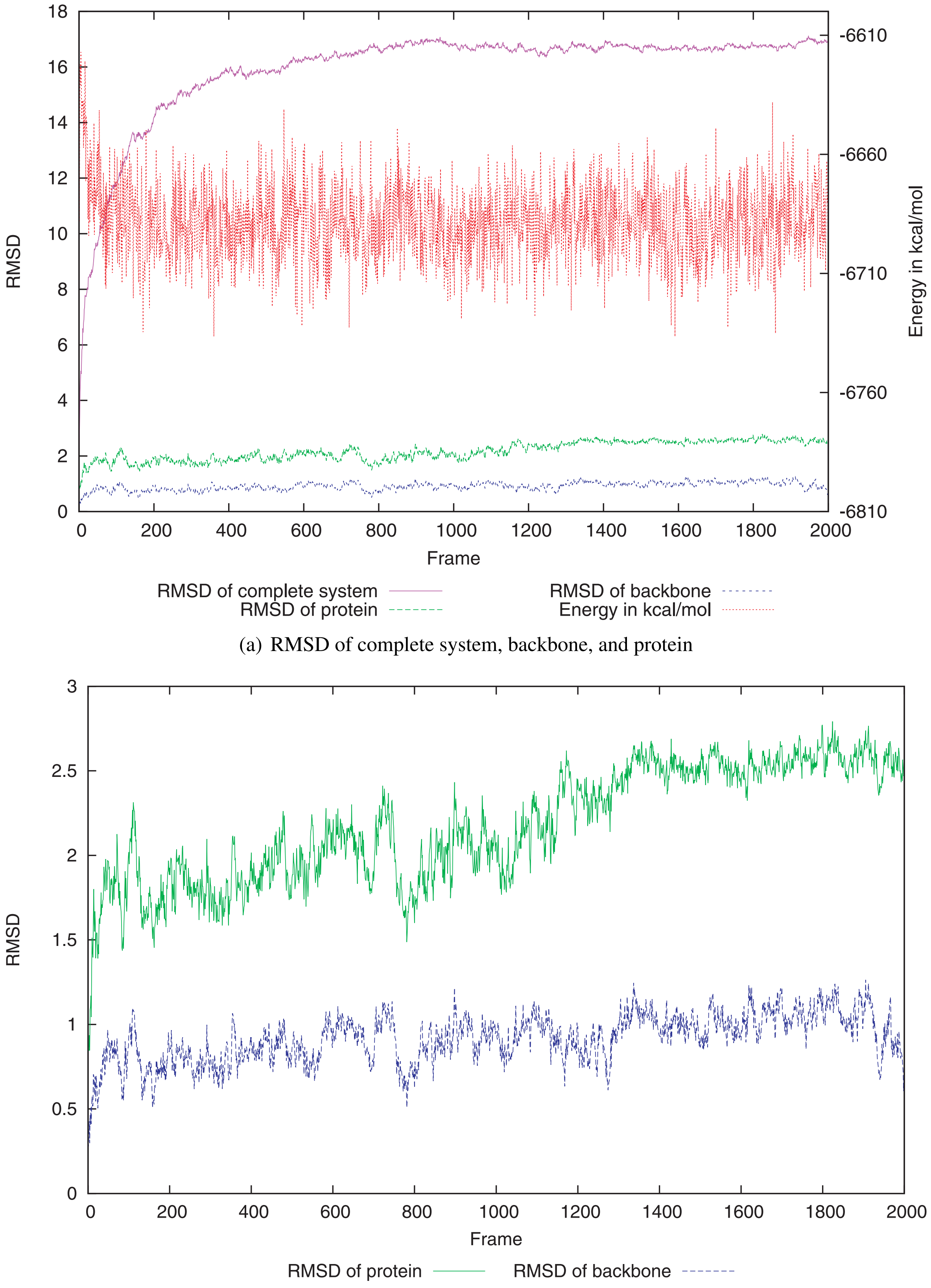
Optimization method	E^a_{min}	#global^b (%)	#steps^c	CPU time^d
MD	2.1	63	32	1.6
SA	0.0	27	47	2.4
BH	0.0	87	830	1.2
GOTS	0.0	13	276	0.5
GOTS/BH	0.0	93	113	0.7
MD-StartOpt	0.0	-	45	2.3
SA-StartOpt	0.0	25	56	2.8
BH-StartOpt	0.0	92	915	1.3
GOTS-StartOpt	0.4	-	365	0.7
GOTS-StartOpt/Mult	0.0	10	392	0.7
GOTS/BH-StartOpt/Mult	0.0	98	125	0.7

Table A.3: Results for peptide (**4**) containing 75 atoms. [a]Relative energy of the energetically lowest minim found in the given simulation with respect to the lowest minimum found in all simulations (E = -263.5 kcal mol^{-1}). Energies are given in kcal mol^{-1}. [b]Percentage of simulations runs which found the minimum depicted in column one. [c]Average number of steps (MCM and GOTS) or snap shot (MD and SA) needed to find the minimum in column one the first time. The averaging is only done for runs where the minimum was found. [d]Corresponding averaged CPU time in minutes.

Optimization method	E^a_{min}	#global[b] (%)	#steps[c]	CPU time[d]
MD	7.2	67	36	4.0
SA	1.5	20	59	6.5
BH	0.0	57	2610	19.3
GOTS	0.0	-	159	1.2
GOTS/BH	0.0	87	397	4.5
MD-StartOpt	2.0	15	49	3.9
SA-StartOpt	2.0	46	48	5.4
BH-StartOpt	0.0	42	2417	17.9
GOTS-StartOpt	0.4	-	35	0.3
GOTS-StartOpt/Mult	0.0	4	172	1.3
GOTS/BH-StartOpt/Mult	0.0	76	464	4.8

Table A.4: Benchmark of different Tabu-Search versions. All benchmarks were performed on an AMD Phenom II X4 955 (3.2 GHz) with 8Gb DDR3 1066MHz RAM using Suse 11.2 64 bit. Each Tabu-Search run was performed for 1000 iterations. The convergence criteria for local optimization was set to a gradient-RMSD of 0.01 for all calculations. All timings are given in CPU-seconds. TINKER_IO is the first implementation using Tinker with I/O; TINKER is the I/O optimized version. The original version using ChemShell for energy and gradient calculation is not included any more as the performance is much worse. FORCE is the first version with an own implementation of a force field. CAST and CAST_DIMER are the optimized implementations of GOTS within CAST, the first is using the standard neighborhood search, the other the adapted Dimer-method. The ending _MC implies the usage of MCM for diversification search of GOTS, "extern" calls the MCM routine via system call, "intern" is a direct implementation of MCM.

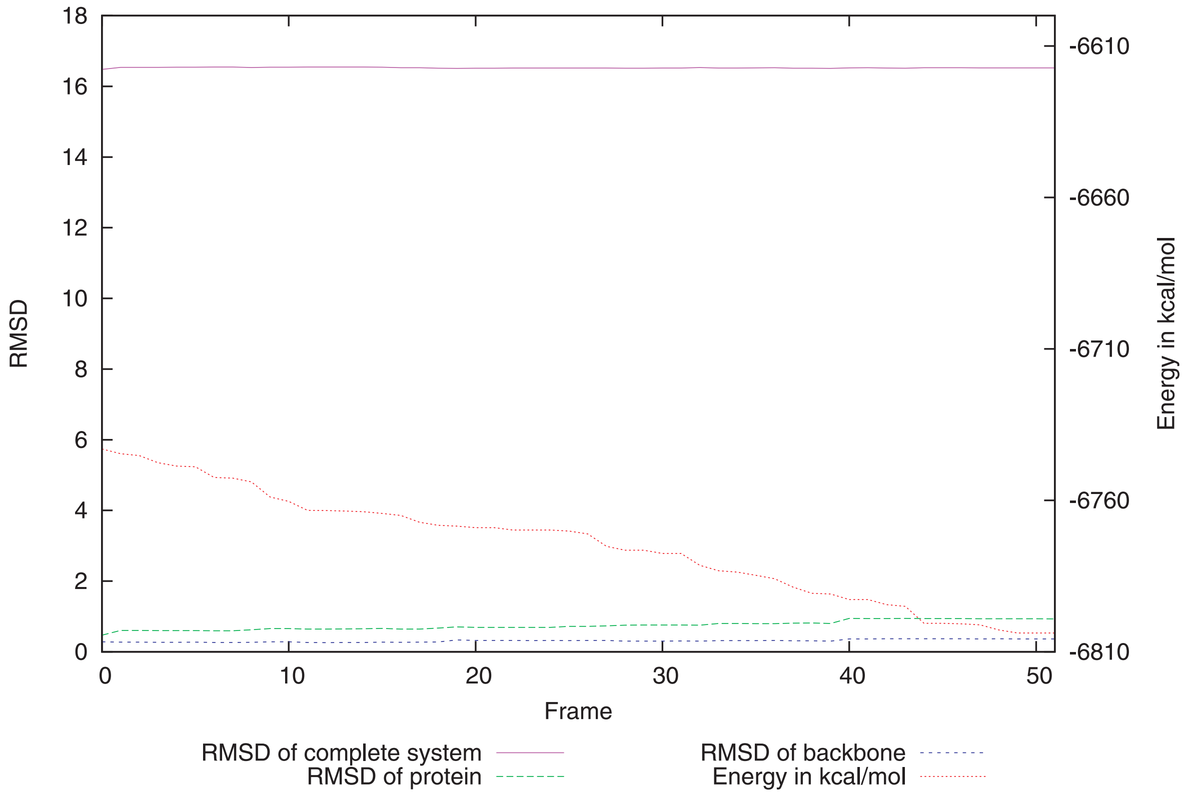
	Test system				
Tabu-Search version	Gly-Ala-Ser 31 atoms	[Met]^5-Enkephalin 75 atoms	Glu-Lys-Ser-Cys-Pro 76 atoms	Chignolin 138 atoms	Ubiquitin 1231 atoms
GOTS_TINKER_IO	2410	3629	4189	6606	n/a
GOTS_TINKER	1869	11215	5482	5597	n/a
GOTS_FORCE	248	1677	2718	4749	658141
GOTS_CAST	48	190	119	378	869064
GOTS_CAST_DIMER	27	174	211	786	20576
GOTS_TINKER_IO_MC-extern	65021	97652	60761	512649	more than 21 days
GOTS_TINKER_MC-extern	5697	17164	15808	42827	more than 21 days
GOTS_FORCE_MC-extern	2282	78042	35533	131982	more than 21 days
GOTS_CAST_MC-intern	94	409	429	912	371287
GOTS_CAST_DIMER_MC-intern	76	465	492	1485	34655



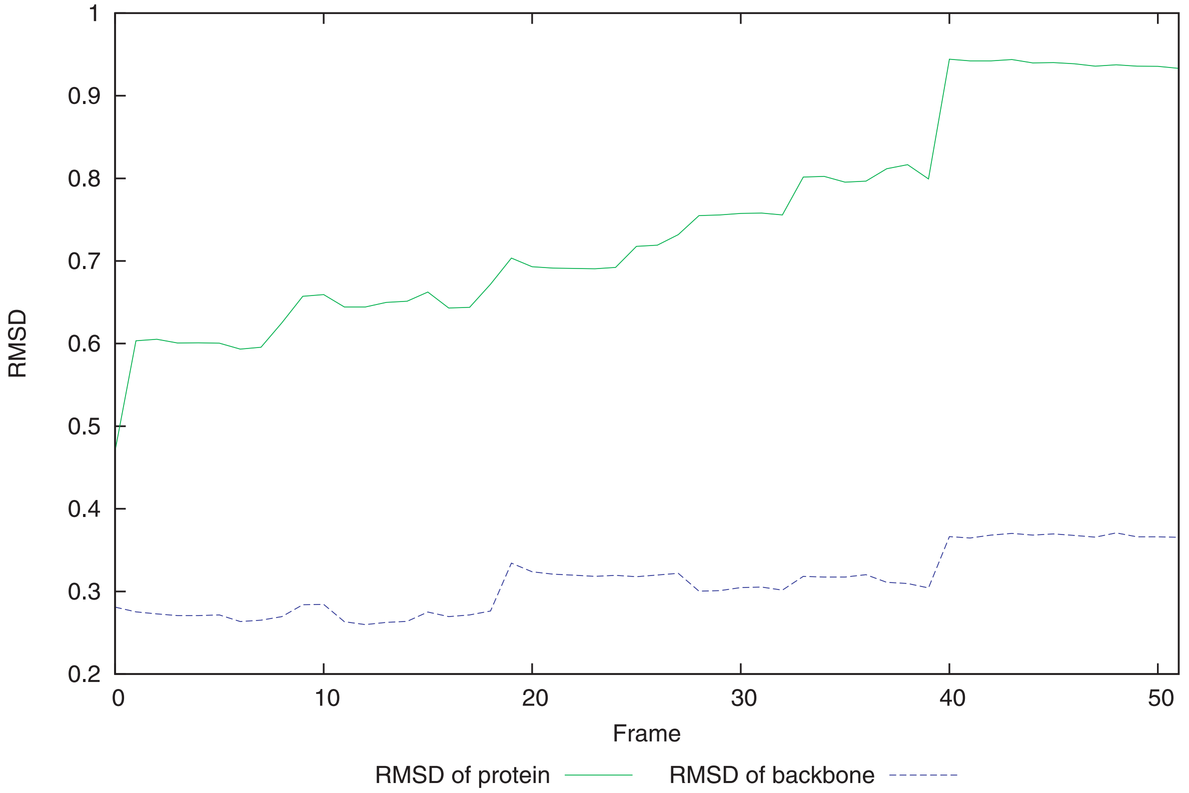
(a) RMSD of complete system, backbone, and protein

(b) RMSD of backbone and protein

Figure A.1: RMSD values (relative to the NMR structure) of the *MD-free* simulations; Red: absolute energy in kcal/mol, Blue: RMSD value of backbone, Green: RMSD value of the protein, Violet: RMSD value of the complete system.

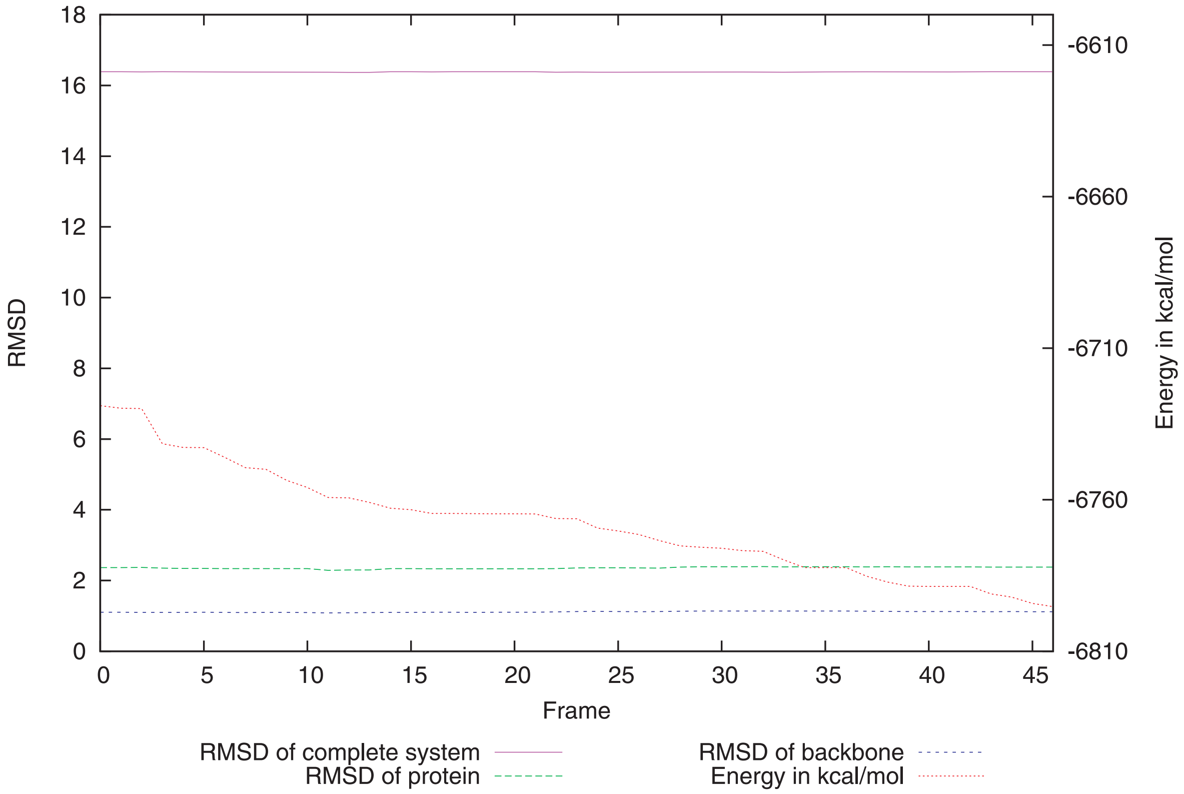


(a) RMSD of complete system, backbone, and protein

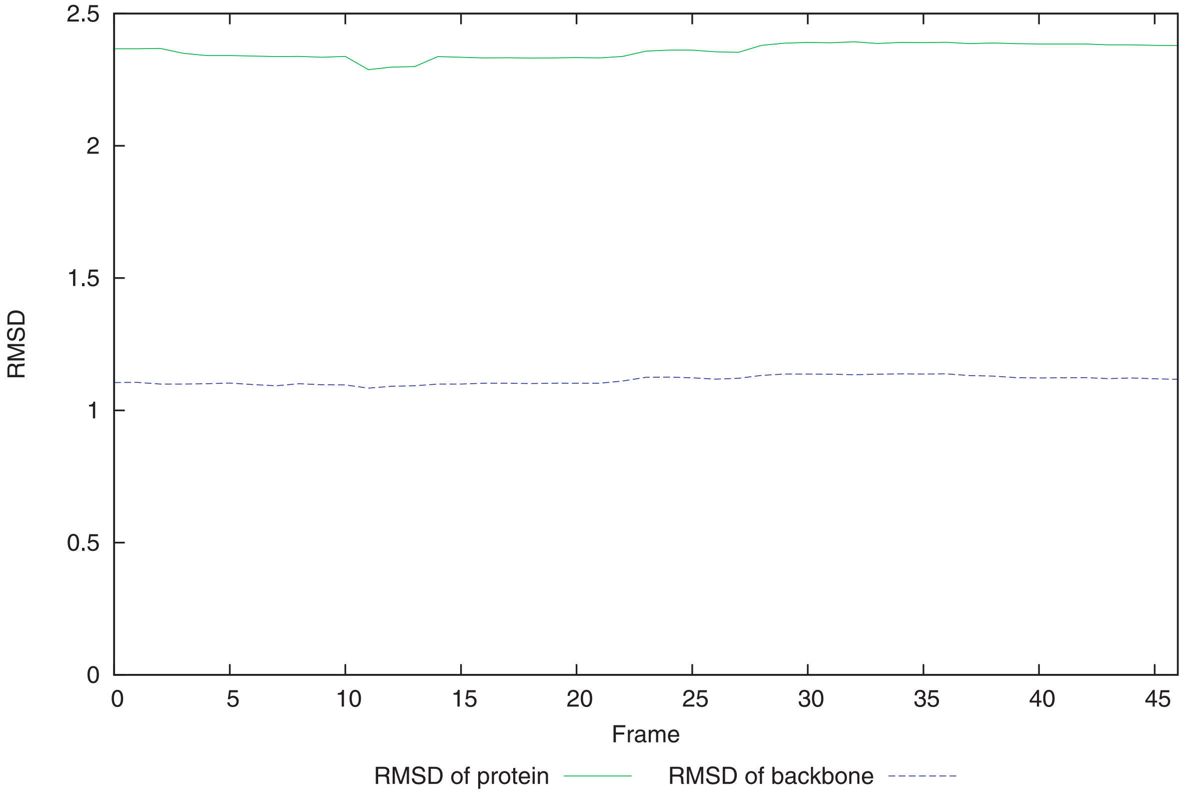


(b) RMSD of backbone and protein

Figure A.2: RMSD values (relative to the NMR structure) of the *TS-fixed* simulations; Red: absolute energy in kcal/mol, Blue: RMSD value of backbone, Green: RMSD value of the protein, Violet: RMSD value of the complete system.



(a) RMSD of complete system, backbone, and protein



(b) RMSD of backbone and protein

Figure A.3: RMSD values (relative to the NMR structure) of the *TS-free* simulations; Red: absolute energy in kcal/mol, Blue: RMSD value of backbone, Green: RMSD value of the protein, Violet: RMSD value of the complete system.

Table A.5: Atom-based RMSD values in Å of the aromatic systems of the Tyr-2 and Trp-9 residues relative to the NMR structure. Atom numbering is done according to Figure 5.14

Residue	Atom number	MD-free	TS-fixed	TS-free
Tyr-2				
	17	0.75	0.25	1.64
	18	1.08	0.36	1.40
	19	0.84	0.24	2.91
	20	1.22	0.48	2.43
	21	1.15	0.32	3.84
	22	1.23	0.46	3.58
	23	1.59	0.60	4.59
	26	1.39	0.38	0.92
	27	0.96	0.22	3.30
	28	1.50	0.60	2.50
	29	1.47	0.30	4.85
	30	1.78	0.57	3.99
Trp-9				
	114	0.98	0.12	1.45
	115	1.74	0.29	2.40
	116	0.84	0.19	1.14
	117	2.16	0.43	2.56
	118	1.62	0.29	1.54
	119	0.53	0.44	1.69
	120	1.81	0.45	1.27
	121	0.41	0.54	1.83
	122	1.07	0.49	1.02
	125	2.05	0.49	3.11
	126	2.91	0.88	3.50
	127	1.01	0.63	2.41
	128	2.52	0.67	1.89
	129	0.64	0.73	2.75
	130	1.20	0.63	1.08

Table A.6: Summary of RMSD values for SARS-CoV $\mathrm{M}^{\mathrm{pro}}$ and inhibitor (**12**) with respect to first simulation frame or x-ray structure. Averaged RMSD values* for MD simulations of all inhibitor poses. * Protein RMSD values take backbone atoms into account, inhibitor RMSD values all heavy atoms. (1st) = RMSD relative to first frame structure after 1ns equilibration. (xray) = RMSD value relative to minimized starting structure from XRay.

Structure	protein(1st)	protein(xray)	(**12**) (1st)	(**12**) (xray)
S-1	1.45±0.21	1.43±0.19	2.45±0.49	2.48±0.50
S-2	1.33±0.15	1.40±0.14	3.97±1.07	3.86±1.16
S-3	1.62±0.29	1.61±0.27	2.23±0.59	2.52±0.61
S-4	1.28±0.23	1.30±0.21	2.24±0.57	2.62±0.38
S-5	1.34±0.14	1.30±0.12	2.30±0.34	2.38±0.30
S-6	1.21±0.13	1.22±0.11	2.16±0.26	2.67±0.24
S-7	1.56±0.23	1.53±0.24	2.25±0.42	2.99±0.34
S-8	1.45±0.26	1.52±0.26	4.18±0.83	4.34±0.65
S-9	1.42±0.20	1.43±0.18	4.81±0.51	4.16±0.51
S-10	1.39±0.22	1.43±0.20	5.23±0.59	5.37±0.58
S-11	1.84±0.34	1.83±0.34	3.02±0.51	3.21±0.56
S-12	1.34±0.13	1.34±0.12	2.75±0.69	2.81±0.60
R-1	1.36±0.20	1.36±0.19	1.71±0.59	1.99±0.48
R-2	1.36±0.16	1.40±0.15	1.62±0.40	2.26±0.30
R-3	1.30±0.15	1.26±0.14	2.82±0.62	3.40±0.67
R-4	1.37±0.27	1.39±0.26	2.10±0.52	1.88±0.47
R-5	1.31±0.18	1.35±0.15	2.57±0.80	2.88±0.85

Danksagung

Diese Arbeit wurde in der Zeit von November 2009 bis Dezember 2012 am Institut für Physikalische und Theoretische Chemie an der Julius-Maximilians-Universität Würzburg durchgeführt. An dieser Stelle möchte ich mich bei allen bedanken, die zum Gelingen dieser Arbeit beigetragen haben:

Als Erstes möchte ich mich bei Prof. Dr. Bernd Engels für die Möglichkeit bedanken, meine Doktorarbeit in seiner Arbeitsgruppe durchführen zu können. Das sehr interessante Thema, die vielen Freiheiten bei der Gestaltung meiner Arbeit und die Möglichkeit, mich bei zahlreichen nationalen und internationalen Konferenzen fortbilden zu können, sind nur einige Punkte, welche die Doktorarbeit so angenehm gemacht haben.

Des Weiteren danke ich meinem Zweitgutachter Jun.-Prof. Dr. Johannes Kästner für die Begutachtung dieser Arbeit und für die vielen, hilfreichen Diskussionen.

Besonders möchte ich mich auch bei der gesamten Arbeitsgruppe bedanken. Die stets angenehme Stimmung und die vielen lustigen Unternehmungen lassen mich die Zeit der Doktorarbeit in sehr schöner Erinnerung behalten.
Damit auch keiner vergessen werden kann, habe ich mich entschlossen, an dieser Stelle auf die Nennung einzelner Namen zu verzichten, da jeder in irgendeiner Art und Weise an der schönen Zeit beteiligt war.
Mein besonderer Dank gilt allerdings meinen Praktikanten und Diplomanden, die durch ihre Hilfe viel zu dem Gelingen meiner Arbeit beigetragen haben.

An dieser Stelle möchte ich mich auch bei allen Bedanken, die meine Arbeit Korrektur gelesen haben: Zarah Falk, Prisca Eckert, Stefan Niebling, Alexander Paasche, Johannes Becker und Wiebke Grebner.

Ganz herzlich möchte ich mich bei meiner Familie, insbesondere bei meinen Eltern, für die fortwährende Unterstützung bedanken.

Zu guter Letzt gilt mein ganz persönlicher Dank meiner Frau Wiebke.

Erklärung

Hiermit erkläre ich an Eides statt, dass ich die Dissertation

New Tabu-Search Algorithms for the Exploration of Energy Landscapes of Molecular Systems

selbständig angefertigt und keine anderen als die von mir angegebenen Quellen und Hilfsmittel benutzt habe.

Ich erkläre außerdem, dass diese Dissertation weder in gleicher oder anderer Form bereits in einem anderen Prüfungsverfahren vorgelegen hat.

Ich habe früher außer den mit dem Zulassungsgesuch urkundlich vorgelegten Graden keine weiteren akademischen Grade erworben oder zu erwerben versucht.

Würzburg, 4. Dezember 2012

Christoph Grebner