

UNIVERSITÄT BONN
Institut für Angewandte Mathematik

**Zur globalen Minimierung von
Energie-Funktionen**

von

Lukas Jäger

Dieser Forschungsbericht wurde als Diplomarbeit von der Mathematisch - Naturwissenschaftlichen Fakultät der Universität Bonn angenommen.

Angenommen am: 08.02.2002
Referent: Prof. Dr. Michael Griebel
Koreferent: Prof. Dr. Hartmut Monien

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Überblick	2
2	Verfahren zur globalen Minimierung	9
2.1	Graduated-Non-Convexity (GNC)	9
2.2	Diffusionsglättung	10
2.2.1	Diffusion Equation Method (DEM)	11
2.2.2	Der Basin-Hopping-Algorithmus	15
2.3	Simulated Annealing	18
2.3.1	Monte-Carlo (MC)	19
2.3.2	Molekül-Dynamik (MD)	19
2.4	Simulated Annealing in Temperature	22
2.4.1	Gaussian Density Annealing (GDA)	23
2.4.2	adiabatic Gaussian Density Annealing (aGDA)	26
2.4.3	Quantum-Mechanical Annealing (QMA)	26
2.4.4	Packet Annealing Method (PAM)	29
2.4.5	Minimierung mittels des Effective Diffused Potential (EDP)	31
2.5	Zusammenfassung	35
3	Lokale Minimierungsverfahren	39
3.1	Die Verfahren	39
3.1.1	Methode des steilsten Abstiegs	40
3.1.2	Konjugierte Gradienten	41
3.2	Liniensuche	42
3.3	Abbruchkriterien	43
4	Optimierung von Lennard-Jones-Clustern	45
4.1	Das Lennard-Jones-Potential	45
4.2	Anwendung der DEM	47
4.2.1	Probleme mit dem Verfahren nach Scheraga	52

4.3	Anwendung des aGDA und des QMA	54
4.4	Minimierung von Lennard-Jones-Clustern mittels des EDP	56
4.5	Minimierung mit dem Basin-Hopping-Algorithmus	61
4.6	Bilder einiger Minima	63
4.7	Zusammenfassung	66
5	Optimierung von Wasser-Clustern	69
5.1	Das TIP4P Potential	69
5.1.1	Bewegung eines Moleküls als starrer Körper	70
5.2	Anwendung der DEM auf Wasser	71
5.3	Anwendung des Basin-Hopping auf Wasser	76
5.4	Zusammenfassung	79
6	Optimierung von Proteinen	83
6.1	Einleitung	83
6.2	Das CHARMM-Potential	83
6.3	Das Minimierungsverfahren	85
6.4	Anwendung auf BPTI	89
6.5	Zusammenfassung	92
7	Zusammenfassung und Ausblick	95
A	Herleitung der Bewegungsgleichungen	97
A.1	Bewegungsgleichungen des GDA	97
A.2	Bewegungsgleichungen des QMA	98

Kapitel 1

Einleitung

1.1 Motivation

Optimierungsprobleme treten in einer Vielzahl von verschiedenen Bereichen auf. So gehört zum Beispiel sowohl das klassische Problem des Handlungsreisenden wie auch die Optimierung von hochintegrierten Schaltungen in der Elektronik oder die Minimierung des Verschnitts beim Zuschneiden von Stoffbahnen zu dieser Klasse von Problemen. In dieser Arbeit wird die Optimierung aber ausschließlich im Zusammenhang der Bestimmung von Grundzuständen physikalischer Systeme verstanden. Mit physikalischen Systemen sind zum Beispiel atomar aufgebaute Moleküle oder Cluster aus ungebundenen Atomen gemeint. Sie werden beschrieben durch Potentialfunktionen, die ausschließlich von der räumlichen Konfiguration der Atome abhängen:

$$U = U(\mathbf{x}) \quad \text{mit} \quad \mathbf{x} = \begin{pmatrix} \vec{x}_1 \\ \vdots \\ \vec{x}_N \end{pmatrix} \quad (1.1)$$

Dabei soll N die Zahl der betrachteten Atome sein und $\mathbf{x} \in \mathfrak{R}^{3N}$ ist der $3N$ -dimensionale Vektor, der ihre Positionen angibt. Bei niedrigen Temperaturen haben die Atome nur wenig kinetische Energie und sind daher durch das Potential auf einen kleinen Bereich des Konfigurationsraumes beschränkt. Statistisch beschreibt man die Konfiguration des Systems mithilfe einer Wahrscheinlichkeitsverteilung, die dann in diesem Bereich konzentriert ist. Die Wahrscheinlichkeitsverteilung realer Systeme stellt zu jeder Temperatur ein Minimum der freien Energie $F = E - TS$ dar, wobei $E = \langle U \rangle$ die thermodynamisch gemittelte potentielle Energie und S die Entropie der Verteilung ist. Überträgt man das auf die physikalisch nicht zu erreichende Temperatur $T = 0K$, so müßte das System sich dort im globalen Minimum \mathbf{x}_{min} der potentiellen Energie befinden:

$$U(\mathbf{x}_{min}) \leq U(\mathbf{x}) \quad \forall \mathbf{x} \in \mathfrak{R}^{3N} \quad (1.2)$$

Man möchte nun eigentlich in der Lage sein, zu einem gegebenen System nicht nur die Wahrscheinlichkeitsverteilung, sondern auch ihr Maximum für beliebige Temperaturen angeben zu können. Besonders interessiert ist man dabei natürlich an den niedrigen oder „normalen“ Temperaturbereichen, bei denen sich zum Beispiel chemische Reaktionen

ereignen. Mit der Verteilung und ihrem Maximum würde man die Konfiguration kennen, um die das System je nach Temperatur schwankt. Die Kenntnis dieser Konfiguration kann beim Verständnis vieler chemischer und biologischer Vorgänge helfen, da die chemischen Eigenschaften von Molekülen wie zum Beispiel Proteinen von ihrer räumlichen Struktur abhängen. Experimentell lassen sich die Strukturen aber meist nur im kristallisierten Zustand bestimmen, so daß man eine andere Möglichkeit sucht, sie zu ermitteln. Numerische Verfahren, die dieses Problem lösen sollen, sind deswegen ein Gebiet aktueller Forschung und könnten bei Erfolg erheblich zu unserem Verständnis der Natur beitragen.

Die Lösung dieser Aufgabe stellt allerdings eine enorme Herausforderung an numerische Verfahren dar und man beschränkt sich daher zunächst auf eine etwas vereinfachte Problemstellung. Man begnügt sich damit, das globale Minimum der Potentialfunktion zu bestimmen, um so das Maximum der Wahrscheinlichkeitsverteilung bei sehr niedrigen Temperaturen zu erhalten. Wird die Wahrscheinlichkeitsverteilung durch eine sehr einfache Näherung repräsentiert, so kann man mit ihr auch die freie Energie berechnen und erhält so eine etwas realistischere Darstellung.

Der Grund für die großen Schwierigkeiten bei der Bestimmung der Grundzustände ist die komplizierte Struktur der Potentiale. Die unter Umständen extrem hohe Zahl an Freiheitsgraden eines Systems bedingt eine Potentialfläche, die eine so große Anzahl von Minima und Maxima aufweist, daß jeglicher Versuch, das globale Minimum zum Beispiel durch Suchen zu bestimmen, zum Scheitern verurteilt ist. Man muß Verfahren entwickeln, die die relevanten Informationen aus der Potentialfunktion herausfiltern können, ohne sie im Detail untersuchen zu müssen. Solche Verfahren sind das Thema dieser Diplomarbeit.

1.2 Überblick

Ziel der in dieser Arbeit vorgestellten Verfahren soll es also sein, diejenige Konfiguration eines Systems zu finden, die ihre potentielle Energie minimiert. Für diese Aufgabe sind in der Vergangenheit bereits eine ganze Reihe von Ansätzen entwickelt worden, die aus ganz verschiedenen Gebieten der Naturwissenschaften kommen. So gibt es zum Beispiel Verfahren, die mithilfe von sogenannten genetischen Algorithmen arbeiten und das biologische Verhalten von Mutation und Auslese in der Evolution simulieren. Da hier aber ausschließlich atomare Systeme betrachtet werden sollen, bietet es sich an, physikalisch motivierte Verfahren zu verwenden, denn das Interesse an den energetisch minimalsten Konfigurationen ist durch das Verhalten der Natur begründet, daß Systeme bei niedrigen Temperaturen in ihre thermodynamischen Grundzustände fallen. Daher müßte die physikalische Beschreibung der Systeme in ihrer Temperaturabhängigkeit dieses Verhalten wiedergeben können. Dabei kann man sich entweder auf die Betrachtung der potentiellen Energie beschränken, oder man nimmt den zusätzlichen Faktor der Entropie hinzu, so daß man eine realistischere Beschreibung des natürlichen Verhaltens erhält.

Die Verfahren benutzen normalerweise die stochastischen Eigenschaften, die entweder aus der quantenmechanischen Unsicherheit oder der thermodynamischen Wahrscheinlichkeit eines Ensembles von Systemen herrühren. Solche stochastischen Betrachtungen führen zur Berechnung von Mittelwerten (Observablen), deren Zeitentwicklung oder Temperaturabhängigkeit angegeben werden kann. Zusätzlich vereinfacht die stochastische Betrachtung die Struktur der jeweiligen Größe, so daß es zum Beispiel möglich wird, die über die

stochastische Verteilung gemittelte Potentialfläche erschöpfend durchsuchen zu können. Die grundlegenden physikalischen Zusammenhänge sollen im nächsten Abschnitt näher erläutert werden.

Physikalischer Hintergrund

Der grundlegende Sinn von Computersimulationen ist es, aus den benutzten Modellen Vorhersagen für die physikalischen Systeme zu gewinnen. Dazu sollen die Modelle möglichst genau durch die physikalischen Gesetze beschrieben werden, die man aus der Beobachtung des realen Systems gewonnen hat. Da in dieser Arbeit primär atomare Cluster und Moleküle betrachtet werden sollen, ist ein geeigneter Ausgangspunkt für die Beschreibung der Eigenschaften der Atome in diesen Systemen die Schrödingergleichung:

$$i\hbar \frac{\partial}{\partial t} \Psi(\mathbf{x}, t) = H \Psi(\mathbf{x}, t) \quad \text{mit} \quad H = -\frac{\hbar^2}{2m} \nabla^2 + U(\mathbf{x}) \quad (1.3)$$

Dabei ist $\mathbf{x} \in \mathfrak{R}^{3N}$ der 3N-dimensionale Vektor, der die Orte der N Atome beschreibt, und $U(\mathbf{x})$ das Potential, welches auf die Atome wirkt. Im Folgenden werden alle Überlegungen der einfacheren Schreibweise wegen nur für eine Dimension beschrieben, gelten aber generell auch für mehr Dimensionen. Desweiteren wird angenommen, daß der Hamiltonoperator nicht explizit zeitabhängig ist, was bei den hier benutzten Potentialen auch nicht der Fall sein wird.

Alle Eigenschaften des Systems werden durch die Schrödingergleichung beschrieben, wenn man das Potential und die Anfangsbedingungen vorgibt. Im Zusammenhang mit der Optimierung interessiert man sich für die Grundzustandswellenfunktion Ψ_0 , die den Zustand mit der niedrigsten Energie repräsentiert. Die allgemeine Lösung der Schrödingergleichung ist im Prinzip bekannt, denn gibt man die Anfangsbedingung $\Psi(\hat{x}, t = 0)$ vor, so kann man schreiben:

$$\Psi(x, t) = \int K(\hat{x}, x, t) \Psi(\hat{x}, 0) d\hat{x} \quad (1.4)$$

Dabei ist $K(\hat{x}, x, t)$ der sogenannte Propagator, der die Wahrscheinlichkeit für den Übergang vom Zustand $(\hat{x}, 0)$ nach (x, t) angibt. Dieser Propagator ist selber wieder eine Lösung der Schrödingergleichung 1.3 und hat somit selber den Charakter einer Wellenfunktion mit der Eigenschaft

$$K(\hat{x}, x, 0) = \delta(\hat{x} - x) \quad (1.5)$$

Seine explizite Darstellung ist formal durch die Feynman-Kac-Formel gegeben, wird aber in physikalischen Zusammenhängen auch als Pfadintegral geschrieben:

$$K(\hat{x}, x, t) = k \int_{\hat{x}}^x e^{\frac{i}{\hbar} \int_0^t L(\dot{y}(t), y(t), t) dt} Dy(t) \quad (1.6)$$

Dabei ist k eine Normierungskonstante und L ist die Lagrange-Funktion. Die Integration wird über alle möglichen Pfade $y(t)$ mit $y(0) = \hat{x}$ und $y(t) = x$ durchgeführt, woher der Name Pfadintegral stammt. Seine Interpretation ist die folgende: Quantenmechanisch trägt zu der Übergangsamplitude $K(\hat{x}, x, t)$ jeder mögliche Weg von \hat{x} nach x mit der Gewichtung $e^{\frac{i}{\hbar} S(\hat{x}, 0, x, t)}$ bei. Im Exponenten der e-Funktion steht die Wirkung des jeweiligen Pfades. Klassisch trägt nur der eine Pfad bei, für den $\delta S = 0$ gilt, da die Wirkungen gegenüber der Skala \hbar so groß sind, daß sich die Phasen aller anderen Pfade wegmitteln. Für

quantenmechanische Systeme jedoch müssen wegen der kleineren Wirkungen viele Pfade berücksichtigt werden.

Leider kann das Integral 1.6 nur für sehr einfache Probleme berechnet werden. Ein Beispiel ist der Fall mit $U(x) = 0$ des freien Teilchens. Dann gilt $L(\dot{y}(t), y(t), t) = \frac{m}{2}\dot{y}^2$ und somit

$$K(\hat{x}, x, t) = k \int_{\hat{x}}^x e^{\frac{i}{\hbar} \int_0^t \frac{m}{2} \dot{y}^2(t) dt} Dy(t) \quad (1.7)$$

Dieses Integral kann man berechnen, indem man die zeitliche Ableitung durch n finite Differenzen ausdrückt,

$$\dot{y}^2 = \sum_{i=1}^n \frac{(x_i - x_{i-1})^2}{\epsilon^2} \quad (1.8)$$

dann die einzelnen Gauß'schen Integrale ausführt und den Grenzwert $\lim_{\epsilon \rightarrow 0}$ bildet. Das Ergebnis ist

$$K(\hat{x}, x, t) = \left[\frac{2\pi i \hbar t}{m} \right]^{-\frac{1}{2}} e^{\frac{im(x-\hat{x})^2}{2\hbar t}} \quad (1.9)$$

Setzt man jetzt die Anfangsbedingung für ein Teilchen, das zum Zeitpunkt $t = 0$ bei x_0 startet, $\Psi(\hat{x}, 0) = \delta(\hat{x} - x_0)$ für die Wellenfunktion in 1.4 ein, so ergibt sich nach richtiger Normierung:

$$\Psi(x, t) = \left[\frac{2\pi i \hbar t}{m} \right]^{-\frac{1}{4}} e^{\frac{im(x-x_0)^2}{4\hbar t}} \quad (1.10)$$

Dieses Ergebnis für die Wellenfunktion eines freien Teilchens ist der Ausgangspunkt für sämtliche Näherungen, der in dieser Arbeit beschriebenen Verfahren. Die Überlegung dabei ist die, daß man die Lösung von 1.4 für allgemeine Potentiale $U(x) \neq 0$ nicht berechnen kann und deswegen die bekannte Lösung für einen Spezialfall nimmt, hier mit $U(x) = 0$, und sie als Ansatz für die Lösung des allgemeinen Problems benutzt. Um die Übereinstimmung mit der tatsächlichen Lösung optimieren zu können, werden x_0 und t als zu bestimmende Parameter des Ansatzes verstanden. Bei den verschiedenen Verfahren wird dieser Ansatz unterschiedlich motiviert, geht aber generell immer auf diese freie Lösung für ein Teilchen zurück. Den Ansatz dann in die verschiedenen Kontexte zu übertragen ist relativ leicht. Bei der quantenmechanischen Imaginary Time Method setzt man $\tau = \frac{it}{\hbar}$ und faßt die Parameter $\lambda = \frac{\tau \hbar^2}{m}$ und x^0 als Funktion von τ auf:

$$\Psi(x, x^0, \Lambda, \tau) = [2\pi\lambda(\tau)]^{-\frac{1}{4}} e^{-\frac{(x-x_0(\tau))^2}{4\lambda(\tau)}} \quad (1.11)$$

Auch bei den Methoden aus der statistischen Physik verwendet man den Ansatz für die Wahrscheinlichkeitsverteilung, denn die (unnormierte) Bloch-Gleichung

$$\frac{\partial}{\partial \beta} p = -Hp \quad (1.12)$$

hat die gleiche Form wie die Schrödingergleichung in imaginärer Zeit. Dann werden die Parameter als Funktionen der inversen Temperatur $\beta = \frac{1}{kT}$ betrachtet:

$$p(x, x^0, \Lambda, \beta) = [2\pi\lambda(\beta)]^{-\frac{1}{2}} e^{-\frac{(x-x_0(\beta))^2}{2\lambda(\beta)}} \quad (1.13)$$

Ziel der Verfahren ist es nun, den Wert von x^0 für große imaginäre Zeit bzw. kleine Temperaturen, wo man annehmen kann, daß das System sich im Grundzustand befindet, zu bestimmen. Dazu werden die Bewegungsgleichungen für x^0 und λ , die man aus der Betrachtung der Ableitungen nach τ bzw. β erhält, bis zu großen τ bzw. β integriert, wo $\lim_{\tau \rightarrow \infty} \lambda(\tau) = 0$ bzw. $\lim_{\beta \rightarrow \infty} \lambda(\beta) = 0$ gilt.

Interessant ist in diesem Zusammenhang die Betrachtung der Verfahren, denen man im Allgemeinen einen deterministischen Ansatz zuschreibt. Davon werden in dieser Arbeit diejenigen beschrieben, die in Anlehnung an bekannte Methoden aus der Bildverarbeitung für die Optimierung der physikalischen Modelle Lösungen der Diffusionsgleichung benutzen. Dazu wird das Potential mittels der Diffusion geglättet und die Lösung läßt sich als

$$\tilde{U}(x^0, t) = \int U(x) G(x, x^0, t) dy \quad (1.14)$$

schreiben. Für den einfachsten Fall der isotropen, linearen Diffusion ist der Kern G eine Gauß-Funktion mit dem Mittelpunkt x^0 und der Breite t . Vergleicht man das mit dem Erwartungswert

$$\langle U \rangle(x^0, \lambda) = \int \Psi^*(x, x^0, \lambda) U(x) \Psi(x, x^0, \lambda) dx \quad (1.15)$$

bzw. mit dem (klassischen) thermodynamischen Mittelwert

$$\langle U \rangle(x^0, \lambda) = \int U(x) p(x, x^0, \lambda) dx \quad (1.16)$$

der potentiellen Energie, so erkennt man, daß alle Ansätze in der gleichen Transformation der Potentialfunktion resultieren. Die physikalische Interpretation für die Benutzung der Diffusionsgleichung in diesem Zusammenhang ist also die, daß die mittlere potentielle Energie, unter Benutzung des Ansatzes für freie Teilchen, Lösung der isotropen, linearen Diffusionsgleichung ist.

Ganz allgemein beruhen alle Verfahren auf einer Integraltransformation der Potentialfunktion, wobei sogar stets der gleiche Kern benutzt wird. Wollte man nun versuchen, die bestehenden Verfahren zu verbessern, so ist es naheliegend, die Transformation mit einem anderen Kern durchzuführen. Denn wegen des physikalischen Hintergrunds des Erwartungs- bzw. Mittelwertes ist es klar, daß die bis jetzt beschriebene Gauß-Funktion nicht die beste Wahl sein kann. Sie resultiert aus einer sehr einfachen Näherung der Lösung der Schrödingergleichung und kann daher das System mit vorhandenem Potential nur eingeschränkt gut beschreiben. Das sieht man besonders gut im Vergleich mit der formalen Lösung der Bloch-Gleichung, die die tatsächliche Wahrscheinlichkeitsverteilung des kanonischen Ensembles darstellt:

$$p_B(x) = e^{-\beta H(x)} \quad (1.17)$$

Wenn $U(x)$ eine Funktion mit mehreren Minima ist, so ist die Verteilung $p_B(x)$ nicht gut durch eine einzige Gauß-Funktion wiederzugeben. Vielmehr hat auch sie dann mehrere Maxima und würde besser durch eine Summe von Gauß-Verteilungen um verschiedene Mittelwerte beschrieben werden. Diesen Ansatz kann man in die bestehenden Verfahren integrieren, steht dann aber vor dem Problem, daß man die Anzahl der Summanden, die für eine ausreichend gute Näherung genügt und sich auch mit der Temperatur ändert,

nicht bestimmen kann. Eine andere Möglichkeit wäre, nicht den Ansatz eines freien Teilchens zu wählen, sondern bereits einen, der Lösung eines Problems mit nichtverschwindendem U ist. Das bedeutet, man könnte eine einfache Form für das Potential wählen, die aber in kleinen Bereichen das Potential des zu untersuchenden Problems gut wiedergibt. Das führt allerdings zu anderen Problemen, denn einerseits gibt es nur sehr wenige Potentiale, für die eine Lösung der Schrödingergleichung angegeben werden kann, und andererseits muß man mit dieser Lösung dann auch die Integraltransformation analytisch durchführen können, weil die betrachteten physikalischen Modelle sehr viele Freiheitsgrade haben können und eine numerische Ausführung der hochdimensionalen Integration daher nicht in Frage kommt. Auf das gleiche Problem stößt man, wenn man versucht einen besseren Kern aus der Verallgemeinerung der Diffusionsgleichung zu erhalten. Man kann die Lösung der anisotropen, nichtlinearen Diffusionsgleichung im Allgemeinen nicht angeben und für die Ausnahmefälle, bei denen es möglich ist, kann das Integral 1.14 nicht berechnet werden.

Hierbei sieht man noch einen wichtigen Zusammenhang zwischen den Differentialgleichungen und den Integraltransformationen. Allgemein ist es nicht möglich, zu jeder Differentialgleichung den korrespondierenden Kern für die Integralgleichung anzugeben und umgekehrt. Normalerweise ist es aber einfacher, die differentielle Form der Gleichung so zu verändern, daß ihre Lösung die gewünschten Eigenschaften besitzt, weil diese Formulierung anschaulicher ist. So beschreibt zum Beispiel die Schrödingergleichung mit dem Potential U das Verhalten des Systems vollkommen korrekt. Kann man dann aber zu der Differentialgleichung keinen Kern mehr angeben, so hilft das für die Darstellung der Lösung als Integraltransformation nicht weiter, da eine numerische Lösung der Differentialgleichung in diesem Zusammenhang nicht möglich ist. Dann hat man theoretisch auch noch die Möglichkeit, einen geeigneten Kern für Gl.1.14 zu „erraten“. Straub schlägt zum Beispiel einen Kern der Form

$$G(x, x^0, t) = \frac{1}{2t} \{ \Theta(x^0 - (x - t)) - \Theta(x^0 - (x + t)) \} \quad (1.18)$$

mit der Heaviside-Funktion $\Theta(x)$ vor. Dieser Kern führt dazu, daß der Gradient des transformierten Potentials wie eine finite-Differenzen Näherung an die Ableitung des Potentials aussieht. Ansonsten ist das Resultat der Faltung mit diesem Kern dem Ergebnis mit der Gauß-Funktion sehr ähnlich, da beide eine Mittelung über einen von t abhängigen begrenzten Bereich bewirken. Im Allgemeinen ist es natürlich sehr schwierig, einen Kern anzugeben, der zu einer vorteilhaften Transformation führt. Deswegen ist diese Methode kaum eine Alternative.

Die freie Energie

Wie anfangs schon beschrieben, reicht physikalisch die Minimierung der potentiellen Energie eigentlich nicht aus, um den thermodynamischen Grundzustand zu bestimmen, denn die Natur berücksichtigt bei der Wahrscheinlichkeit von Zuständen auch deren Ordnung. Die Ordnung wird durch einen Entropieanteil zur inneren Energie berücksichtigt. Man nennt das Resultat die freie Energie:

$$F = E - TS \quad (1.19)$$

$E = \langle U \rangle$ ist hier die innere Energie und S die Entropie bei der Temperatur T . Die meisten Optimierungsverfahren berücksichtigen diesen Entropieteil nicht und beschränken sich somit darauf, wahrscheinliche Konfigurationen für den tatsächlichen Grundzustand zu berechnen. Zwei der in dieser Arbeit vorgestellten Verfahren, optimieren das System aber mittels einer Näherung an die freie Energie. Trotzdem wird wieder der oben beschriebene Ansatz der freien Teilchen verwendet, weil man für ihn auch die Entropie leicht berechnen kann. Es gilt:

$$TS = -kT \int p(x) \ln(p(x)) dx \quad (1.20)$$

Setzt man hier die Gauß-Verteilung ein, so läßt sich die Integration leicht durchführen. Auch hier sieht man also, daß der gewählte Ansatz in allen Verfahren eine explizite Berechnung der vorkommenden Integrale ermöglicht, was für bessere aber damit auch kompliziertere Ansätze im Allgemeinen nicht mehr möglich sein wird.

Abbildung 1.1 soll einen Überblick über den Zusammenhang und die Verfahren geben, die in Kapitel 2 vorgestellt werden.

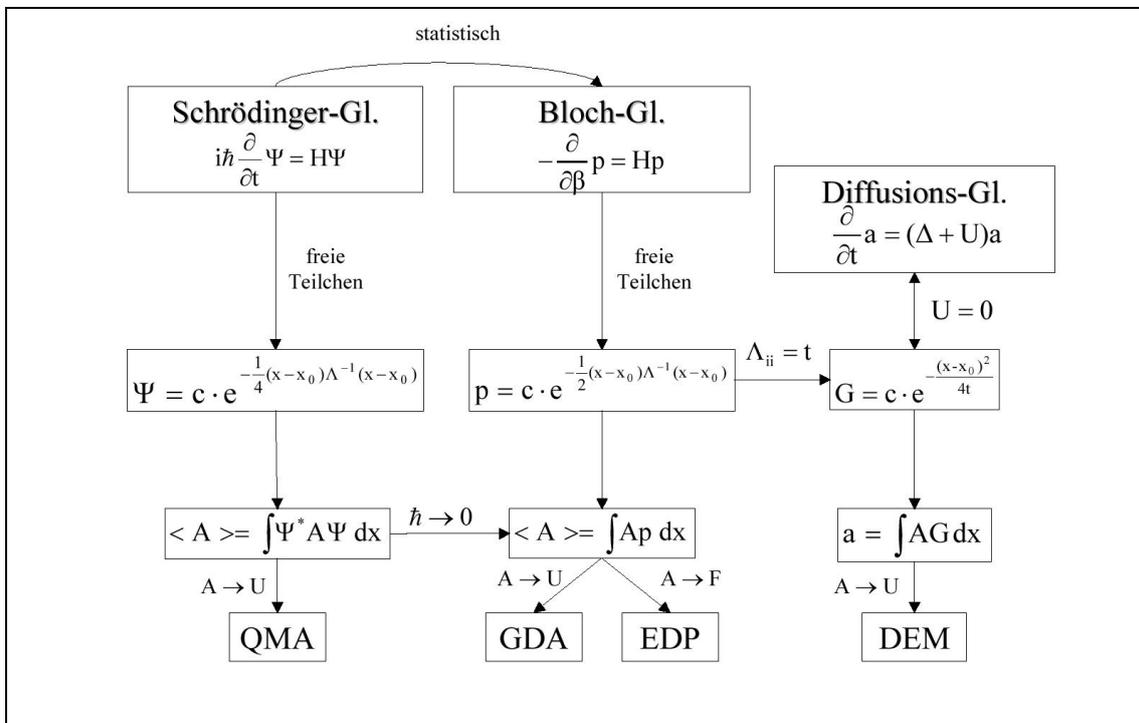


ABBILDUNG 1.1: Überblick über die Verfahren

Kapitel 2

Verfahren zur globalen Minimierung

2.1 Graduated-Non-Convexity (GNC)

Aus anschaulichen Gründen wird als erstes Verfahren das sogenannte Graduated-Non-Convexity vorgestellt. Es hat seinen Ursprung in der Bildverarbeitung und geht von einer sehr einfachen Idee aus. Man will sich dabei zu nutze machen, daß jedes lokale Minimum einer streng konvexen Funktion auch ihr globales Minimum ist. Man nennt eine Funktion $F : \mathfrak{R}^N \rightarrow \mathfrak{R}$ konvex, wenn für $\mathbf{x}, \mathbf{y} \in \mathfrak{R}^N$ und eine reelles $\lambda \in [0, 1]$ gilt

$$F(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda F(\mathbf{x}) + (1 - \lambda) F(\mathbf{y}) \quad (2.1)$$

Man nennt sie streng konvex, wenn die Ungleichung streng gilt. Nun sind die in dieser Arbeit betrachteten Potentialfunktionen nur in den seltensten Fällen konvex, so daß diese Eigenschaft einem nicht direkt weiterhilft.

Die Überlegung des GNC ist nun folgende: Wenn man eine nicht konvexe Funktion $U(\mathbf{x})$ gegeben hat, dann konstruiert man eine Folge von Funktionen $U_{\gamma_k}(\mathbf{x})$ mit den folgenden Eigenschaften:

- $U_{\gamma_0}(\mathbf{x})$ ist konvex
- $\lim_{k \rightarrow \infty} U_{\gamma_k}(\mathbf{x}) = U(\mathbf{x}) \quad \forall \mathbf{x} \in \mathfrak{R}^N$

Man kann dann für jedes k die Funktion $U_{\gamma_k}(\mathbf{x})$ lokal minimieren, so daß gilt

$$\nabla U_{\gamma_k}(\mathbf{x})|_{\mathbf{x}_k^0} = 0 \quad (2.2)$$

Dabei geht man jeweils von dem Minimum \mathbf{x}_{k-1}^0 des letzten Schrittes aus. Die Hoffnung ist, daß die \mathbf{x}_k^0 gegen das globale Minimum von U konvergieren.

Der kritische Punkt bei diesem Verfahren ist natürlich die Tatsache, daß man ohne genaue Kenntnisse über die Struktur von U keine geeignete Reihe von Funktionen konstruieren kann. Hat man solche Kenntnisse nicht, so braucht man allgemeine Transformationen, die Funktionen mit bestimmten Eigenschaften konvexifizieren können. Der Nachteil solcher allgemeiner Methoden ist wiederum, daß unter Umständen wichtige Eigenschaften

der ursprünglichen Funktion verloren gehen. Man kann dann versuchen, die Transformation an das Problem anzupassen, was sich jedoch als äußerst schwierig herausstellt, weil nur wenige solcher Transformationen in der Anwendung durchführbar sind. In den nächsten Kapiteln werden Verfahren vorgestellt, die prinzipiell auf allgemeine Potentialfunktionen angewendet werden können.

2.2 Diffusionsglättung

Die Diffusion ist bekannt als ein Prozeß, der zum Beispiel in einer Flüssigkeit Konzentrationsschwankungen ausgleicht, ohne dabei Masse zu erzeugen oder zu zerstören. Mathematisch wird das durch das Ficksche Gesetz ausgedrückt.

$$\vec{j} = -D \nabla u \quad (2.3)$$

Der Gradient der Konzentration u erzeugt einen Strom, der dem Gradienten entgegenwirkt. Dabei ist D der sogenannte Diffusionstensor, eine positiv definite, symmetrische Matrix. Falls der Strom \vec{j} und der Gradient ∇u parallel sind, so nennt man die Diffusion isotrop, man kann den Diffusionstensor D durch ein Skalar d ersetzen. Sind Strom und Gradient hingegen nicht parallel, so nennt man das anisotrop.

Die Tatsache, daß bei der Diffusion keine Masse erzeugt oder zerstört wird, wird durch die Kontinuitätsgleichung ausgedrückt.

$$\frac{\partial u}{\partial t} = -\operatorname{div} \vec{j} \quad (2.4)$$

Setzt man nun das Ficksche Gesetz in die Kontinuitätsgleichung ein, so erhält man die Diffusionsgleichung.

$$\frac{\partial u}{\partial t} = \operatorname{div} (D \cdot \nabla u) \quad (2.5)$$

Im Zusammenhang dieser Arbeit identifiziert man u mit der Potentialfunktion des betrachteten Systems. Die Wirkung der Diffusion ist dann eine Glättung der Funktion, bei der die Höhe der Berge und Täler auf der Potentialfläche mit der Zeit immer geringer wird. Die Lösung der Differentialgleichung läßt sich direkt angeben.

$$u(\mathbf{x}, t) = \det^{-\frac{1}{2}}(4\pi D_t) \int u(\mathbf{y}) e^{-\frac{1}{4}(\mathbf{x}-\mathbf{y})^T D_t^{-1} (\mathbf{x}-\mathbf{y})} d\mathbf{x} \quad (2.6)$$

Dabei gilt $D_t = D \cdot t$. Diese Lösung gilt allerdings nur für den homogenen Fall, daß der Diffusionstensor D über den ganzen Raum konstant ist. Im allgemeinsten Fall ist D vom Ort (inhomogen) oder sogar von u selber (nicht-linear) abhängig. Eine geschlossene Lösung der nicht-linearen Diffusionsgleichung

$$\frac{\partial u}{\partial t} = \operatorname{div} (D(u) \cdot \nabla u) \quad (2.7)$$

läßt sich aber nicht mehr angeben. Das ist insofern sehr bedauerlich, weil man durch eine nicht-lineare Diffusion eine Fülle von neuen Möglichkeiten gewinnen könnte. So wäre zum Beispiel eine Kanten-erhaltende Glättung möglich, durch die man die Bevorzugung von breiten Minima gegenüber schmalen bei der linearen Glättung verhindern könnte. Eine

weitere Verallgemeinerung ist durch die Hinzunahme eines Potentials in die Diffusionsgleichung möglich:

$$\frac{\partial u}{\partial t} = (\Delta + U)u \quad (2.8)$$

Die Lösung ist dann durch die Feynman-Kac-Formel gegeben, welche jedoch auch nur unter ganz bestimmten Voraussetzungen an das Potential einen Kern liefert, so daß man u als Integraltransformation darstellen kann. Im folgenden Abschnitt wird der einfachste Fall der Anwendung der Diffusionsglättung genauer beschrieben.

2.2.1 Diffusion Equation Method (DEM)

Bei der von Scheraga [6] vorgeschlagenen Diffusion Equation Method wird die Funktion $U(x)$ mittels der isotropen, linearen Diffusionsgleichung geglättet, so daß sich die Anzahl ihrer lokalen Minima reduziert. Um die geglättete Funktion zu erhalten, berechnet man die Lösung dieser Gleichung:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} - \Delta u(\mathbf{x}, t) = 0 \quad t \geq 0 \quad (2.9)$$

$$\text{mit } u(\mathbf{x}, 0) = U(\mathbf{x}) \quad (2.10)$$

Dabei spielt t die Rolle einer Glättungszeit. Für $t = 0$ erhält man die ursprüngliche Funktion und für $t > 0$ die geglättete Funktion $u(\mathbf{x}, t)$ als Lösung. Die Lösung läßt sich darstellen als Faltung mit einer Gauß-Funktion:

$$u(\mathbf{x}, t) = U(\mathbf{y}) \otimes G(\mathbf{x}, \mathbf{y}, t) = (4\pi t)^{-\frac{d}{2}} \int U(\mathbf{y}) e^{-\frac{(\mathbf{x}-\mathbf{y})^2}{4t}} d\mathbf{y} \quad (2.11)$$

Dabei ist d die Dimension von \mathbf{x} bzw. \mathbf{y} . Man kann hier schön sehen, daß das Ergebnis eine über die Breite $4t$ der Gauß-Funktion räumlich gemittelte Funktion $U(\mathbf{x})$ ist. Je größer die Glättungszeit t ist, desto größer ist das Gebiet, über das gemittelt wird, und die Glättung wird stärker.

Um nun mittels DEM das globale Minimum einer Funktion zu finden geht man wie folgt vor:

DEM-Algorithmus

Schritt 1: Setze $k = 0$. Wähle $t_0 > 0$ genügend groß, so daß die Zahl der Minima stark reduziert wird, und ein zufälliges $\mathbf{x}_0(t_0)$.

Schritt 2: Ausgehend von $\mathbf{x}_0(t_k)$ berechnet man $\mathbf{x}_{min}(t_k)$ gemäß

$$\nabla_{\mathbf{x}} u(\mathbf{x}, t)|_{\mathbf{x}=\mathbf{x}_{min}(t_k)} = 0 \quad (2.12)$$

Schritt 3: Falls $t = 0$ gilt, ist $\mathbf{x}_{min}(t_k)$ das gefundene Minimum. Ansonsten setze

$$k = k + 1$$

$$t_k = t_{k-1} - \Delta t$$

$$\mathbf{x}_0(t_k) = \mathbf{x}_{min}(t_{k-1}) \quad (2.13)$$

und gehe zu **Schritt 2.**

Mit diesem Algorithmus verfolgt man eine Trajektorie $\mathbf{x}_{min}(t)$ in dem Sinne, daß das zur Zeit t_0 gefundene Minimum sich stetig durch die Zeit zu einem Minimum bei $t = 0$ bewegt. Für den eindimensionalen Fall $d = 1$ kann man zeigen, daß dies stets der Fall ist. In höheren Dimensionen allerdings können diese Trajektorien enden, bevor $t = 0$ erreicht worden ist. Das bedeutet, man findet in der Umgebung eines zur Zeit $t + \epsilon$ ($\epsilon > 0$) vorhandenen Minimums zur Zeit t kein Minimum mehr. Der Algorithmus wird sich dann ein weiter entferntes Minimum suchen, welches auf einer anderen Trajektorie liegt. Das führt zu einer Unstetigkeit in $\mathbf{x}_{min}(t)$, die sich messen läßt, wenn man eine Geschwindigkeit der Trajektorie wie folgt definiert:

$$v_T(t_k) = \left| \frac{\mathbf{x}_{min}(t_k) - \mathbf{x}_{min}(t_{k-1})}{\Delta t} \right| \quad (2.14)$$

Berechnet man diese Geschwindigkeit nach jeder Minimierung, so kann man den Übergang zu einer anderen Trajektorie durch einen sprunghaften Anstieg von $v_T(t_k)$ erkennen. Zum Beispiel zeigt Abbildung 2.1 den Verlauf von $v_T(t)$ für eine Anwendung des obigen

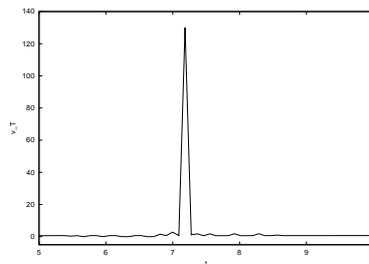


ABBILDUNG 2.1: Wechsel der Trajektorie

Algorithmus auf ein 15 Atom Lennard-Jones-Cluster. Man sieht deutlich den Sprung bei $t \approx 7.2$, der einen Wechsel zu einer anderen Trajektorie anzeigt. Tritt ein solcher Sprung auf, so wird die Trajektorie nicht stetig von $t = t_{max}$ bis $t = 0$ verfolgt und das Ergebnis wird eher zufällig. Scheragags Idee, das globale Minimum durch eine Verfolgung von $\mathbf{x}_{min}(t)$ ausgehend von einer Glättungszeit, wo nur ein Minimum existiert, zu finden, erweist sich als nicht durchführbar. Wie sich zeigen wird, treten solche Sprünge bei Lennard-Jones-Clustern ab einer gewissen Größe immer auf, so daß das Resultat nicht mehr eindeutig ist. Dennoch stellt das Verfahren eine sehr einfache und schnelle Möglichkeit dar, kleine und einfache Probleme mit gutem Erfolg zu minimieren.

Zur Veranschaulichung der Funktionsweise des DEM-Algorithmus wird im nächsten Abschnitt seine Anwendung auf eine eindimensionale Funktion beschrieben.

Ein eindimensionales Beispiel

In der Diffusion Equation Method wird das Potential als Folge einer Anwendung der Diffusionsgleichung mit einem Gauß-Kern gefaltet. Diese Faltung bewirkt eine Glättung und verringert dadurch die Anzahl der lokalen Minima der Funktion. Um die wichtigsten Eigenschaften dieser Glättung und auch die Probleme, die dadurch entstehen, zu veranschaulichen, ist es sinnvoll, ein eindimensionales Beispiel zu betrachten. Dazu modelliert

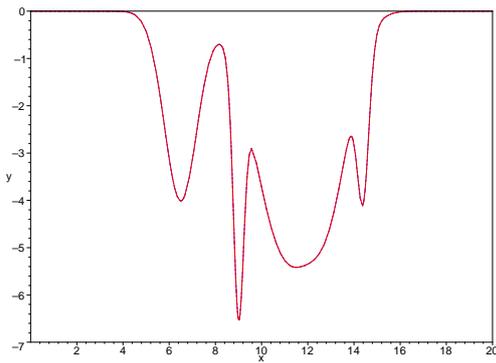


ABBILDUNG 2.2: Die Funktionen $f(x)$ (---) und $\tilde{f}(x)$ (---) bei $t = 0$

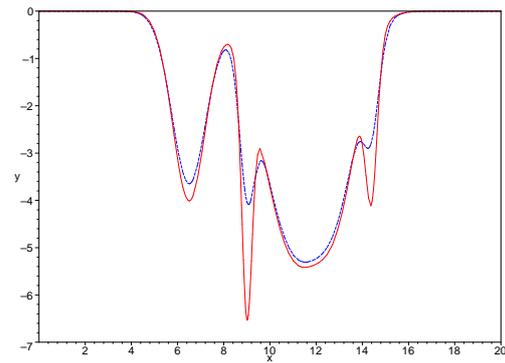


ABBILDUNG 2.3: Die Funktionen $f(x)$ (---) und $\tilde{f}(x)$ (---) bei $t = 0.1$

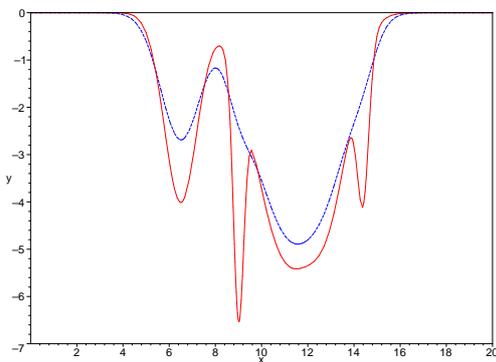


ABBILDUNG 2.4: Die Funktionen $f(x)$ (---) und $\tilde{f}(x)$ (---) bei $t = 0.5$

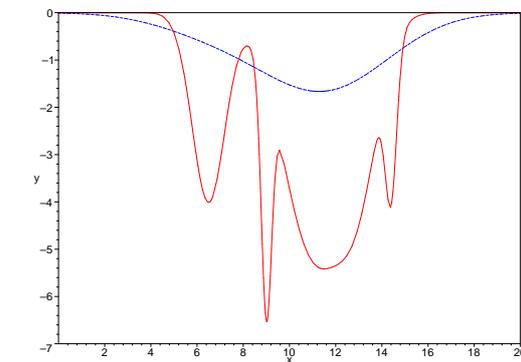


ABBILDUNG 2.5: Die Funktionen $f(x)$ (---) und $\tilde{f}(x)$ (---) bei $t = 10.0$

man eine Funktion mit mehreren Minima, deren Faltung mit dem Gauß-Kern leicht analytisch berechenbar ist. Es bietet sich daher an, die Funktion selbst als eine Summe über Gauß-Funktionen darzustellen:

$$f(x) = \sum_{i=1}^L G(x, x_i^0, a_i, b_i) \quad \text{mit} \quad G(x, x_i^0, a_i, b_i) = a_i e^{-b_i(x-x_i^0)^2} \quad (2.15)$$

Dann berechnet sich die Faltung einfach nach:

$$\tilde{f}(x) = (\pi t)^{-\frac{1}{2}} \int f(y) e^{-\frac{(y-x)^2}{t}} dy \quad (2.16)$$

$$= \sum_{i=1}^L a_i (1 + b_i t)^{-\frac{1}{2}} e^{-\frac{b_i(x-x_i^0)^2}{1+b_i t}} \quad (2.17)$$

Den Effekt der Glättung auf die Funktion $f(x)$ kann man dann gut anhand von Graphen erkennen. Als Beispielfunktion wird eine Summe aus fünf Gauß Funktionen gewählt. Die zugehörigen Parameter stehen in Tabelle 2.1. In den Abbildungen 2.2 bis 2.5 sieht man die Funktion $f(x)$ und $\tilde{f}(x)$ im Vergleich bei verschiedenen Werten von t . Die wichtigste Eigenschaft, die Verringerung der Zahl der Minima, wird sehr deutlich und kann auch theoretisch gezeigt werden:

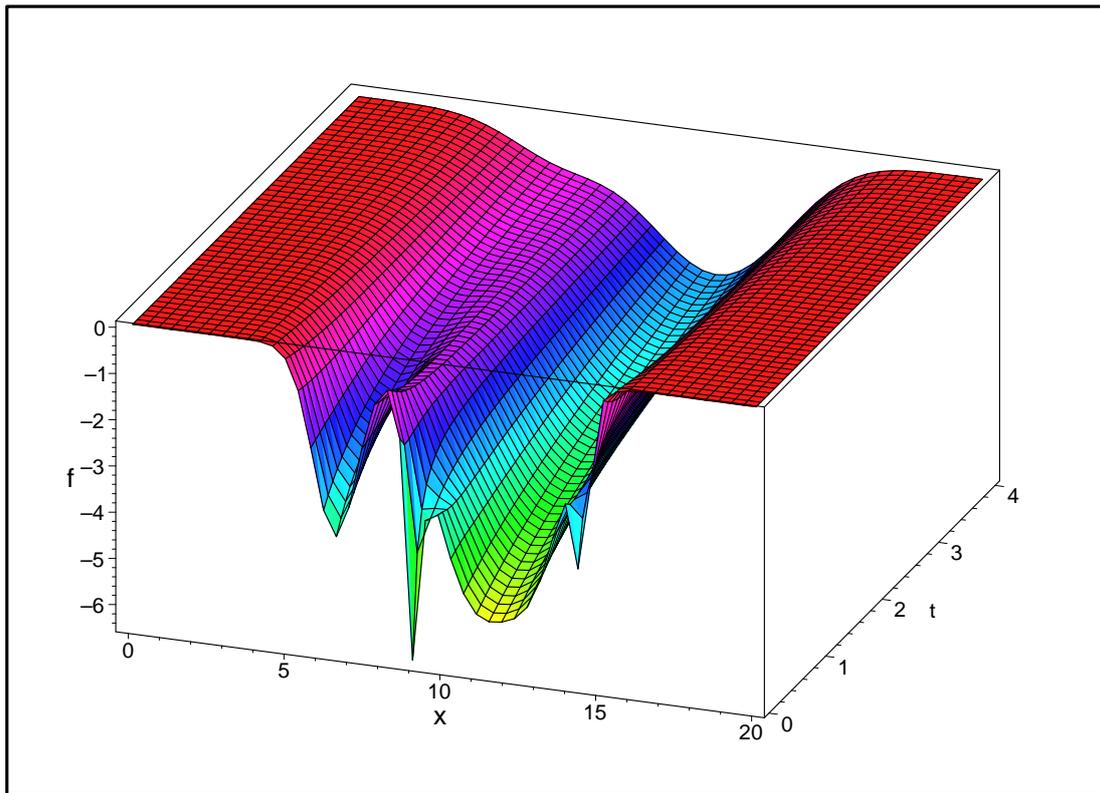


ABBILDUNG 2.6: Die Funktion $\tilde{f}(x, t)$

Bezeichnet man mit $\langle f \rangle(\omega)$ die Fourier-Transformierte von $f(x)$, also

$$\langle f \rangle(\omega) = \frac{1}{2\pi} \int f(x) e^{i\omega x} dx, \quad (2.18)$$

und ist

$$\tilde{f}(x) = f(y) \otimes g(x - y) = \int f(y) g(x - y) dx \quad \text{mit} \quad g(x) = \frac{1}{\sqrt{\pi t}} e^{-\frac{x^2}{t}} \quad (2.19)$$

so gilt nach [15]

$$\langle \tilde{f} \rangle(\omega) = \langle f(y) \otimes g(x - y) \rangle \quad (2.20)$$

$$= \langle f \rangle(\omega) \cdot \langle g \rangle(\omega) \quad (2.21)$$

$$= e^{-\frac{1}{4}t\omega^2} \langle f \rangle(\omega) \quad (2.22)$$

i	1	2	3	4	5
a_i	-4.0	-5.0	-5.0	-4.0	-3.0
b_i	1.0	10.0	0.3	0.6	8.0
x_i^0	6.5	9.0	11.0	13.0	14.4

TABELLE 2.1: Parameter für die Funktion $f(x)$

Offensichtlich gilt

$$\lim_{t \rightarrow 0} \langle \tilde{f} \rangle(\omega) = \langle f \rangle(\omega) \quad (2.23)$$

Für größer werdendes t unterdrückt der Faktor $e^{-\frac{1}{4}t\omega^2}$ die hohen Frequenzen von f . Genau das ist der Grund für die glättende Eigenschaft der Faltung. Die hohen Frequenzen werden immer stärker unterdrückt, so daß lokale Strukturen der Funktion immer mehr verschwinden und bei großem t ausschließlich die globalen Tendenzen sichtbar werden.

Gleichzeitig ist diese Art der Glättung aber auch das Problem bei Minimierungsverfahren. Denn, wie man den Abbildungen entnehmen kann, bleibt bei starker Glättung nicht unbedingt das globale Minimum als das dominanteste übrig, sondern das breiteste Minimum. In diesem Beispiel ist das globale Minimum sogar schon bei einem Wert von $t = 0.1$ fast ganz verschwunden, so daß ein Algorithmus, der wie die DEM einen Weg von $t > 4.0$ bis $t = 0$ durch die Minima verfolgen würde, hier im breiteren und höher liegenden Minimum landen würde. Abbildung 2.6 verdeutlicht das, indem sie die Funktion \tilde{f} in Abhängigkeit von x und t zeigt. Der Algorithmus würde bei $t > 4.0$ im Minimum beginnen und bei kleiner werdendem t auch immer dieses eine Minimum weiterverfolgen, bis er im Minimum bei $x \approx 11.9$ landet.

Im Hinblick auf die Frage, welches Minimum das physikalisch sinnvollste ist, spielt die Breite allerdings auch eine entscheidende Rolle. Betrachtet man nämlich die freie Energie $F = E - TS$, so beeinflußt für endliche Temperaturen der Entropiefaktor die Bevorzugung der Minima. Dieser Faktor ist für breite Minima größer und kann deshalb dazu führen, daß physikalisch tatsächlich das höher liegende, breite Minimum das bevorzugte ist. Bei der Faltung mit einem Gauß-Kern, bei dem die Breite der Verteilung durch eine einfache Vorschrift „künstlich“ geregelt wird, ist der Zusammenhang des Glättungsparameters mit der physikalischen Temperatur aber nicht klar, so daß man keine Aussage darüber machen kann, inwieweit das physikalische Verhalten richtig wiedergegeben wird. Denn da bei der DEM kein expliziter Entropieterm vorkommt, entsteht die Dominanz der breiteren Minima eher zufällig durch eine ungenügende Näherung. Dieses Verhalten wird vor allem im Vergleich mit den physikalisch motivierten Annealing Verfahren deutlich. Deshalb wird in Kap.2.5 noch einmal ausführlicher darauf eingegangen.

2.2.2 Der Basin-Hopping-Algorithmus

Ein sehr erfolgreicher Algorithmus bei der Suche nach der energetisch minimalsten Konfiguration von Lennard-Jones- und TIP4P-Wasser-Clustern ist der sogenannte Basin-Hopping-Algorithmus von Wales und Doye [8]. Er stellt eine Kombination von zufälligen Sprüngen nach dem Monte-Carlo Prinzip und lokalen Minimierungen dar. Die grundlegende Idee ist die, daß man die Potentialfläche auf eine Art transformiert, die zwar die Energiebarrieren entfernt, aber die Energien aller (lokalen) Minima beibehält. Die folgende Gleichung stellt eine solche Transformation dar

$$\tilde{U}(\mathbf{x}) = \min_{\mathbf{x} \in G} \{U(\mathbf{x})\} \quad (2.24)$$

Man weist also jedem Ort den Energiewert des Minimums zu, welches man durch eine lokale Minimierung von diesem Ort aus erreicht. G bezeichnet also die Umgebung, aus der jede lokale Minimierung im selben Minimum endet. Abbildung 2.7 zeigt an einem eindimensionalen Beispiel, wie die transformierte Funktion \tilde{U} aussieht. Auf einer solchen

Potentialfläche kann das System ungehindert von den Energiebarrieren den Konfigurationsraum durchlaufen. Die Suche wird nur von der relativen Lage der einzelnen Minima bestimmt.

Gl.2.24 hat aber den Nachteil, daß man $\tilde{U}(\mathbf{x})$ nicht analytisch berechnen kann. Man muß also jedesmal einen Minimierungsalgorithmus durchführen, um den Wert an einer Stelle \mathbf{x} zu berechnen. Das bedeutet natürlich einen erheblichen numerischen Aufwand, weswegen für die Suche auf der so umgeformten Potentialfläche nur ein Verfahren in Frage kommt, welches ausschließlich den Energiewert der jeweils betrachteten Konfiguration benötigt. Deswegen benutzt man ein standard Monte-Carlo Verfahren mit Metropolis-Kriterium.

Im Detail geht man also wie folgt vor: Man wählt eine zufällige Startkonfiguration \mathbf{x}_0 und berechnet $\tilde{U}(\mathbf{x}_0)$, indem man von \mathbf{x}_0 ausgehend eine lokale Minimierung durchführt. Danach verschiebt man die Positionen der einzelnen Atome um einen zufälligen Wert $\boldsymbol{\lambda}$ und berechnet wiederum die minimierte Energie von der neuen Position aus. Dann entscheidet man, ob die neue Konfiguration beibehalten oder wieder verworfen wird, bevor man wieder eine zufällige Verschiebung vornimmt. Diesen Vorgang wiederholt man eine vorgegebene Anzahl von Schritten. Formal stellt sich der Algorithmus wie folgt dar:

Basin-Hopping-Algorithmus

Schritt 1: Setze $k = 0$, wähle einen Startpunkt \mathbf{x}_0 und berechne $\tilde{U}_0 = \tilde{U}(\mathbf{x}_0)$

Schritt 2: Setze $k = k + 1$, wähle ein $\boldsymbol{\lambda}$ mit $\lambda_i \in [-c, c]$, $i = 1, \dots, N$, $c \in \Re$ und setze

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \boldsymbol{\lambda} \quad (2.25)$$

Schritt 3: Berechne $\tilde{U}_k = \tilde{U}(\mathbf{x}_k)$, falls

$$e^{-\beta(U_k - U_{k-1})} > r_{random} \quad (2.26)$$

erfüllt ist, wird \mathbf{x}_k akzeptiert, ansonsten setze $\mathbf{x}_k = \mathbf{x}_{k-1}$

Schritt 4: Falls $k = k_{max}$ stoppe, sonst gehe zu Schritt 2

k_{max} bezeichnet hier die maximale Anzahl von Schritten, und r_{random} ist eine Zufallszahl aus $[0, 1]$. Im Gegensatz zum standard Monte-Carlo Verfahren rechnet man allerdings mit einer konstanten Temperatur, $\beta = const$. Es handelt sich also nicht um ein Simulated-Annealing Verfahren, so daß man sich zusätzlich stets die Konfiguration mit der niedrigsten Energie merken muß. Man wählt β so, daß Schritte, die eine höhere Energie zur Folge haben, auch akzeptiert werden können, wenn der Energieunterschied nicht zu groß ist. Den tatsächlich besten Wert für β kann man aber nur durch einige Tests ermitteln. Ist der Algorithmus beendet, minimiert man die Konfiguration mit der niedrigsten Energie gegebenenfalls mit höherer numerischer Genauigkeit nochmals, um das Ergebnis zu erhalten.

Man muß sich bei diesem Verfahren auf eine konstante Temperatur beschränken, da man in einer angemessenen Zeit zu wenige Schritte machen kann, als daß man von einem equilibrierten Zustand sprechen könnte. Das ist aber eine wichtige Voraussetzung für ein

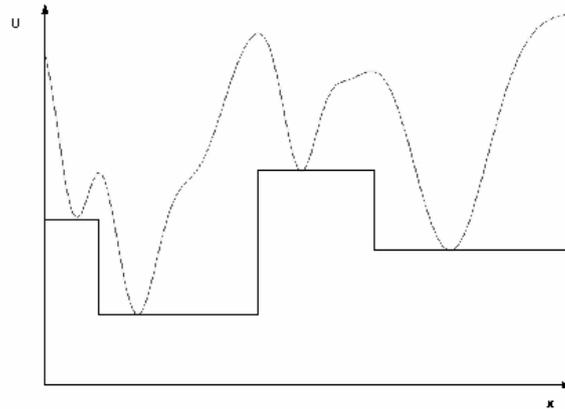


ABBILDUNG 2.7: Schematische Darstellung von \tilde{U}

sinnvolles Abkühlen des Systems. Das Monte-Carlo Verfahren mit konstanter Temperatur ist jedoch nur ein Suchverfahren, welches durchaus Energiebarrieren überwindet, aber vor allem zu Konfigurationen mit niedrigeren Energie strebt. Das macht einen gravierenden Nachteil des Verfahrens deutlich. Man hat keine Informationen darüber, wann der Algorithmus seine beste Konfiguration finden wird. Deshalb hat man keinerlei Anhaltspunkt über die Zahl der Schritte, die sinnvollerweise durchgeführt werden sollte. Das bedeutet vor allem auch, daß der Algorithmus einen wesentlichen Teil der Laufzeit überhaupt nichts Sinnvolles tut, weil er eine große Zahl an Monte-Carlo Schritten wieder verwirft. Alle anderen vorgestellten Algorithmen werden mit der Ausgabe der energetisch tiefsten Konfiguration beendet. Trotzdem ist der Basin-Hopping-Algorithmus zur Zeit noch das Verfahren, welches die besten Ergebnisse bei der Minimierung von Lennard-Jones- und TIP4P-Wasser-Clustern liefert.

Der Grund, warum dieses Verfahren als Unterkapitel der Diffusionsglättung erscheint, ist die Ähnlichkeit seiner Potentialtransformation mit der, die man als Grenzfall einer nicht-linearen Diffusion erhält. Die Potentialfunktion wird in eine Treppenfunktion umgewandelt, deren Stufenhöhen durch die Energien der Minima des Potentials gegeben sind. Der Einfachheit halber ist die folgende Darstellung für den eindimensionalen Fall, gilt aber auch für höhere Dimensionen [12]. Als Lösung der Gleichung

$$\frac{\partial}{\partial t} u = \operatorname{div} (g(|\nabla u|^2) \nabla u) \quad \text{mit} \quad u(x, 0) = U(x) \quad (2.27)$$

mit dem Perona-Malik Filter (siehe [12])

$$g(s^2) = \frac{1}{1 + \frac{s^2}{\gamma^2}} \quad (2.28)$$

ergibt sich ebenfalls eine Treppenfunktion \tilde{u} , wenn man die Grenzübergänge

$$\tilde{u} = \lim_{t \rightarrow \infty} \lim_{\gamma \rightarrow 0} u \quad (2.29)$$

macht. Diese Funktion \tilde{u} hat allerdings Stufenhöhen mit den Energien aller Minima und Maxima des Potentials, wobei sich die Übergänge an den Wendepunkten von U befinden.

Den Effekt des Filters sieht man am besten, wenn man Gl.2.27 mithilfe des Flusses $\Phi(s) = sg(s^2)$ umschreibt zu:

$$\frac{\partial}{\partial t}u = \Phi'(\nabla u)\Delta u \quad (2.30)$$

Für $|\nabla u| > \gamma$ ist $\Phi' < 0$ und man erhält eine Kantenverstärkung. Läßt man nun $\gamma \rightarrow 0$ gehen, so werden alle Kanten unabhängig von ihrer Steigung verstärkt. Der Übergang $t \rightarrow \infty$ sorgt für die Form der Treppenstufen, da dann für $\frac{\partial^2}{\partial x^2}u = 0$ die Steigung $|\nabla u| \rightarrow \infty$ geht und sonst $|\nabla u| = 0$ gilt.

Wie schon gesagt, ist die Funktion \tilde{u} nicht identisch mit der im Basin-Hopping benutzten \tilde{U} , da letztere den Vorteil hat, daß ihr die hohen Energiebarrieren fehlen. Der Vergleich zeigt aber trotzdem, wie eng alle Verfahren miteinander verwandt sind, wenn es gelingt, die benutzten Transformationen als Lösungen verschiedener Spezialfälle der gleichen Differentialgleichung zu verstehen.

2.3 Simulated Annealing

Eine der ältesten Methoden zur globalen Minimierung ist das sogenannte Simulated Annealing. Es beruht auf der Tatsache, daß der Prozeß des Abkühlens bei geeigneter Wahl der Kühlrate mit hoher Wahrscheinlichkeit im globalen Energieminimum endet.

Bei der Computer-Simulation bedient man sich dabei zweier verschiedener Algorithmen, um das System von hohen Temperaturen, bei denen sich die einzelnen Atome schnell und ungehindert von Potentialbarrieren bewegen, bis nahe an den absoluten Nullpunkt, bei dem das System eine beinahe starre Konfiguration einnimmt, abzukühlen. Diese Algorithmen dienen im Prinzip dazu, typische Konfigurationen des Systems bei der momentanen Temperatur zu liefern. Und zwar sollten die Konfigurationen mit der Wahrscheinlichkeit auftreten, die man in diesem physikalischen Zusammenhang auch erwartet. Man kontrolliert die Temperatur meist durch ein Wärmebad. Das System kann mit diesem viel größeren Wärmebad Energie austauschen und wird sich deshalb an seine Temperatur angleichen. Dabei ist zu beachten, daß weder die Gesamtenergie des (kleinen) Systems noch seine kinetische Energie konstant sind, da die Temperatur um die des Wärmebads schwanken kann. Ein solches System gehorcht der Gibbs/Boltzmann-Wahrscheinlichkeitsverteilung:

$$p_B(\mathbf{x}) = \frac{e^{-\beta U(\mathbf{x})}}{\int e^{-\beta U(\mathbf{x})} d\mathbf{x}} \quad (2.31)$$

Dabei ist $U(\mathbf{x})$ die zur räumlichen Konfiguration gehörende potentielle Energie und $\beta = \frac{1}{k_b T}$. Aus der Gleichung liest man ab, daß es immer unwahrscheinlicher wird, eine Konfiguration zu finden, je höher ihre potentielle Energie ist. Bei hohen Temperaturen T , also kleinem β , kann das System jedoch solche Gebiete mit hoher Energie überwinden, da die Wahrscheinlichkeit noch groß genug ist. Senkt man dann die Temperatur ab, so entstehen auf der Potentialfläche Gebiete, die durch Berge hoher Energien voneinander getrennt sind, so daß das System, in einem dieser Täler gefangen, nicht mehr den ganzen Konfigurationsraum erreichen kann. Anders ausgedrückt ist die Zeitskala für Bewegungen innerhalb dieser Region viel kleiner als die für Bewegungen aus diesem Gebiet hinaus. Man spricht dann von einem metastabilen Makrozustand, bei dem die einzelnen Atome immer noch fluktuieren. Die Wahrscheinlichkeit, daß ein bestimmter Makrozustand α eingenommen

wird, berechnet sich entsprechend der der Gibbs/Boltzmann-Verteilung wie folgt:

$$p_\alpha = \frac{\int_\alpha e^{-\beta U(\mathbf{x})} d\mathbf{x}}{Z} \quad Z = \int e^{-\beta U(\mathbf{x})} d\mathbf{x} \quad (2.32)$$

Senkt man die Temperatur weiter ab, bis sie fast den absoluten Nullpunkt erreicht, so befindet sich das System in einem Minimum der Potentialfunktion und kann daraus auch nicht mehr entweichen, da jede kleine Veränderung der Konfiguration einen Anstieg der potentiellen Energie zur Folge hätte, der die Wahrscheinlichkeit für diese Konfiguration stark herabsenkt. Man hat dann eine stabile Konfiguration des Systems, in der die einzelnen Atome nicht mehr fluktuieren.

Die Art, wie die Algorithmen nun die statistischen Eigenschaften eines Systems im Wärmebad verwirklichen, ist unterschiedlich und soll im folgenden erläutert werden.

2.3.1 Monte-Carlo (MC)

Bei der Monte-Carlo-Methode besteht ein Schritt aus einer zufälligen Bewegung aller Atome des Systems. Die neue Konfiguration wird dann entweder akzeptiert oder wieder verworfen, und zwar in Abhängigkeit vom Unterschied der potentiellen Energien ΔU der Ausgangsposition und der neuen Position. Benutzt man die oben erläuterte Statistik, so berechnet sich die sogenannte Metropolis-Wahrscheinlichkeit für Beibehaltung der neuen Konfiguration nach

$$p = \min \left[1, e^{-\beta \Delta U} \right] \quad (2.33)$$

Bei hohen Temperaturen haben dann auch Bewegungen zu höheren potentiellen Energien eine gute Chance, akzeptiert zu werden, wodurch auch Berge in der Energielandschaft überwunden werden können. Bei niedrigen Temperaturen wirkt sich das Verfahren wie das eines steilsten Abstiegs aus, so daß das System dann in das lokale Minimum fällt.

Der offensichtliche Nachteil dieses Verfahrens ist die Durchführung von Schritten, die dann wieder verworfen werden. Diese schlechte Nutzung der Rechenzeit tritt vor allem bei niedrigen Temperaturen auf, wo nur noch Positionen mit niedrigerem Potential akzeptiert werden („critical slowing down“). Deswegen und wegen der hohen Zahl an Schritten, die erforderlich sind, um den Konfigurationsraum genügend zu durchlaufen, ist dieses Verfahren bei der Suche des globalen Minimums nicht sehr erfolgreich. Es gibt allerdings Variationen des Algorithmus, die die Ergebnisse erheblich verbessern. So berichten Andricioaei und Straub [4] zum Beispiel eine deutliche Verbesserung des Monte-Carlo-Verfahrens, wenn man statt des Metropolis-Kriteriums ein anderes verwendet, welches auf der sogenannten Tsallis-Statistik basiert. Eine andere Möglichkeit stellt die Durchführung von Monte-Carlo Schritten auf einer transformierten Potentialfunktion dar, wie sie in Kap.2.2.2 vorgestellt wurde.

2.3.2 Molekül-Dynamik (MD)

Verwendet man zur Durchsuchung des Konfigurationsraumes das Verfahren der Moleküldynamik, so simuliert man das Verhalten der Natur. Die Atome bewegen sich in den durch das Potential verursachten Kraftfeldern und haben eine durch die Temperatur vorgegebene kinetische Energie. Physikalisch ausgedrückt bedeutet das, man integriert die Newton'schen

Bewegungsgleichungen:

$$\begin{aligned} m \cdot \ddot{\vec{x}} &= \vec{F} \\ \dot{\vec{x}} &= \vec{v} \end{aligned}$$

Dabei ist $\vec{F} = -\nabla U$. Gibt man die Anfangspositionen, Anfangsgeschwindigkeiten und die Potentiale vor, so stellt sich nach einigen Simulationsschritten ein thermisches Equilibrium ein, so daß für den Zusammenhang von Temperatur und Energie nach dem Gleichverteilungssatz der Thermodynamik gilt

$$E_{kin} = \frac{f}{2} N k_b T$$

f bezeichnet hier die Anzahl der Freiheitsgrade der N Atome. Die Summe von kinetischer und potentieller Energie, also die Gesamtenergie, bleibt dabei konstant, während die Temperatur starken Schwankungen unterliegen kann. Die Geschwindigkeiten der einzelnen Atome gehorchen dann einer Maxwell-Boltzmann-Verteilung:

$$f(\vec{v}) = 4\pi \left(\frac{m}{2\pi k_b T} \right)^{\frac{3}{2}} v^2 e^{-\frac{mv^2}{2k_b T}} \quad (2.34)$$

Ein solches, thermisch isoliertes System ist allerdings ungeeignet, um durch Abkühlung das globale Energieminimum zu finden, da jeder Mikrozustand, also jede räumliche Konfiguration gleich wahrscheinlich ist. Man braucht das schon oben erläuterte System im Temperaturbad, in dem die Temperatur nahezu konstant ist (kleine Schwankungen treten auch hier auf). Um das bei der Simulation zu realisieren, muß man in einigen Zeitschritten die kinetische Energie und damit die Temperatur verändern. Bei einem einfachen System aus ungebundenen Atomen, in dem jedes Atom 3 Freiheitsgrade hat, könnte das wie folgt aussehen:

Die zum Zeitschritt k herrschende Temperatur T^k berechnet sich wie oben schon gesehen zu

$$T^k = \frac{2}{3} \frac{E_{kin}^k}{k_b N}$$

Wobei die kinetische Energie des Systems durch

$$E_{kin}^k = \frac{1}{2} \sum_{i=1}^N m_i v_i^2$$

gegeben ist. Um nun die Temperatur genau auf den gewünschten Wert T_{TB} , also die Temperatur des Temperaturbades zu bringen, müßte man die Geschwindigkeiten der Atome mit dem Faktor

$$\gamma = \sqrt{\frac{T_{TB} 3N}{2E_{kin}^k}}$$

multiplizieren. Um allerdings kleine Schwankungen um die Temperatur T_{TB} zu gestatten, kann man stattdessen die Geschwindigkeiten der Atome in einigen Zeitschritten mit dem Faktor 0.999 multiplizieren, falls $\gamma < 1.0$ gilt, oder mit 1.001, falls $\gamma > 1.0$ ist. Ein so simuliertes System gehorcht der oben beschriebenen Gibbs/Boltzmann-Verteilung und hat größere Aufenthaltswahrscheinlichkeiten in den Potentialtälern als auf den Potentialbergen. Eine andere Möglichkeit zur Regulierung der Temperatur ist die Verwendung von Thermostaten, die als Zwangsbedingungen in den Lagrange-Gleichungen auftreten werden.

Das Abkühlen des Systems wird nun realisiert, indem man nach einer bestimmten Zahl von Zeitschritten, die Temperatur des Temperaturbades um einen kleinen Faktor herabsenkt. Ist man bei einer sehr kleinen Temperatur angekommen, so reicht die kinetische Energie der Atome nicht mehr aus, um aus dem momentanen Potentialtal herauszugelangen. Das dann gefundene Minimum ist bei der richtigen Wahl der Kühlrate hoffentlich das globale Minimum des Systems.

Abschätzung der Simulationszeit

Die große Schwierigkeit des Verfahrens ist allerdings die Wahl eben dieser Kühlrate. Eine typischen Wahl wäre eine nach einer festen Anzahl von Schritten um einen Faktor $\alpha < 1$ sinkende Temperatur:

$$T_{k+1} = \alpha \cdot T_k \quad (2.35)$$

Obwohl eine solche Wahl in vielen Fällen funktioniert, ist sie doch vollkommen willkürlich. Man muß sicherstellen, daß das System während des Abkühlens genug Zeit hat, den gesamten zur Verfügung stehenden Konformationsraum zu durchsuchen. Man kann nach Straub [1] eine einfache eindimensionale Abschätzung der gesamten benötigten Simulationszeit wie folgt machen:

Setzt man das globale Minimum als den Nullpunkt der Energie fest und bezeichnet mit E_{max} die Höhe der größten Energiebarriere, so beginnt man die Simulation bei einer Temperatur T_{max} , für die gilt:

$$k_b T_{max} = \frac{1}{\beta_{min}} \geq E_{max} \quad (2.36)$$

Dann kann das System ungehindert alle möglichen räumlichen Anordnungen einnehmen, denn es gilt:

$$e^{-\beta_{min} E_{max}} = O(1) \quad (2.37)$$

Auf der anderen Seite muß man am Ende der Simulation die Temperatur so weit abgesenkt haben, daß das System im Minimum gefangen ist und keine Energiebarriere mehr überwinden kann. Es muß also gelten:

$$k_b T_{min} = \frac{1}{\beta_{max}} < \Delta E \quad (2.38)$$

Dabei ist ΔE der Energieunterschied des globalen Minimums und des nächst höheren Minimums. Dabei sollte diese Ungleichung so erfüllt sein, daß die Wahrscheinlichkeit X_0 , das System im globalen Minimum zu finden, im Verhältnis zur Wahrscheinlichkeit $1 - X_0$, das System woanders zu finden, groß ist. Nimmt man an, daß sich das System bei niedriger Temperatur ausschließlich in den untersten beiden Energieminima befinden kann, so soll

$$\frac{X_0}{1 - X_0} = \frac{e^{-\beta_{max} E_{min}}}{e^{-\beta_{max} (E_{min} + \Delta E)}} = e^{\beta_{max} \Delta E} \quad (2.39)$$

groß sein, was man auch als $X_0 \approx 1$ schreiben kann.

Um nun den gesamten Konfigurationsraum möglichst gut zu durchsuchen, sollte das System bei jeder Temperatur genug Zeit t verbringen, um jede Energiebarriere überwinden zu können. Bezeichnet man mit τ_s die charakteristische Zeitskala für Vibrationen, in

welcher dann auch die Simulationsschrittweite gewählt wird, so kann man diese Forderung schreiben als:

$$t = \tau_s e^{\beta E_{max}}$$

Anders ausgedrückt, sollte das System in jedem Temperaturintervall $(\beta, \beta + \Delta\beta)$ mindestens folgende Zeit verbringen:

$$dt = E_{max} \tau_s e^{\beta E_{max}} d\beta$$

Integriert man diese Gleichung von β_{min} bis β_{max} , so erhält man eine untere Grenze für die gesamte Simulationszeit bzw. die Anzahl der Simulationsschritte $N_{sim} = \frac{t_{sim}}{\tau_s}$. Daraus ergibt sich folgende Proportionalität zwischen der Anzahl der Simulationsschritte und dem Verhältnis der größten Energiebarriere und dem kleinsten Energieabstand:

$$N_{sim} \propto \left(\frac{X_0}{1 - X_0} \right) e^{\frac{E_{max}}{\Delta E}}$$

Die Dauer einer Simulation, die mit der Wahrscheinlichkeit X_0 das globale Minimum der Energiefunktion finden soll und eine konstante Absenkung der Temperatur verwendet, hängt also nicht nur von der Größe der unteren oder oberen Energieskala ab, sondern von deren Unterschied. Eine grobe Abschätzung liefert eine Vorstellung der Folge dieser Tatsache. Nimmt man ein Verhältnis von $\frac{E_{max}}{\Delta E} = 10$ an und verlangt $X_0 = 0.1$, so erhält man

$$N_{sim} \propto 2 \cdot 10^3$$

Erhöht sich das Verhältnis $\frac{E_{max}}{\Delta E}$ zum Beispiel auf das Doppelte, so ergibt sich bereits

$$N_{sim} \propto 5 \cdot 10^7$$

Man erkennt, daß die Zahl der erforderlichen Simulationsschritte schnell die Grenze des heute Machbaren erreicht, wenn die Energieskalen des Systems ungünstig sind. Deswegen findet ein Moleküldynamik-Verfahren nur selten Anwendung bei globalen Minimierungsproblemen. Allgemein verbrauchen die Verfahren des Simulated Annealing in Time zu viel Zeit, um die Potentialfläche auf ihrer feinsten Skala zu durchlaufen, und sind deshalb gegenüber anderen Verfahren im Nachteil, die die Möglichkeit bieten, das Potential auf verschiedenen Skalen zu betrachten. Diese Möglichkeit bietet zum Beispiel das Simulated Annealing in Temperature.

2.4 Simulated Annealing in Temperature

Bei den oben beschriebenen Verfahren des Simulated Annealing beschreibt man das System in seinem zeitlichen Verlauf. Durch die geeignete Simulation der physikalischen Bedingungen, wie die Vorgabe der Temperatur durch ein Temperaturbad oder durch die Benutzung der Wahrscheinlichkeiten für die Akzeptanz bei Monte-Carlo Schritten, erhält man ein System, das der gewünschten Statistik gehorcht und deshalb mit einer gewissen Wahrscheinlichkeit das globale Minimum findet.

Stattdessen kann man aber auch versuchen, das Maximum der Wahrscheinlichkeitsverteilung $p_B(\mathbf{x}, \beta)$ für den Grenzwert $\beta \rightarrow \infty$ direkt zu finden. Ausgangspunkt ist die Bloch-Gleichung, die die Temperaturentwicklung der Wahrscheinlichkeitsverteilung beschreibt:

$$\frac{\partial p}{\partial \beta} = -(H - \langle H \rangle) p \quad (2.40)$$

Dabei ist H die klassische Hamiltonfunktion. Für nicht explizit temperaturabhängige Potentiale ist die Lösung dieser Gleichung die schon bekannte Gibbs/Boltzmann-Verteilung

$$p_B(\mathbf{x}, \beta) = \frac{e^{-\beta H(\mathbf{x})}}{\int e^{-\beta H(\mathbf{x})} d\mathbf{x}} \quad (2.41)$$

Ziel ist es nun, das Maximum dieser Verteilung zu bestimmen. Dazu geht man von einer genäherten Verteilung \tilde{p} bei hoher Temperatur ($\beta \approx 0$) aus, wo die Näherung die Verteilung p_B gut beschreibt. Man versucht dann, durch die Beschreibung der Temperaturentwicklung der Parameter von p den Übergang zu niedrigen Temperaturen ($\beta \rightarrow \infty$) zu machen. Dabei spielt die Berechnung der thermodynamischen Mittelwerte eine große Rolle:

$$\langle f \rangle = \int f(\mathbf{x}) p(\mathbf{x}, \beta) d\mathbf{x} \quad (2.42)$$

Da das Maximum der Gibbs/Boltzmann-Verteilung auch das Minimum der potentiellen Energie darstellt, kann man die Kenntniss einer Näherung an die Wahrscheinlichkeitsverteilung bei niedrigen Temperaturen auf die Grundzustandskonfiguration des betrachteten Systems übertragen. Der Erfolg hängt entscheidend davon ab, wie gut die Näherung die Gibbs/Boltzmann-Verteilung beschreibt. Im Folgenden werden einige Verfahren vorgestellt, die auf diesem Prinzip beruhen.

2.4.1 Gaussian Density Annealing (GDA)

Eine mögliche Näherung an die Funktion p_B ist die, daß man annimmt, die Funktion ließe sich als Produkt von Einteilchen-Funktionen (Hartree) darstellen. Desweiteren nähert man die Einteilchen-Funktionen zu einfachen Gauß-Funktionen an, so daß man schreiben kann:

$$\begin{aligned} p_B(\mathbf{x}, \beta) &\approx p_{GDA}(\mathbf{x}, \beta) = \prod_{i=1}^N p_i(\vec{x}_i) = \prod_{i=1}^N (2\pi\sigma_i(\beta))^{-\frac{3}{2}} e^{-\frac{[\vec{x}_i - \vec{x}_i^0(\beta)]^2}{2\sigma_i(\beta)}} \\ &= \det^{-\frac{1}{2}}(2\pi\Lambda^2) e^{-\frac{1}{2}[\mathbf{x} - \mathbf{x}^0(\beta)]\Lambda^{-1}(\beta)[\mathbf{x} - \mathbf{x}^0(\beta)]} \end{aligned} \quad (2.43)$$

Dabei gilt

$$\mathbf{x} = \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_N \end{pmatrix} \quad \Lambda_{ij} = \delta_{ij}\sigma_i \quad (2.44)$$

Desweiteren wird das Zentrum der Gauß-Funktion als der thermodynamische Mittelwert der Position \mathbf{x} und die Einträge der Diagonal-Matrix Λ als Mittelwert der quadratischen Abweichung des Ortes vom Zentrum definiert.

$$\begin{aligned} \mathbf{x}^0 &= \langle \mathbf{x} \rangle_{GDA} \\ (\Lambda)_{ii} &= \langle (\vec{x}_i - \vec{x}_i^0)^2 \rangle_{GDA} \end{aligned}$$

Der thermodynamische Mittelwert einer Funktion berechnet sich dabei wie folgt

$$\langle f \rangle = \int f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x}$$

Das Subscript $\langle . \rangle_{GDA}$ bedeutet dabei, daß man im Integral die genäherte Funktion p_{GDA} verwendet.

Schon hier kann man erkennen, daß diese Näherung für große Temperaturen recht gut funktionieren wird, weil hier eine breite Funktion, die sich über den gesamten Konfigurationsraum erstreckt, sicherlich eine gute Annahme ist. Für kleine Temperaturen wird sie allerdings zu simpel sein, da sich die Aufenthaltswahrscheinlichkeit in voneinander durch hohe Energieberge getrennten Energietälern durch diese Funktion nicht richtig wiedergeben läßt. Dennoch kann man auf ein gutes Ergebnis der Berechnungen hoffen, wenn sich das Zentrum \mathbf{x}_0 der Gauß-Funktion immer im Tal des globalen Minimums befindet.

Um nun die Funktion $p_{GDA}(\mathbf{x}, \beta)$ für den Grenzwert $\beta \rightarrow \infty$ zu berechnen, geht man wie folgt vor:

GDA-Algorithmus

Schritt 1: Wähle bei hoher Temperatur $\beta \approx 0$ ein zufälliges \mathbf{x}^0 . Da bei hohen Temperaturen alle Konfigurationen beinahe gleichwahrscheinlich sind, wähle ein Λ , welches mindestens so groß ist, wie die Partikelabstände.

Schritt 2: Mit diesen gewählten \mathbf{x}^0 und Λ integriere für $\beta \rightarrow \infty$ folgende Gleichungen:

$$\frac{d}{d\beta} \mathbf{x}^0 = -\Lambda \nabla_{\mathbf{x}^0} \langle U \rangle_{GDA} \quad (2.45)$$

$$\frac{d}{d\beta} \Lambda = -\Lambda^2 \nabla_{\mathbf{x}^0}^2 \langle U \rangle_{GDA} \quad (2.46)$$

Mit

$$\langle U \rangle_{GDA} = \det^{-\frac{1}{2}}(2\pi\Lambda^2) \int U(\mathbf{x}) e^{-\frac{1}{2}[\mathbf{x}-\mathbf{x}^0(\beta)]\Lambda^{-1}(\beta)[\mathbf{x}-\mathbf{x}^0(\beta)]} d\mathbf{x} \quad (2.47)$$

Die Herleitung der Bewegungsgleichungen kann im Anhang (siehe A.1) nachgelesen werden. Die Form dieser Gleichungen ist äußerst interessant. Das Zentrum \mathbf{x}^0 bewegt sich gemäß eines steilsten Abstiegs auf der effektiven Potentialfläche, während die Breite Λ sich an die Krümmung der effektiven Potentialfläche anpaßt. Um als Ergebnis eine stabile Konfiguration zu erhalten, muß man die Gleichungen bis zu einem β_{max} integrieren, das die Bedingung

$$\frac{1}{\beta_{max}} \ll \Delta E$$

erfüllt, wobei ΔE wieder den Abstand der untersten beiden Energietäler darstellt.

Dieses Verfahren hat gegenüber dem traditionellen Simulated Annealing zwei entscheidende Vorteile. Zum einen werden die Gleichungen direkt in der Temperatur integriert, wodurch man keine zeitliche Dynamik zu beachten hat und die schwierige Wahl einer Kühlrate wegfällt. In den Temperaturgebieten, die zu einer Phasentransformation von gasförmig zu flüssig oder von flüssig zu fest gehören, wird die Anpassung von Λ automatisch sehr langsam sein, weil in solchen Gebieten die Krümmung von $\langle U \rangle_{GDA}$ relativ klein ist. Diese Tatsache spiegelt gerade den Unterschied zur Wahl einer "globalen" Kühlrate

wieder, die so klein gewählt werden muß, daß man genug Zeit bei den Temperaturen der Phasenübergänge verbringt. Der andere wichtige Vorteil ist, daß man durch die Benutzung einer Mittelung über die Wahrscheinlichkeitsverteilung mit einem geglätteten Potential rechnet. Diese Glättung bewirkt eine Reduktion der Anzahl von lokalen Minima und vereinfacht so die Suche nach dem globalen Minimum.

Ein eindimensionales Beispiel

Man kann sich die Wirkung der Bewegungsgleichungen 2.45 und 2.46 gut an einem eindimensionalen Beispiel verdeutlichen. Die Funktion $f : \mathfrak{R} \rightarrow \mathfrak{R}$ sei zum Beispiel wie folgt definiert:

$$f(x) = 2x^4 - 4x^2 - \frac{1}{2}x \quad (2.48)$$

Dann gilt für die mit der Wahrscheinlichkeitsverteilung $p_{GDA}(x_0, x, \sigma)$ gemittelte Funktion

$$\langle f \rangle_{GDA}(x_0, \sigma) = 6\sigma^2 + (12x_0^2 - 4)\sigma + f(x_0) \quad (2.49)$$

Um nun veranschaulichen zu können, auf welcher Funktion ein Teilchen, beschrieben durch die Bewegungsgleichungen, sich bewegt, muß man das oben angegebene Verfahren leicht abwandeln. Man integriert Gl.2.46 für jedes x_0 solange, bis sie konvergiert. Dann gilt $\frac{d}{d\beta}\sigma = 0$ und man kann man das Resultat für das eindimensionale Beispiel leicht ausrechnen:

$$\sigma(x_0) = \begin{cases} \frac{1}{3} - x_0^2 & \text{für } x_0^2 \leq \frac{1}{3} \\ 0 & \text{sonst} \end{cases} \quad (2.50)$$

Dabei wurde beachtet, daß die Funktion $\langle f \rangle_{GDA}$ nur für positive Werte von σ definiert ist. Setzt man 2.50 in $\langle f \rangle_{GDA}(x_0, \sigma(x_0))$ ein, so erhält man das in Abbildung 2.8 gezeigte Ergebnis. Man sieht, daß sich das System auf einer Potentialfläche bewegt, bei der die Energiebarrieren weggeglättet sind, so daß sie leichter überwunden werden können. Befände sich in Abbildung 2.8 ein Teilchen in der Nähe von $x = 0$, so würden die Bewegungsgleichungen in diesem Bereich wegen der schwachen Krümmung der Funktion nur

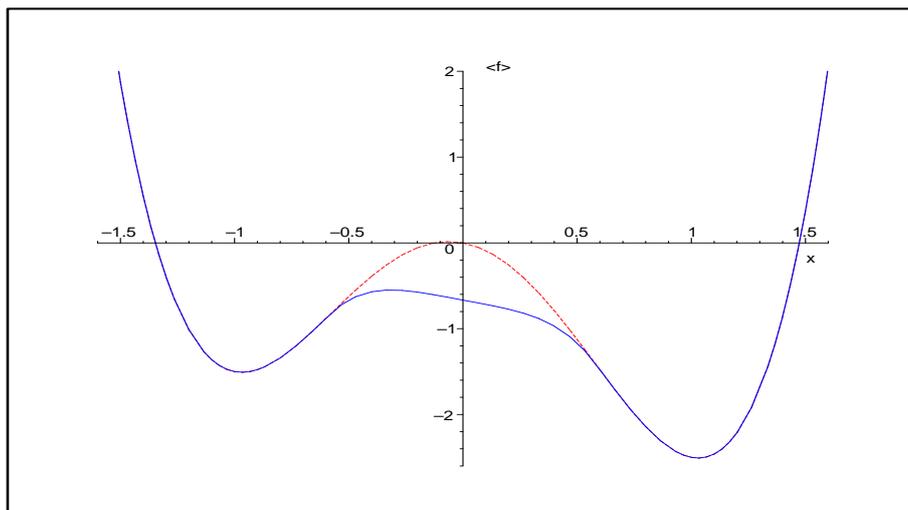


ABBILDUNG 2.8: Gemittelte Funktion $\langle f \rangle(x, \sigma(x))$ (—) und Funktion $f(x)$ (- - -)

sehr kleine Änderungen der Breite σ zulassen. Diese Eigenschaft ist sehr wichtig für die Berechnung einer realistischen Wahrscheinlichkeitsverteilung, da in den entscheidenden Gebieten eine hohe Genauigkeit gewährleistet ist.

2.4.2 adiabatic Gaussian Density Annealing (aGDA)

Wie oben schon erwähnt ist die Näherung an die Wahrscheinlichkeitsverteilung p_B für kleine Temperaturen nicht sehr gut. Weil die Basis für p_B unvollständig ist, kann es bei der Berechnung zu Fehlern kommen, die sich im Laufe der Integration noch verstärken. Tsou und Brooks (siehe [1]) haben festgestellt, daß diese Fehler die Verteilung p_{GDA} von der wirklichen Verteilung p_B wegführen können und so das Ergebnis verschlechtern. Sie haben vorgeschlagen, das Zentrum der Gauß-Funktion \mathbf{x}^0 nicht gemäß Gleichung 2.45 zu bewegen, sondern in jedem Schritt das \mathbf{x}^0 an das lokale Minimum anzupassen. So erhält man folgenden Algorithmus:

aGDA-Algorithmus

Schritt 1: Setze $k = 0$, wähle eine zufällige Startkonfiguration \mathbf{x}_0^0 und Breite Λ_0 größer als die Partikelabstände.

Schritt 2: Löse ausgehen von \mathbf{x}_k^0

$$\nabla_{\mathbf{x}^0} \langle U \rangle_{GDA} (\mathbf{x}^0, \Lambda_k) |_{\mathbf{x}^0 = \mathbf{x}_{k+1}^0} = 0 \quad (2.51)$$

und setze dann

$$\Lambda_{k+1} = \Lambda_k - \Delta\beta \cdot \Lambda^2 \nabla_{\mathbf{x}^0}^2 \langle U \rangle_{GDA} \quad (2.52)$$

Schritt 3: Setze $k = k + 1$, ist $|\Lambda_k| < \lambda_{max}$ breche ab, sonst zurück zu **Schritt 2**.

$\Delta\beta$ wird so klein gewählt, daß die Änderungen von Λ in jedem Schritt nicht zu groß sind. Das Abbruchkriterium für den Algorithmus ist hier anders gewählt worden, als beim ursprünglichen GDA-Verfahren, weil es einfacher zu handhaben ist. Man berechnet einfach die (euklidische oder Maximums-) Norm der Matrix Λ_k und überprüft, ob sie einen gewissen Wert unterschreitet. Setzt man zum Beispiel $\lambda_{max} = 1.0 \cdot 10^{-4}$ und wählt die Maximumsnorm, so kann man bei Abbruch des Algorithmus sicher sein, daß die Temperatur klein genug ist. Die Konfiguration wird sich dann nicht mehr ändern. Man führt dann wie immer eine lokale Minimierung auf dem ungeglätteten Potential aus, um das endgültige Ergebnis zu erhalten. Diese Variation des GDA-Algorithmus ist erfolgreicher bei der Optimierung von Lennard-Jones-Clustern und Wasser-Molekülen, so daß sie auch in den späteren Kapiteln Anwendung finden wird.

2.4.3 Quantum-Mechanical Annealing (QMA)

Das Quantum-Mechanical Annealing beschreibt eigentlich keine Entwicklung der statistischen Wahrscheinlichkeitsverteilung in der Temperatur, sondern bestimmt die Grundzustandswellenfunktion als Lösung der Schrödingergleichung. Trotzdem hat es sehr große Ähnlichkeit mit den oben beschriebenen Annealing Verfahren, so daß es zum Simulated

Annealing in Temperature dazugezählt wird.

Hinter dem Begriff des Quantum-Mechanical Annealing verbirgt sich eigentlich eine Vielzahl von unterschiedlichen Verfahren, die Grundzustands-Wellenfunktion eines Systems zu bestimmen. Ausgangspunkt für alle Verfahren ist die Schrödinger-Gleichung

$$i\hbar \frac{\partial}{\partial t} \Psi(\mathbf{x}, t) = \left[-\frac{\hbar^2}{2m} \nabla^2 + U(\mathbf{x}) \right] \Psi(\mathbf{x}, t) \quad (2.53)$$

Oder in ihrer zeitunabhängigen Form

$$\left[-\frac{\hbar^2}{2m} \nabla^2 + U(\mathbf{x}) \right] \Phi(\mathbf{x}) = E \Phi(\mathbf{x}) \quad (2.54)$$

mit $\Psi(\mathbf{x}, t) = \Phi(\mathbf{x}) e^{-\frac{i}{\hbar} E t}$.

Bei der sogenannten „time-independent Variational Method“ geht man davon aus, daß der Zustand mit jeder berechneten Wellenfunktion $\tilde{\Phi}$ größere Energie als der Grundzustand hat:

$$\tilde{E} = \langle \tilde{\Phi} | H | \tilde{\Phi} \rangle \geq E_0 \quad (2.55)$$

Da die Hamiltonfunktion H vom Potential abhängt, welches wiederum die räumliche Konfiguration des Systems als Parameter enthält, löst man iterativ jeweils die Schrödinger-Gleichung mit konstantem Potential und berechnet mit der Wellenfunktion das Potential neu. Das muß man so lange durchführen, bis sich Potential und Wellenfunktion nicht mehr verändern, also selbstkonsistent sind. Das tut man für verschiedene Anfangskonfigurationen, so daß die Wellenfunktion mit der niedrigsten Energie als dem Grundzustand am nächsten angesehen wird.

Die „Imaginary Time Methods“ führen eine imaginäre Zeit $\tau = \frac{it}{\hbar}$ ein, so daß sich die Schrödinger-Gleichung wie folgt schreibt:

$$\frac{\partial}{\partial \tau} \Psi(\mathbf{x}, \tau) = \left[\frac{\hbar^2}{2m} \nabla^2 - U(\mathbf{x}) \right] \Psi(\mathbf{x}, \tau) \quad (2.56)$$

Dann läßt sich die Zeitabhängigkeit der Wellenfunktion als

$$\Psi(\mathbf{x}, \tau) = e^{-E\tau} \Phi(\mathbf{x}) \quad (2.57)$$

schreiben. Drückt man eine Wellenfunktion durch die Summe über eine Basis $u_n(\mathbf{x})$ mit Eigenwerten E_n aus,

$$\Psi(\mathbf{x}, \tau) = \sum_n c_n u_n(\mathbf{x}) e^{-E_n \tau} \quad (2.58)$$

dann sieht man, daß für große Werte von τ nur noch der Grundzustand n_0 merklich zur Summe beiträgt:

$$\Psi(\mathbf{x}, \tau) \approx u_0(\mathbf{x}) e^{-E_0 \tau} \quad (2.59)$$

Die Verfahren nutzen dieses Verhalten aus, indem sie, von einer zufälligen Wellenfunktion ausgehend, Gl.2.56 integrieren. Bei großem τ stellt die resultierende Wellenfunktion eine Näherung an den Grundzustand dar.

Eine Möglichkeit, die Lösung von Gl.2.56 zu finden, ist nach Straub [3] das Quantum-Mechanical Annealing unter Benutzung von Gaussian Wave-Packets. Dieses Verfahren hat

sehr große Ähnlichkeit mit dem Gaussian-Density Annealing. Man stellt die Wellenfunktion durch ein Produkt aus Einteilchen-Gauß Funktionen dar:

$$\Psi(\mathbf{x}, \tau) = \det^{-\frac{1}{4}}(2\pi\Lambda) e^{-\frac{1}{4}(\mathbf{x}-\mathbf{x}^0)\Lambda^{-1}(\mathbf{x}-\mathbf{x}^0)} \quad (2.60)$$

Dabei gelten die gleichen Beziehungen für $(\Lambda)_{ij} = \delta_{ij}\sigma_i$ und \mathbf{x}^0 wie in Kap.2.4.1. Also

$$\mathbf{x}^0 = \langle \mathbf{x} \rangle \quad (2.61)$$

$$\sigma_i = \langle (\vec{x}_i - \vec{x}_i^0)^2 \rangle \quad (2.62)$$

wobei sich hier die quantenmechanischen Erwartungswerte eines Operators A nach

$$\langle A \rangle = \int \Psi^* A \Psi d\mathbf{x} \quad (2.63)$$

berechnen. Die (imaginäre) zeitliche Entwicklung eines solchen Erwartungswertes $\langle A \rangle$ erhält man dann im Allgemeinen wie folgt¹

$$\frac{d\langle A \rangle}{d\tau} = -\langle HA + AH \rangle + 2\langle A \rangle \langle H \rangle \quad (2.64)$$

Setzt man in Gl.2.64 die Erwartungswerte \mathbf{x}^0 und σ_i ein, so kann man die Zeitentwicklung durch folgende zwei Gleichungen ausdrücken:

$$\frac{d\mathbf{x}^0}{d\tau} = -2\Lambda \nabla_{\mathbf{x}^0} \langle U \rangle \quad (2.65)$$

$$\frac{d\Lambda}{d\tau} = \frac{\hbar^2}{2m} - 2\Lambda^2 \nabla_{\mathbf{x}^0}^2 \langle U \rangle \quad (2.66)$$

Es fällt sofort die große Ähnlichkeit mit den Gleichungen 2.45 und 2.46 des GDA auf. Einzig der Term $\frac{\hbar^2}{2m}$ unterscheidet hier die quantenmechanische Berechnung von der klassischen. Deshalb ist auch der Algorithmus kaum vom Gaussian-Density Annealing zu unterscheiden.

QMA-Algorithmus

Schritt 1: Setze $k = 0$, wähle zufälliges \mathbf{x}_0^0 und ein Λ_0 größer als die Partikelabstände, setze $\hbar_0 = \hbar_{max}$.

Schritt 2: Löse ausgehend von \mathbf{x}_k^0 und Λ_k Gl.2.65 und Gl.2.66 mit $\hbar_k = const$. Setze $k = k + 1$, $\hbar_k = \hbar_{k-1} - \Delta\hbar$.

Schritt 3: Falls $\hbar_k \approx 0$ gilt, stoppe, sonst zurück zu **Schritt 2**.

Trotz der großen Ähnlichkeit zum GDA-Algorithmus gibt es hier doch auch wesentliche Unterschiede. Zum einen werden hier für jedes \hbar_k die beiden Bewegungsgleichungen für das Zentrum und die Breite der Gauß-Verteilung gelöst und zum anderen bewirkt der zusätzliche Faktor $\frac{\hbar^2}{2m}$ eine Verbreiterung der Verteilung, was die quantenmechanische Natur der Partikel deutlich macht. Das Wellenpaket wird durch die Bewegungsgleichungen so

¹Für die Herleitung der Bewegungsgleichungen siehe auch A.2

angepaßt, daß der Erwartungswert der Gesamtenergie minimiert wird. Dann verkleinert man \hbar um einen konstanten Faktor und erhält im klassischen Grenzfall $\lim_{\hbar \rightarrow 0}$ die Lösung. Beim Gaussian-Density Annealing beschränkt man sich ausschließlich auf den klassischen Fall und führt demnach nur den letzten Schritt des QMA-Algorithmus aus. Dabei ist allerdings zu beachten, daß es nicht darum geht, ob das betrachtete System im physikalischen Sinn klassisch oder quantenmechanisch behandelt werden soll. Vielmehr will man durch das Quantum-Mechanical Annealing die Tatsache ausnutzen, daß Teilchen Energiebarrieren durchtunneln können auch wenn sie nicht genügend kinetische Energie haben, um die Barriere zu überwinden, wobei sich die zusätzliche Möglichkeit des Durchtunnelns nur in der Breite der Verteilung widerspiegelt. Man darf jedoch nicht vergessen, daß im klassischen Fall ein einzelnes System zwar keine Energiebarrieren überwinden kann, wenn die eigene Energie nicht ausreicht, aber ein Ensemble von Systemen ist dazu durchaus in der Lage, obwohl seine mittlere Energie eventuell geringer ist als die Höhe der Barriere. Betrachtet man nämlich Gleichungen 2.65 und 2.66, so sieht man, daß sich \mathbf{x}^0 auch für $\hbar = 0$ (oder auch beim GDA-Algorithmus) nicht auf der ursprünglichen Potentialfläche U , sondern stets auf einer über die Verteilung gemittelten Potentialfläche $\langle U \rangle$ bewegt. Die Gleichungen minimieren jeweils $\langle U \rangle$ in \mathbf{x}^0 und Λ , so daß Maxima von U immer geglättet werden, Minima jedoch nicht. Anders ausgedrückt, führt Gl.2.66 zu einer Verbreiterung des Wellenpaketes in Gebieten, wo $\langle U \rangle$ konkav, also $\nabla_{\mathbf{x}^0}^2 \langle U \rangle < 0$ ist, und tendiert in konvexen Gebieten im Gegensatz dazu, die Breite zu verringern, so daß dort für $\hbar = 0$ $\langle U \rangle \approx U$ gilt (siehe auch Abbildung 2.8). Der Effekt des Tunnelns tritt also auch schon beim klassischen Gaussian-Density Annealing auf, wird beim QMA nur noch durch einen zusätzlichen Faktor verstärkt.

2.4.4 Packet Annealing Method (PAM)

Beim Gaussian-Density-Annealing wird die Gibbs/Boltzmann-Verteilung als ein Produkt aus Einteilchen-Gauß-Funktionen angenähert. Diese Näherung hat zwei offensichtliche Nachteile. Zum einen sind die Einteilchen-Verteilungen völlig unabhängig voneinander und zum anderen konzentriert sich die Verteilung ausschließlich um einen Punkt im Konfigurationsraum. Diese Nachteile versucht die Packet Annealing Method von Shalloway [5] zu beheben.

Die Wahrscheinlichkeitsverteilung p_B kann in viele voneinander getrennte Gebiete mit großen Werten aufgeteilt sein. Das ist einfach eine Folge der Tatsache, daß auch die potentielle Energie tiefe Minima in ganz verschiedenen Gebieten des Konfigurationsraums haben kann. Um dieser Eigenschaft gerecht zu werden, macht man nun die folgende Näherung an p_B

$$p_B^{\frac{1}{2}}(\mathbf{x}) \approx p_{PAM}^{\frac{1}{2}}(\mathbf{x}) = \sum_{\{\alpha\}} \Phi_{\alpha}^0(V_{\alpha}^0, \mathbf{x}_{\alpha}^0, \Lambda_{\alpha}^0, \mathbf{x}) \quad (2.67)$$

Die Φ_{α}^0 sind dabei die sogenannten charakteristischen Packets der Form

$$\Phi_{\alpha}^0 = e^{-\frac{1}{2}[\beta V_{\alpha}^0 + \frac{1}{2}(\mathbf{x} - \mathbf{x}_{\alpha}^0)\Lambda_{\alpha}^{-2}(\mathbf{x} - \mathbf{x}_{\alpha}^0)]} \quad (2.68)$$

Jedes Packet stellt eine Gauß-Funktion um den Punkt \mathbf{x}_{α}^0 dar, die mit dem Faktor $e^{-\frac{1}{2}\beta V_{\alpha}^0}$ mit $V_{\alpha}^0 = U(\mathbf{x}_{\alpha}^0)$ multipliziert wird. Im Gegensatz zur GDA können hier die Λ_{α}^0 anisotrop, d.h. symmetrische, semi-definite Matrizen sein. Diese Verallgemeinerung bewirkt, daß die

Achsen der Schwankung in beliebigen Richtungen im Konformationsraum liegen dürfen. Das kann zum Beispiel wichtig sein, wenn bei großen Molekülen korrelierte Bewegungen von ganzen Atomgruppen nötig sein sollten.

Die einzelnen Packets sind in Gebieten lokalisiert, die durch hohe Energiebarrieren voneinander getrennt sind. Man kann daher jedes Packet mit einem möglichen Makrozustand (siehe Gl. 2.32) identifizieren, wenn für das Zentrum und die mittlere quadratische Abweichung eines solchen Packets die Bedingung erfüllt ist, daß sie in erster Ordnung nicht abhängig von der Zeit sind. Also

$$\frac{\partial}{\partial t} \langle \mathbf{x} \rangle_{\alpha}^{PAM} = 0 \quad \frac{\partial}{\partial t} \langle (\mathbf{x} - \mathbf{x}_{\alpha}^0)(\mathbf{x} - \mathbf{x}_{\alpha}^0) \rangle_{\alpha}^{PAM} = 0 \quad (2.69)$$

Die Mittelwerte werden jetzt wie folgt berechnet

$$\langle f \rangle_{\alpha}^{PAM} = \frac{\int f \cdot p_{PAM}^{\frac{1}{2}} \Phi_{\alpha}^0 d\mathbf{x}}{\int p_{PAM}^{\frac{1}{2}} \Phi_{\alpha}^0 d\mathbf{x}} \quad (2.70)$$

Ähnlich wie beim GDA setzt man wieder

$$\mathbf{x}_{\alpha}^0 = \langle \mathbf{x} \rangle_{\alpha}^{PAM} \quad \Lambda_{\alpha}^2 = \langle (\mathbf{x} - \mathbf{x}_{\alpha}^0)(\mathbf{x} - \mathbf{x}_{\alpha}^0) \rangle_{\alpha}^{PAM} \quad (2.71)$$

Die Gleichungen 2.71 stellen die Integralform der „Packet Equations“ dar. Ihre Lösungen bestimmen die Packets. Um mit diesen Gleichungen das globale Energieminimum zu finden, löst man sie zuerst bei einer hohen Temperatur, wo nur wenige Lösungen existieren, und verfolgt dann die Packets bis zu einer Temperatur nahe 0. Eine wichtige Rolle dabei spielen dabei die sogenannten „branching transitions“, bei denen sich ein Packet teilt. Das geschieht, wenn die Temperatur gesenkt wird, so daß eine vorher zu niedrige Energiebarriere das Gebiet des Packets teilt und die Wahrscheinlichkeitsverteilung dort in zwei Gebiete getrennt wird. Die Verfolgung solcher neuer Packets ist sehr wichtig, da man das Gebiet des globalen Minimums nur dann mit großer Wahrscheinlichkeit lokalisieren kann, wenn man möglichst alle gefundenen Packets bis zur tiefsten Temperatur verfolgt. Aber aus Gründen der Effizienz des Algorithmus ist das nicht möglich, so daß man sich Kriterien überlegen muß, nach denen man die Packets auswählt, die dicht mehr weiterverfolgt werden sollen.

Interessant ist die Betrachtung der Differentialform der „Packet Equations“

$$\left. \frac{\partial H_{\Lambda_{\alpha}, T}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_{\alpha}^0} = 0 \quad \left. \frac{\partial^2 H_{\Lambda_{\alpha}, T}(\mathbf{x})}{\partial \mathbf{x}^2} \right|_{\mathbf{x}=\mathbf{x}_{\alpha}^0} = \frac{\Lambda_{\alpha}^{-2}}{2\beta} \quad (2.72)$$

mit der effektiven Energie,

$$H_{\Lambda_{\alpha}, T}(\mathbf{x}) = -2\beta^{-1} \ln \left[\det^{-\frac{1}{4}}(2\pi\Lambda^2) \cdot \int e^{-\frac{1}{2}\beta U(\mathbf{x}')} e^{-\frac{1}{4}(\mathbf{x}-\mathbf{x}')\Lambda^{-2}(\mathbf{x}-\mathbf{x}')} d\mathbf{x}' \right] \quad (2.73)$$

die die freie Energie des Packets Φ_{α}^0 annähert. Die Differentialform der Gleichungen machen auch den Zusammenhang der PAM zum GDA und der DEM deutlich. Darauf wird noch in Abschnitt 2.5 näher eingegangen.

Der PAM-Algorithmus sieht insgesamt dann wie folgt aus:

PAM-Algorithmus

Schritt 1: Suche bei einer hohen Temperatur T_{hi} , die Lösungen der Packet Equations 2.71 $\mathbf{x}_\alpha^0(T_{hi})$ und $\Lambda_\alpha(T_{hi})$.

Schritt 2: Setze $T \rightarrow T - \Delta T$.

Aktualisiere die $\mathbf{x}_\alpha^0(T)$ und $\Lambda_\alpha(T)$ für jedes Packet.

Teste, welche Packets sich teilen, und entscheide, welche weiterverfolgt werden.

Schritt 3: Falls $T \approx 0$ stoppe, ansonsten wiederhole **Schritt 2**.

Am Ende des Algorithmus ist das Packet mit der geringsten effektiven Energie $H_{\Lambda_\alpha, T}(\mathbf{x}_\alpha^0)$ mit der größten Wahrscheinlichkeit in der Umgebung des globalen Minimums der freien Energie lokalisiert. Das Verfahren bietet einige enorme Vorteile. Zum einen wird mit einer Näherung an die freie Energie gerechnet, wodurch das physikalische Verhalten besser wiedergegeben wird. Zum anderen ist die Näherung der Wahrscheinlichkeitsverteilung wesentlich besser als die einfache Einteilchen-Näherung, so daß die Gibbs/Boltzmann-Verteilung in jedem Temperaturbereich gut wiedergegeben werden kann. Der Nachteil des Verfahrens ist jedoch, daß es schon bei der Anwendung auf einfache Potentiale große Schwierigkeiten bereitet. Probleme dabei sind vor allem die Berechnung der effektiven Energie $H_{\Lambda_\alpha, T}(\mathbf{x}_\alpha^0)$ und die Feststellung der Teilungen eines Packets. Deshalb ist es in seiner vollständigen Form bis heute nicht implementiert worden und bietet noch wenig praktischen Nutzen.

2.4.5 Minimierung mittels des Effective Diffused Potential (EDP)

Die stabilste Konfiguration von physikalische Systemen in einem Temperaturbad wird durch die freie Energie

$$F = E - TS \quad (2.74)$$

bestimmt. Dabei spielt nicht nur die potentielle (innere) Energie eine Rolle, sondern ebenso die Entropie S . Das System versucht die Energie zu minimieren und die Entropie zu maximieren. Das führt dazu, daß bei niedriger Temperatur ein enges globales Minimum der potentiellen Energie gegenüber einem höher liegenden, breiteren Minimum mit größerer Entropie thermodynamisch benachteiligt sein kann. Bei der Minimierung des Potentials erhält man also nicht unbedingt die physikalisch wahrscheinlichste Konfiguration. Kennt man jedoch das Minimum der freien Energie bei einer Temperatur T , so hat man ein physikalisch sinnvolles Ergebnis. Verschelde und Schelstraete [13] schlagen vor, sich das sogenannte Gibb'sche Prinzip zu nutze machen, um die freie Energie zu einer gegebenen Temperatur zu berechnen. Dazu betrachtet man folgende Funktion:

$$\Psi(p) = \int H p \, dv \, dq + \beta^{-1} \int p \ln(p) \, dv \, dq \quad (2.75)$$

Dabei ist H die klassische Hamiltonfunktion, p die Wahrscheinlichkeitsdichte und $\int \, dv \, dq$ meint die Integration über den gesamten Phasenraum. Das Gibb'sche Prinzip besagt nun, daß das Minimum dieser Funktion Ψ die freie Energie ist. Es gilt also

$$\Psi(\bar{p}) = F \quad (2.76)$$

falls \bar{p} die obige Funktion minimiert, und es ist

$$\bar{p} = \frac{e^{-\beta H}}{\int e^{-\beta H} dv dq} \quad (2.77)$$

Um das zu sehen, schaut man sich Variationen von Ψ in Abhängigkeit von einer kleinen Änderung δp an:

$$\delta\Psi = \int \{ [H + \beta^{-1}(1 + \ln(p))] \delta p \} dv dq \quad (2.78)$$

$$\delta^2\Psi = \int (\beta p)^{-1} dv dq \quad (2.79)$$

Da p eine positive Funktion und damit $\delta^2\Psi$ ebenfalls positiv ist, ist Ψ konvex und die Stelle $\delta\Psi = 0$ liefert ein Minimum. Berücksichtigt man nun noch die Normalisierung $\int p dv dq = 1$ durch einen Lagrange-Multiplikator, so erhält man folgende Bedingung

$$0 = \delta\Psi + \lambda \cdot \delta \left(\int p dv dq \right) = \int \{ [H + \beta^{-1}(1 + \ln(p)) + \lambda] \delta p \} dv dq \quad (2.80)$$

Die Lösung \bar{p} dieser Gleichung ist in der Tat die Gibbs/Boltzmann- Wahrscheinlichkeitsfunktion aus 2.77 und es gilt

$$\Psi(\bar{p}) = -\beta^{-1} \ln \left(\int e^{-\beta H} dv dq \right) = F \quad (2.81)$$

Etwas vereinfacht kann man also schreiben

$$F = \underset{\{p\}}{\text{Min}} (E(p) - kTS(p)) \quad (2.82)$$

$$= \underset{\{p\}}{\text{Min}} \left(\int p(\mathbf{x})U(\mathbf{x})d\mathbf{x} + kT \int p(\mathbf{x}) \ln(p(\mathbf{x}))d\mathbf{x} \right) \quad (2.83)$$

Man könnte nun die freie Energie berechnen, indem man die Verteilung aus 2.77 in die rechte Seite von 2.83 einsetzt. Aber mit der Gibbs/Boltzmann-Verteilung lassen sich die Integrale nicht analytisch ausführen. Will man daher die Verteilung durch einen geeigneten Ansatz annähern, so kann man die rechte Seite mit diesem Ansatz minimieren. Man beschränkt sich dadurch allerdings auf einen bestimmten Unterraum aller möglichen Wahrscheinlichkeitsverteilungen, so daß man nur eine obere Grenze für die freie Energie bestimmen kann. Genau wie beim Gaussian-Density-Annealing ist ein Einteilchenansatz mit Gauß-Funktionen wieder eine einfache Möglichkeit:

$$p(\mathbf{x}, \bar{\mathbf{x}}, \Lambda) = \det^{-\frac{1}{2}}(\pi\Lambda) e^{-(\mathbf{x}-\bar{\mathbf{x}})\Lambda^{-1}(\mathbf{x}-\bar{\mathbf{x}})} \quad (2.84)$$

mit

$$(\Lambda)_{ij} = \delta_{ij}\sigma_i \quad (2.85)$$

wie in Kapitel 2.4.1. Mit diesem Ansatz reduziert sich das Problem aus Gl.2.83 auf eine Minimierung in $4N$ Variablen, wenn N die Anzahl der Atome im System ist. Die explizite Form der freien Energie sieht mit diesem Ansatz wie folgt aus:

$$F(\bar{\mathbf{x}}, \Lambda, T) = \det^{-\frac{1}{2}}(\pi\Lambda) \int U(\mathbf{x})e^{-(\mathbf{x}-\bar{\mathbf{x}})\Lambda^{-1}(\mathbf{x}-\bar{\mathbf{x}})} d\mathbf{x} - \frac{kT}{2} \left[3 \sum_{i=1}^N \ln(\pi\sigma_i) + 3N \right] \quad (2.86)$$

Die rechte Seite kann man wieder in zwei Teile zerlegen,

$$F(\bar{\mathbf{x}}, \Lambda, T) = U_{eff}(\bar{\mathbf{x}}, \Lambda, T) - kTS_{eff}(\bar{\mathbf{x}}, \Lambda, T) \quad (2.87)$$

so daß man die Beziehung zur Definition der freien Energie 2.74 sehen kann. Die beiden Terme werden als die effektive interne Energie und die effektive Entropie dieser Näherung interpretiert. Die Parameter $\bar{\mathbf{x}}$ und σ_i erhält man nun durch Lösen der folgenden Gleichungen, die die notwendige Bedingung für ein Minimum von F darstellen:

$$\frac{\partial F(\bar{\mathbf{x}}, \Lambda, T)}{\partial \bar{\mathbf{x}}} = 0 \quad (2.88)$$

$$\frac{\partial F(\bar{\mathbf{x}}, \Lambda, T)}{\partial \sigma_i} = 0 \quad i = 1, \dots, N \quad (2.89)$$

Das globale Minimum von $F(\bar{\mathbf{x}}, \Lambda, T)$ ist die freie Energie dieser Approximation bei der gegebenen Temperatur. Andere Minima können als metastabile Zustände interpretiert werden, die mit steigender Temperatur verschwinden. Denn wählt man die Temperatur nur groß genug, so hat $F(\bar{\mathbf{x}}, \Lambda, T)$ nur ein einziges Minimum. Die Existenz weiterer Minima ist außerdem nur eine Folge der Näherung der Wahrscheinlichkeitsverteilung durch eine unvollständige Basis. Das wird weiter unten noch näher erläutert werden.

Der Algorithmus sieht dann wie folgt aus:

EDP-Algorithmus

Schritt 1: Setze $k = 0$ und $T_0 = T_{max}$. Wähle eine beliebige Startkonfiguration $\bar{\mathbf{x}}_0$ und Λ_0 .

Schritt 2: Setze $k = k + 1$ und $T_k = T_{k-1} - \Delta T$. Berechne die Lösungen \mathbf{x}_k und Λ_k der Gl.2.88 und 2.89 ausgehend von \mathbf{x}_{k-1} und Λ_{k-1} .

Schritt 3: Falls $T_k < T_{min}$ stoppe, sonst gehe zu **Schritt 2**.

Der Algorithmus berechnet also zunächst die freie Energie bei einer hohen Temperatur, um die Verteilung p dann durch Abkühlen möglichst gut der Gibbs/Boltzmann Verteilung bei der Temperatur $T = 0$ anzunähern.

Ähnlich wie bei den Gaussian-Density-Annealing Verfahren, hängt die Güte des Algorithmus in erster Linie von der Näherung der Wahrscheinlichkeitsverteilung durch eine Gauß-Funktion ab. Diese Näherung ist mit Sicherheit unzureichend, da die Annahme, die Verteilung sei bei endlichen Temperaturen nur um einen Punkt im Konfigurationsraum konzentriert, nicht korrekt ist. Sie ist aber nötig, um das Verfahren praktikabel zu machen. Am Beispiel einer eindimensionalen Funktion kann man verschiedene Eigenschaften des EDP deutlich machen.

Ein eindimensionales Beispiel

Es ist sinnvoll, sich die Eigenschaften des Effective Diffused Potentials an einem eindimensionalen Beispiel zu verdeutlichen. Dem Beispiel aus [14] folgend, kann man die einfache Funktion

$$U(x) = 2x^4 - 4x^2 - \frac{1}{2}x \quad (2.90)$$

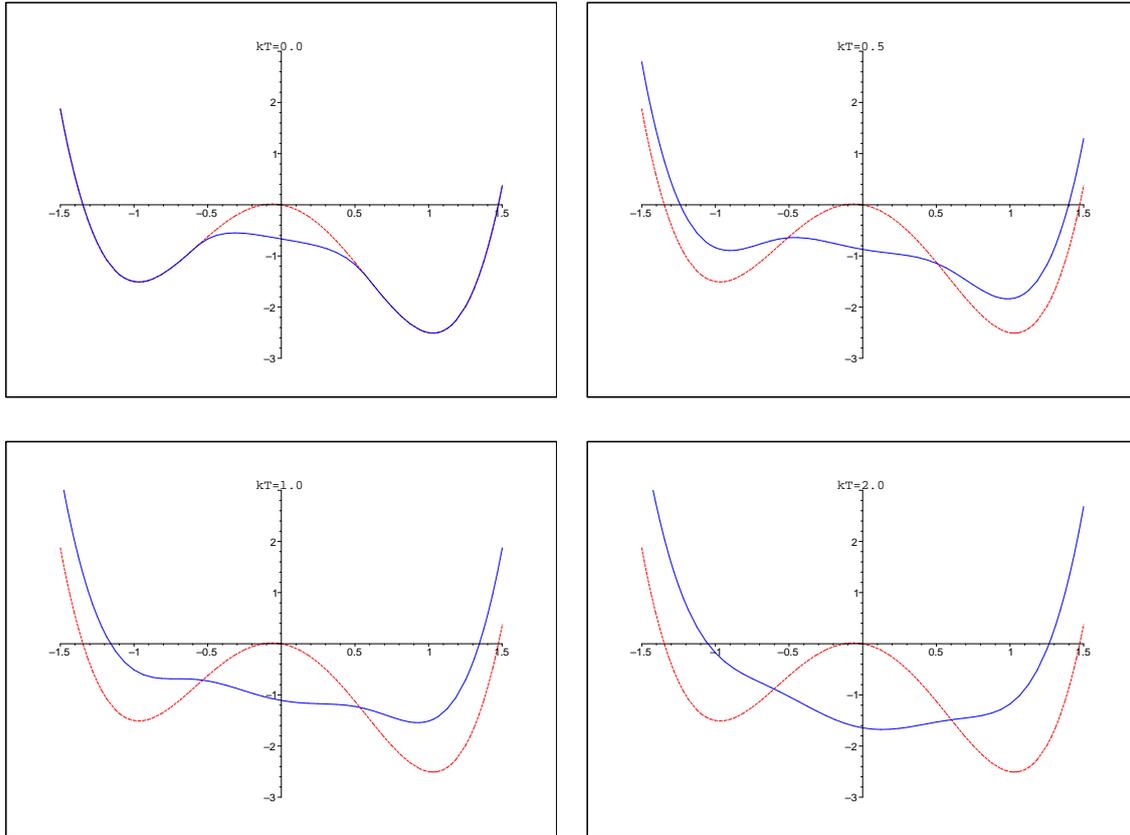


ABBILDUNG 2.9: Die effektive freie Energie F^{eff} (—) und das Potential U (- - -) bei verschiedenen Temperaturen

betrachten. Eingesetzt in Gl.2.86 ergibt sich für diese Potential

$$F(\bar{x}, \sigma, T) = -\frac{kT}{2}(\ln(\pi\sigma) + 1) + 1.5\sigma^2 + \sigma(6\bar{x}^2 - 2) + U(\bar{x}) \quad (2.91)$$

Die Bedingung 2.89 läßt sich damit leicht ausrechnen:

$$\frac{\partial F(\bar{x}, \sigma, T)}{\partial \sigma} = -\frac{kT}{2\sigma} + 3\sigma + (6\bar{x}^2 - 2) \quad (2.92)$$

$$\Rightarrow \sigma(\bar{x}, T) = \frac{1}{6} \left[\sqrt{(6\bar{x}^2 - 2)^2 + 6kT} - (6\bar{x}^2 - 2) \right] \quad (2.93)$$

Setzt man das wiederum in Gl.2.91 ein, so erhält man die effektive freie Energie zu jeder Temperatur als Funktion des Ortes $F^{eff}(\bar{x}, T)$. In Abbildung 2.9 sieht man die effektive freie Energie und das Potential für verschiedene Temperaturen. Je höher die Temperatur ist, desto stärker ist der Effekt der Glättung. Vor allem interessant ist aber, daß auch bei $T = 0$ eine Glättung des Maximums eintritt. Die effektive freie Energie stimmt in den Gebieten der Minima mit dem Potential überein, während sie im Bereich des Maximums nicht gegen das Potential konvergiert. Warum das so ist, wird deutlich, wenn man sich den Verlauf von σ für $T = 0$ ansieht.

$$\sigma(\bar{x}, T = 0) = \frac{1}{6} \left[|(6\bar{x}^2 - 2)| - (6\bar{x}^2 - 2) \right] \quad (2.94)$$

Die rechte Seite verschwindet offensichtlich nur für $(6x^2 - 2) \geq 0$ bzw. $\frac{\partial^2 U}{\partial \bar{x}^2} \geq 0$. Das bedeutet, es wird zwischen konvexen und konkaven Gebieten der Potentialfunktion unterschieden. Die konkaven Gebiete werden auch bei $T = 0$ geglättet, was zunächst willkürlich aussieht. Aber hier spiegelt sich eine schon anfangs erwähnte Tatsache der exakten freien Energie wieder. Könnte man F nämlich exakt berechnen, so wäre sie überall konvex in dem Sinne, daß jede Abweichung der Verteilung p von der „richtigen“ Verteilung \bar{p} ein höheres F zur Folge hat. Da man p aber durch die einfache Parametrisierung auf einen kleinen Unterraum aller möglichen Wahrscheinlichkeitsverteilungen beschränkt, verliert man diese Eigenschaft. Trotzdem behält F^{eff} zumindest die Tendenz über, die Krümmung in konkaven Gebieten des Potentials zu verringern und die Funktion so glatter zu machen.

2.5 Zusammenfassung

Alle Verfahren des Simulated Annealing gehen von einer Näherung aus Einteilchen-Gauß-Funktionen an die benutzten Verteilungen aus und entwickeln daraus einen Algorithmus, bei dem im Wesentlichen lokale Minima auf einer transformierten Potentialfunktion gesucht werden.

$$\nabla_{\mathbf{x}^0} \langle U \rangle (\mathbf{x}^0, \Lambda) |_{\mathbf{x}^0 = \mathbf{x}_{min}^0} = 0 \quad (2.95)$$

$$\text{mit} \quad \langle U \rangle (\mathbf{x}^0, \Lambda) = \det^{-\frac{1}{2}}(2\pi\Lambda) \int U(\mathbf{x}) e^{-\frac{1}{2}[\mathbf{x} - \mathbf{x}^0]\Lambda^{-1}[\mathbf{x} - \mathbf{x}^0]} d\mathbf{x} \quad (2.96)$$

Dabei gelten die für \mathbf{x} bzw. \mathbf{x}^0 und Λ die Beziehungen 2.44 zu den Einteilchen-Parametern. Vergleicht man das mit Gleichung 2.11, die die Lösung der Diffusionsgleichung darstellt, dann stellt man fest, daß die Gleichungen ineinander übergehen, wenn man die Parameter Λ_{ii} wie folgt setzt:

$$\Lambda_{ii} = 2t \quad \text{für} \quad i = 1, \dots, N \quad (2.97)$$

Das zeigt nocheinmal genauer den anfangs erläuterten Zusammenhang zwischen allen Verfahren. Man kann die DEM als einen Spezialfall des adiabatic Gaussian Density Annealing betrachten, bei dem die Breiten aller Einteilchen-Gauß-Funktionen identisch sind und nicht als Funktion der Temperatur betrachtet, sondern nach einem geeigneten Schema Schrittweise verkleinert werden. Diese Vereinfachung entstand ursprünglich aus der deterministischen Herkunft des Algorithmus, so daß sich die dadurch fehlende Vorschrift zur Wahl des Glättungsparameters t als eines der größten Probleme des Verfahrens erweist.

Die Einschränkung auf eine einzige Breite der Verteilungsfunktionen wirkt sich ebenfalls negativ aus, da durch diese zusätzliche Vereinfachung nicht die lokalen Unterschiede, die zum Beispiel durch zufällige Anfangsbedingungen entstehen, berücksichtigt werden können. Das soll an einem Beispiel erläutert werden: Im Extremfall könnten sich alle bis auf ein Atom in einer energetisch günstigen Konfiguration befinden. Setzt man dann den Parameter t auf einen großen Wert, so verschwindet diese günstige Konfiguration wieder vollkommen. Beim aGDA hingegen würden sich die Breiten σ_i der günstig liegenden Atome schnell durch das Befinden in einem Potentialminimum verkleinern, während sich hauptsächlich das ungünstig liegende Atom auf einer stärker geglätteten Potentialfunktion bewegen würde.

Physikalisch läßt sich dieser Vergleich verstehen, wenn man den Zusammenhang zwischen Temperatur und der Breite der Verteilungsfunktion betrachtet. Bei hohen Temperaturen erstreckt sich die Verteilung über große Gebiete des Konformationsraumes, da

die kinetische Energie der Teilchen ihnen die Überwindung aller Energiebarrieren erlaubt. Das entspricht der Gasphase eines physikalischen Systems. Bei bestimmten Temperaturen treten Phasenübergänge auf, die sich durch eine deutliche Verringerung der Breite der Verteilung bemerkbar machen. Die Annealing Algorithmen halten sich in den Bereichen solcher Phasenübergänge länger auf, als in den anderen Temperaturbereichen, da ihre Bewegungsgleichungen diese Bereiche erkennen und die Λ_{ii} dementsprechend langsamer verändert werden. Das gewählte Schema zur Berechnung von t bei der DEM funktioniert aber vollkommen unabhängig von den Eigenschaften der Potentialfläche und geht daher unter Umständen zu schnell über solche Bereiche hinweg. Scheraga [6] versucht dieses Defizit auszugleichen, indem er seine Vorschrift zur Berechnung von t so wählt, daß Δt besonders klein ist bei großem t und nahe $t = 0$. Er versucht damit bei den beiden Phasenübergängen gasförmig-flüssig und flüssig-fest, die Verfolgung der Trajektorie $\mathbf{x}_{min}(t)$ sehr genau zu machen. Das ist mit Sicherheit eine besseres Schema, als die Glättungszeit gleichmäßig zu verringern, kann aber nicht die Bewegungsgleichungen der Annealing Verfahren ersetzen, denn die dynamische Berechnung kann in den entscheidenden Bereichen extrem kleine Veränderungen von Λ_{ii} zur Folge haben. Da dieser Bereich für jedes Atom auch noch bei verschiedenen Temperaturen auftreten kann, müßte man bei der DEM über einen weiten Bereich Δt extrem klein wählen, was zu einem enormen Anstieg der Laufzeit führen würde. Trotzdem würde auch dann die Vereinfachung auf eine Breite für die Verteilung aller Atome den Nachteil mit sich bringen, daß die lokalen Unterschiede für die Atome nicht berücksichtigt werden können. Um dem in der Diffusionsgleichung Rechnung tragen zu können, kann man sie allgemeiner formulieren:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \text{div} (D \cdot \nabla u(\mathbf{x}, t)) \quad (2.98)$$

Die Lösung dieser anisotropen Diffusionsgleichung hat dann eine ähnliche Form wie Gleichung 2.96:

$$u(\mathbf{x}, t) = \det^{-\frac{1}{2}}(4\pi D_t) \int U(\mathbf{y}) e^{-\frac{1}{4}(\mathbf{x}-\mathbf{y})^T D_t^{-1}(\mathbf{x}-\mathbf{y})} d\mathbf{y} \quad (2.99)$$

mit $D_t = D \cdot t$

Allerdings ist der Diffusionstensor D konstant, so daß auch so nicht die dynamische Anpassung der Breiten erreicht werden kann. Insgesamt stellt die DEM eben eine Vereinfachung des aGDA dar und kann deshalb nicht alle ihrer Eigenschaften beinhalten.

Bezieht man in den Vergleich die Verfahren ein, die mit einer Näherung an die freie Energie rechnen, so werden auch hier Gemeinsamkeiten sichtbar. Beim Effective Diffused Potential ist diese Gemeinsamkeit noch sehr deutlich, da die Näherung der inneren Energie identisch ist mit der des GDA und des QMA. Hinzu kommt lediglich der Entropieterm, der die freie Energie bei hohen Temperaturen zusätzlich konvexifiziert. Den Zusammenhang mit der PAM, die eine bessere Näherung an die Wahrscheinlichkeitsverteilung benutzt, sieht man, wenn man in Gl.2.73 den Übergang $\lim_{\beta \rightarrow 0}$ zu hohen Temperaturen macht. Dann kann man die Taylorentwicklung von $e^{-\frac{1}{2}\beta U}$ ohne großen Fehler nach dem linearen Term abbrechen und schreiben:

$$\lim_{\beta \rightarrow 0} H_{\Lambda, T}(\mathbf{x}) = \det^{-\frac{1}{2}}(4\pi\Lambda) \int U(\mathbf{x}') e^{-\frac{1}{4}(\mathbf{x}-\mathbf{x}')^T \Lambda^{-1}(\mathbf{x}-\mathbf{x}')} d\mathbf{x}' \quad (2.100)$$

$$- \beta^{-1} \ln \left[\det^{\frac{1}{2}}(8\pi\Lambda) \right] \quad (2.101)$$

$$= U_{\Lambda, \infty}(\mathbf{x}) - \beta^{-1} \ln \left[\det^{\frac{1}{2}}(8\pi\Lambda) \right] \quad (2.102)$$

In diesem Grenzwert besteht die effektive Energie aus einer Differenz des thermodynamisch gemittelten Potentials und einem räumlich unabhängigen Term, der proportional zum Logarithmus des Volumens des aktuell erreichbaren Konformationsraumes ist. Der zweite Term entspricht der Entropie des entsprechenden Makrozustandes und ist dem zweiten Anteil der freien Energie der EDP in Gl.2.86 sehr ähnlich. Der erste Term aus 2.102 ist wieder identisch mit dem gemittelten Potential der anderen Annealing-Verfahren, wenn Λ nur auf Diagonalmatrizen beschränkt wird. Dann gilt nämlich

$$U_{\Lambda, \infty}(\mathbf{x}, \Lambda) = \langle U \rangle (\mathbf{x}, 2\Lambda) \quad (2.103)$$

Die Packet Annealing Method hat aber dennoch enorme theoretische Vorteile gegenüber den anderen Verfahren. Für hohe Temperaturen stellt das Produkt aus Einteilchen-Gauß-Funktionen eine gute Näherung an die Gibbs/Boltzmann-Verteilung dar. Bei niedrigen Temperaturen jedoch genügt die Konzentration der Verteilung um einen Punkt im Konformationsraum nicht mehr aus, da getrennte Gebiete mit hohen Wahrscheinlichkeiten existieren. Dann hat die Näherung der PAM mit ihren getrennt lokalisierten Packets entscheidende Vorteile, da sie diese Eigenschaft der Verteilung besser wiedergeben kann. Aber um die effektive Energie der PAM berechnen zu können, müssen enorme Anstrengungen unternommen werden, die benutzten Potentiale so darzustellen, daß das Integral 2.73 berechnet werden kann. Bis jetzt ist aber keine Implementation des Verfahrens gelungen, die alle seine Eigenschaften beinhaltet, so daß die Vorteile zunächst nur theoretisch vorhanden sind.

Kapitel 3

Lokale Minimierungsverfahren

Beinahe alle in Kapitel 2 vorgestellten Verfahren beinhalten Schritte, bei denen eine lokale Minimierung durchgeführt wird. Es gibt eine Fülle von Verfahren, die die Minimierung einer Funktion sehr effizient durchführen können. Sie unterscheiden sich vor allem durch ihre Konvergenzverhalten und ihren numerischen Aufwand. In diesem Kapitel werden nur diejenigen Verfahren vorgestellt, die Anwendung bei der Implementierung der globalen Verfahren finden. Dabei werden keineswegs die effizientesten benutzt, sondern eher auf die einfache Handhabung bei Änderungen im Code geachtet. Insbesondere wird keine Methode verwendet, die die Berechnung der Hess'schen Matrix und deren Inverse erfordert, da der numerische Aufwand bei großen Problemen zu groß ist. Verfahren, die Näherungen an die inverse Hess'sche Matrix berechnen, wie zum Beispiel die BFGS-Methode, benötigen immernoch eine Funktion zur Aufdatierung der Matrix, die mit einem Aufwand von $O(n^3)$ skaliert, wobei n die Dimension des Problems darstellt. Deswegen werden im wesentlichen die Methoden des steilsten Abstiegs und der konjugierten Gradienten vorgestellt.

3.1 Die Verfahren

Man betrachtet im Allgemeinen eine Funktion

$$f : \mathbb{R}^n \rightarrow \mathbb{R}, \quad (3.1)$$

die mittels der globalen Verfahren minimiert werden soll. Ziel der lokalen Minimierungsmethoden ist es jedoch nur, von einem vorgegebenen Startvektor $\mathbf{x} \in \mathbb{R}^n$ aus Schritte in eine Richtung \mathbf{p} zu machen, die den Funktionswert kleiner werden läßt, bis ein \mathbf{x}_{min} gefunden ist, welches die Bedingung

$$\nabla_{\mathbf{x}} f(\mathbf{x})|_{\mathbf{x}_{min}} = 0 \quad (3.2)$$

erfüllt. Um sicher zu sein, daß der erreichte Punkt wirklich ein Minimum darstellt, müßte man alle Eigenwerte der Hess'schen Matrix

$$H = \nabla(\nabla f)^T \quad (3.3)$$

auf ihre Positivität überprüfen. Da man aber meist Verfahren benutzt, bei denen die explizite Form der Hess'schen Matrix nicht bekannt ist, begnügt man sich mit dem Erreichen der notwendigen Bedingung 3.2.

Man will also Schritte einer angemessenen Länge λ in eine Richtung \mathbf{p} machen, so daß

$$f(\mathbf{x} + \lambda\mathbf{p}) < f(\mathbf{x}) \quad (3.4)$$

für genügend kleines λ erfüllt wird. Das führt zu der Bedingung

$$(\nabla f)^T \mathbf{p} < 0 \quad (3.5)$$

an die Abstiegsrichtung \mathbf{p} . Danach muß man die Schrittweite λ bestimmen, die möglichst durch

$$\min_{\lambda > 0} f(\mathbf{x} + \lambda\mathbf{p}) \quad (3.6)$$

gegeben sein soll, also die optimale Länge in der verwendeten Richtung darstellt. Die verschiedenen Verfahren unterscheiden sich zunächst in der Wahl der Abstiegsrichtungen p . Die Schrittweite λ wird dann durch eine geeignete Methode der Liniensuche bestimmt, die meist statt des optimalen Wertes nur einen geschätzten Wert bestimmt, um Rechenzeit zu sparen.

3.1.1 Methode des steilsten Abstiegs

Bei der Wahl der Abstiegsrichtung ist es eine natürliche Überlegung, diejenige Richtung zu wählen, die für eine vorgegebene Schrittweite λ die Funktion f am stärksten verringert. Da die Verringerung von der Länge des Vektors \mathbf{p} abhängt, setzt man zunächst $|\mathbf{p}| = 1$ als Nebenbedingung fest und erhält damit

$$\mathbf{p}_{SD} = \min_{\mathbf{p}} [(\nabla f)^T \mathbf{p}] \quad \text{mit} \quad |\mathbf{p}| = 1 \quad (3.7)$$

Die Lösung von Gl.3.7 ist

$$\mathbf{p}_{SD} = -\frac{\nabla f}{|\nabla f|} \quad (3.8)$$

also der auf eins normierte negative Gradient der Funktion. Dieses \mathbf{p}_{SD} wird Richtung des steilsten Abstiegs (engl.: steepest descent) genannt. Damit kann man einen einfachen lokalen Minimierungsalgorithmus angeben:

Steepest-Descent-Algorithmus

Schritt 1: Gegeben ist ein Startvektor \mathbf{x}_0 , setze $k = 0$.

Schritt 2: Bestimme λ_k durch

$$\min_{\lambda_k > 0} f\left(\mathbf{x}_k - \lambda_k \frac{\nabla f(\mathbf{x}_k)}{|\nabla f(\mathbf{x}_k)|}\right) \quad (3.9)$$

Setze $\mathbf{x}_{k+1} = \mathbf{x}_k - \lambda_k \frac{\nabla f(\mathbf{x}_k)}{|\nabla f(\mathbf{x}_k)|}$.

Schritt 3: Falls Abbruchkriterium erreicht, breche ab, sonst setze $k = k + 1$ und gehe zu **Schritt 2**.

Dieser Algorithmus stellt ein relativ leicht zu implementierendes und sehr intuitives Verfahren dar, hat aber einen großen Nachteil. Man kann zeigen, daß der Algorithmus zu einem Minimum oder Sattelpunkt konvergiert, allerdings ist die Konvergenzrate nur linear und hängt entscheidend vom Problem ab. Bezeichnet man zum Beispiel den kleinsten und größten Eigenwert der als positiv definit angenommenen Hess'schen Matrix mit e_{min} und e_{max} und das Minimum mit \mathbf{x}_{min} , so gilt für \mathbf{x}_k

$$\limsup_{k \rightarrow \infty} \frac{|\mathbf{x}_{k+1} - \mathbf{x}_{min}|}{|\mathbf{x}_k - \mathbf{x}_{min}|} \leq c \quad \text{mit} \quad c \propto \left(\frac{e_{max} - e_{min}}{e_{max} + e_{min}} \right) \quad (3.10)$$

Für $e_{max} \approx e_{min}$ ist c sehr klein und die Konvergenz ist gut, aber für ein ungünstiges Verhältnis der Eigenwerte ist $c \approx 1$ und die Konvergenz kann sehr langsam sein. Deswegen sollte man im Allgemeinen nicht auf dieses Verfahren zurückgreifen. Allerdings stellt es unter schwierigen Bedingungen eine sichere Alternative dar, deren Implementierung wesentlich weniger Schwierigkeiten bereitet als die anderen Verfahren.

3.1.2 Konjugierte Gradienten

Wie oben erwähnt, hat die Methode des steilsten Abstiegs unter Umständen sehr schlechte Konvergenzeigenschaften. Deshalb ist diesem ein anderes Verfahren vorzuziehen, welches ebenfalls mit der Berechnung der Funktion und deren Ableitung auskommt, das Verfahren der Konjugierten Gradienten. Seinen Ursprung hat es bei der Anwendung auf lineare Gleichungssysteme der Art $A\mathbf{x} = \mathbf{b}$. Ist A nämlich eine positiv definite $n \times n$ -Matrix, so muß man von jedem Startvektor \mathbf{x}_0 aus höchstens n Schritte in zu A konjugierten Richtungen $\mathbf{r}_1, \dots, \mathbf{r}_n \in \mathbb{R}^n \setminus \{0\}$ ausführen, um die Lösung \mathbf{x}_{min} der Gleichung zu erhalten. Die Richtungen heißen konjugiert, wenn gilt

$$\mathbf{r}_i^T A \mathbf{r}_j = \delta_{ij} \quad 1 \leq i, j \leq n \quad (3.11)$$

Überträgt man dieses Verfahren auf die Minimierung allgemeiner Funktionen f , so ergibt sich ebenfalls eine sehr effiziente Methode, die vor allem bei hochdimensionalen Problemen ein starker Konkurrent der Newton-Verfahren ist.

Im Zusammenhang der nichtlinearen Optimierung benutzt man oft das Verfahren nach Fletcher und Reeves [20], welches wie folgt durchgeführt wird:

CG-Verfahren

Schritt 1: Gegeben ist ein Startvektor \mathbf{x}_0 , setze $k = 0$, $\mathbf{g}_0 = \nabla f$ und $\mathbf{d}_0 = -\mathbf{g}_0$.

Schritt 2: Falls Abbruchkriterium erreicht, breche ab, sonst berechne Schrittweite λ_k und

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k \quad (3.12)$$

$$\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1}) \quad (3.13)$$

$$\gamma_k = \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} \quad (3.14)$$

$$\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \gamma_k \mathbf{d}_k \quad (3.15)$$

Schritt 3: Setze $k = k + 1$ und gehe zu **Schritt 2**.

Es gibt auch andere Varianten mit konjugierten Gradienten, die sich durch die Berechnung der γ_k unterscheiden. Als vorteilhaft erweist sich noch eine kleine Änderung, bei der man $\gamma_k = 0$ setzt, falls k ein ganzes Vielfaches von n , der Dimension des Problems, ist.

Das Verfahren der konjugierten Gradienten hat immer noch schlechtere Konvergenzeigenschaften als ein (Quasi-)Newton-Verfahren, bei dem auch die zweiten Ableitungen der Funktion berechnet werden. Aber wegen seines geringen Rechenaufwandes ist es vor allem bei großen n klar vorzuziehen. Deswegen wurde es bei der Implementierung der globalen Verfahren in dieser Arbeit meistens benutzt.

3.2 Liniensuche

Wie in den vorherigen Kapiteln schon beschrieben, ist nicht nur die Wahl der Abstiegsrichtung sondern auch die der Länge λ_k entscheidend für das Konvergenzverhalten des Minimierungsverfahrens. Im Steepest-Descent-Algorithmus wurde das λ_k durch eine Minimierung von $f(\mathbf{x}_k + \lambda_k \mathbf{p}_k)$ bestimmt. Das ist natürlich keine gute numerische Methode, da wieder ein Minimierungsverfahren nötig wäre. Die Frage ist also, wie man die λ_k so wählen kann, daß sie weder zu klein noch zu groß sind und damit die Konvergenz verschlechtern. Das Problem bei zu großen Schritten ist, daß sie unter Umständen zwar die Bedingung $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$ erfüllen, kleinere Schritte aber trotzdem effektiver sein können, da sie einen größeren Abstieg im Verhältnis zur Länge bewirken können. Im Extremfall konvergiert das Verfahren bei zu großen Schrittweiten sogar gegen einen Punkt, der weder ein Minimum noch ein Sattelpunkt ist (siehe [19]). Eine einfache Möglichkeit, die Schritte nicht zu groß zu wählen, stellt die Schrittweitenbestimmung nach Armijo dar. Sie besagt, daß die Schrittweite λ_k die Bedingung

$$f(\mathbf{x}_k + \lambda_k \mathbf{p}_k) \leq f(\mathbf{x}_k) + \alpha \lambda_k \nabla f(\mathbf{x})^T \mathbf{p}_k \quad (3.16)$$

erfüllen muß. Dabei ist $\alpha \in (0, 1)$. Mit dieser Bedingung erreicht man, daß der Abstieg einen bestimmten Bruchteil des Gradienten im Punkt \mathbf{x}_k erreicht und somit nicht beliebig klein werden kann. Allerdings bewahrt diese Bedingung nicht vor zu kleinen Schritten, aber dieses Problem läßt sich noch einfacher umgehen. Man verwendet dazu meist das Prinzip des „Backtracking“, bei dem man zunächst immer versucht, große Schritte zu machen, und verkleinert λ_k solange, bis die Bedingung 3.16 erfüllt wird:

Bei den Implementierungen für diese Arbeit werden die Schrittweiten obigen Erläuterungen folgend deshalb wie folgt gewählt.

- Beim ersten Schritt wird zunächst eine große Länge von $\lambda_0 = 1$ versucht. λ_0 wird dann solange verkleinert ($\lambda_{k+1} = \lambda_k/1.5$), bis Bedingung 3.16 erfüllt ist.
- Bei jedem weiteren Schritt wird zunächst $\lambda_k = \lambda_{k-1}$ versucht und dann wieder verkleinert bis wiederum Bedingung 3.16 erfüllt ist.
- Etwa alle 20 Schritte wird $\lambda_k = 1.5 \cdot \lambda_{k-1}$ gesetzt, aber wenn nötig wieder wie oben verkleinert.

Mit diesen Vorschriften erreicht man Schrittweiten, die ein gutes Konvergenzverhalten des Verfahrens zur Folge haben und trotzdem einen geringen numerischen Aufwand erfordern.

3.3 Abbruchkriterien

Das Abbruchkriterium ist für einen Minimierungsalgorithmus ebenso wichtig wie die Wahl der Abstiegsrichtung und der Schrittweiten, da es entscheidend dafür verantwortlich ist, ob man wirklich in einem Minimum landet. In der Literatur werden allerdings oft Abbruchkriterien angegeben, die nur schlecht für Minimierungsverfahren geeignet sind. Man möchte den Algorithmus dann beenden, wenn entweder das Problem gelöst ist (oder nicht gelöst werden kann) oder der Aufwand, eine Lösung zu erreichen, zu groß ist. Im Prinzip gibt es drei Parameter, aus denen man schließen kann, ob noch weitere Schritte gemacht werden sollten. Zum einen kann man die Änderung der Funktionswerte $\Delta f_k = f_{k-1} - f_k$ beobachten und abbrechen, wenn sie unter einen bestimmten Wert sinken, also $\Delta f_k < \epsilon_f$. Dieses Verhalten weist allerdings keinesfalls immer auf das Erreichen eines Minimums hin und ist daher ungeeignet.

Die notwendige Bedingung für das Erreichen eines Minimums ist $|\nabla f| = 0$. Will man das auf ein numerisches Verfahren übertragen, so kann man fordern

$$|\nabla f| < \epsilon_{\nabla} \quad (3.17)$$

oder noch einfacher

$$\max_{i \leq n} |\nabla_{x_i} f| < \epsilon_{\nabla} \quad (3.18)$$

Erfüllt der Algorithmus diese Bedingung, so hat er in einem Extremum oder einem Sattelpunkt gestoppt. Falls er vorher einige Schritte durchgeführt hat, so waren diese absteigend und man kann ein Maximum ausschließen. Wurde er allerdings aus irgendeinem Grund an einem Punkt gestartet, der bereits das Abbruchkriterium erfüllt, so kann es sich auch um ein Maximum handeln! Um das Erreichen eines Minimums sicherzustellen müßte man die Hess'sche Matrix auf ihre positive Definitheit überprüfen, was sich aber in der Praxis als unnötig erweist, da nur selten Sattelpunkte oder gar Maxima angenommen werden.

Um zu überprüfen, ob mit akzeptablem Aufwand noch eine Lösung erreicht werden kann, beobachtet man Δf_k in Abhängigkeit von der Schrittweite λ_k . Fällt die Schrittweite unter einen bestimmten Wert ϵ_{λ} und wird das Verhältnis $\Delta f_k / \lambda_k$ kleiner eins, so bricht man ab, obwohl Bedingung 3.18 nicht erreicht ist. Das Verfahren kann dann in akzeptabler Zeit kein Minimum finden. Ebenso bricht man ab, wenn die Schrittweite einige Schritte lang zu große Werte von zum Beispiel $\lambda_k > 10$ annimmt. Dann divergiert das Verfahren und wird ebenfalls kein Minimum finden.

Zu beachten ist bei den beiden oben vorgestellten Abbruchkriterien, daß sie natürlich von der Skala von f abhängen. Man kann also in der dargestellten Form kein allgemeingültiges ϵ_{∇} oder ϵ_{λ} angeben. Im Zusammenhang dieser Arbeit wird normalerweise gefordert, daß die Werte der Potentiale und deren Ableitungen auf 10^{-6} genau sind, deshalb wird, wenn nicht anders angegeben,

$$\epsilon_{\nabla} = \epsilon_{\lambda} = 10^{-6} \quad (3.19)$$

gesetzt.

Kapitel 4

Optimierung von Lennard-Jones-Clustern

4.1 Das Lennard-Jones-Potential

Um die in Kapitel 2 vorgestellten Verfahren an einem relativ einfachen Potential zu testen, werden sie in diesem Kapitel auf Cluster von neutralen Atomen angewendet, zwischen denen ausschließlich das Lennard-Jones Potential wirkt. Physikalische Beispiele für Atome, die gut durch dieses Potential beschrieben werden, sind die Edelgase wie Helium und Argon. Das Lennard-Jones-Potential hängt ausschließlich vom Abstand zweier Partikel ab und hat die Form

$$U_{ij}^{LJ} = 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] \quad \text{mit} \quad r_{ij} = |\vec{x}_i - \vec{x}_j| \quad (4.1)$$

wenn die Atome i und j sich an den Orten \vec{x}_i und \vec{x}_j befinden. Es hat sein Minimum bei $r_{ij}^{min} = 2^{\frac{1}{6}}\sigma_{ij}$ und es gilt $U(r_{ij}^{min}) = -1$. Es besteht aus einem anziehenden Teil $\propto \frac{1}{r^6}$,

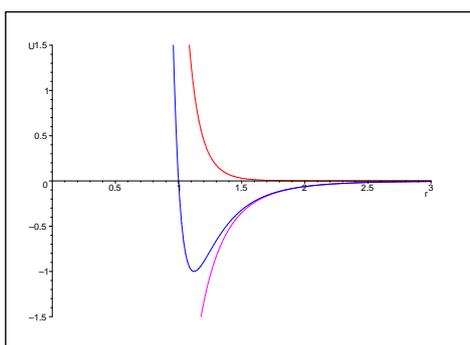


ABBILDUNG 4.1: U_{ij}^{LJ} (—), $\frac{1}{r^{12}}$ (—), $-\frac{1}{r^6}$ (—)

der die Van-der-Waals Anziehung neutraler Atome repräsentiert. Diese Anziehung hat ihren Ursprung in der induzierten Dipol-Dipol-Wechselwirkung der Elektronenwolken. Solche Dipolmomente entstehen durch Fluktuationen der ansonsten sphärisch symmetrischen Ladungsverteilungen und sind der Grund für die $\frac{1}{r^6}$ Proportionalität der Van-der-Waals

Anziehung. Der zweite Teil des Lennard-Jones-Potentials wirkt bei kleinen Abständen stark abstoßend, was durch den Anstieg der Energie zu erklären ist, der entsteht, wenn sich die Ladungswolken verschiedener Atome überlappen. Der Exponent -12 beschreibt dieses abstoßende Verhalten ausreichend, wird allerdings auch aus Gründen effizienter numerischer Berechenbarkeit gewählt.

Die Parameter ϵ_{ij} bzw. σ_{ij} haben die Einheiten einer Energie bzw. eines Abstandes und werden normalerweise mithilfe der Lorentz-Berthelot'schen Mischungsregeln aus Parametern der Atome berechnet:

$$\epsilon_{ij} = \sqrt{\epsilon_i \epsilon_j} \quad \sigma_{ij} = \frac{\sigma_i + \sigma_j}{2} \quad (4.2)$$

Das Potential eines Clusters aus N identischen Atomen besteht dann aus der Summe der $\frac{N(N-1)}{2}$ Paar-Potentiale und hat somit folgende Form:

$$U = \sum_{i=1}^N \sum_{j=i+1}^N U_{ij}^{LJ} \quad (4.3)$$

Für $N \leq 4$ hat diese Funktion genau ein Minimum, aber für größere N steigt die Zahl der lokalen Minima sehr schnell an. Nach einer Schätzung von Hoare [22], die er mittels einer Interpolation der bekannten Zahlen an lokalen Minima von Clustern mit bis zu neun Atomen erhielt, steigt ihre Zahl wie $O(1.03^{N^2})$. Für $N = 20$ ergibt das noch $\sim 10^5$ Minima, während für $N = 50$ bereits $\sim 10^{32}$ zu erwarten sind. Auch wenn die meisten dieser Minima sehr hohe potentielle Energie haben, können solche Cluster nicht mehr durch klassische Verfahren wie dem Simulated Annealing (in time) oder gar durch einfaches Absuchen der Potentialfläche minimiert werden.

Für atomare Cluster ist es in der Regel relativ einfach, Konfigurationen minimaler Energie aufzubauen, indem man die Atome, basierend auf bekannten Gitterstrukturen wie zum Beispiel einem fcc- oder bcc-Gitter, aufbaut. Ordnet man sie dann noch mit minimaler Oberfläche an, so hat man bereits gute Kandidaten für globale Minima. Viele Literaturwerte für Lennard-Jones-Cluster sind so entstanden. Hier sollen aber die vorgestellten Verfahren angewendet werden, um ihre Tauglichkeit im Vergleich beurteilen zu können und um dann ausgewählte Verfahren auf wesentlich schwierigere Optimierungsprobleme anzuwenden. Deshalb werden an keiner Stelle Startkonfigurationen mithilfe solchen Wissens erzeugt.

Wie schon erwähnt werden durch das Lennard-Jones-Potential physikalische Systeme aus Helium- oder Argon-Atomen gut beschrieben. Da in diesem Kapitel aber vorrangig die globale Minimierungverfahren getestet werden sollen, werden nur Cluster aus identischen Atomen behandelt und alle Parameter auf eins skaliert:

$$\epsilon_i = \sigma_i = 1 \quad i = 1, \dots, N \quad (4.4)$$

Dann kann man die Energien der gefundenen Minima mit anderen Literaturwerten¹ vergleichen. Natürlich stellen die Ergebnisse trotzdem auch Minima der modellierten physikalischen Systeme dar, werden hier aber nicht in physikalische Einheiten umgerechnet. Zudem können geringste Änderungen des Potentials vollkommen andere minimierte Strukturen zur Folge haben (siehe [21]), weshalb man mit der Übertragung der Ergebnisse auf physikalische Systeme generell vorsichtig sein sollte.

¹Für Lennard-Jones-Cluster mit bis zu 150 Atomen gibt es in der „Cambridge Cluster Database“ [10] die besten Potentialwerte, die bis zu diesem Zeitpunkt gefunden wurden.

4.2 Anwendung der DEM

Bei Clustern aus Lennard-Jones-Atomen hat man ein Potential der Form

$$U(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N) = \sum_{i=1}^N \sum_{j=i+1}^N U_{ij}^{LJ}(r_{ij}) = \sum_{i=1}^N \sum_{j=i+1}^N 4 \left(\left(\frac{1}{r_{ij}} \right)^{12} - \left(\frac{1}{r_{ij}} \right)^6 \right) \quad (4.5)$$

$$\text{mit } r_{ij} = |\vec{x}_i - \vec{x}_j|$$

Die Parameter σ und ϵ wurden auf eins gesetzt. Um nun die DEM anwenden zu können, muß man in der Lage sein, die Lösung

$$u(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, t) = (4\pi t)^{-\frac{3N}{2}} \int U(\vec{y}_1, \vec{y}_2, \dots, \vec{y}_N) e^{-\frac{1}{4t} \sum_{i=1}^N (\vec{x}_i - \vec{y}_i)^2} d^3y_1 d^3y_2 \dots d^3y_N \quad (4.6)$$

der Gleichung 2.9 schnell zu berechnen. Für das oben angegebene Potential kann man das Ergebnis jedoch nicht analytisch angeben. Eine numerische Berechnung kommt bei so hochdimensionalen Problemen allerdings auch nicht in Frage, so daß man einen anderen Weg gehen muß:

Man weiß, daß die Faltung zweier Gauß-Funktionen wieder eine Gauß-Funktion ergibt. Das kann man sich zu nutze machen, indem man versucht, das (Lennard-Jones-)Potential hinreichend gut durch die Summe einiger Gauß-Funktionen anzunähern.

$$U_{ij}^{LJ}(r_{ij}) \approx \tilde{U}_{ij}(r_{ij}) = \sum_{k=1}^L a_k^{ij} e^{-b_k^{ij} r_{ij}^2} \quad (4.7)$$

$$\tilde{U}(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^L a_k^{ij} e^{-b_k^{ij} r_{ij}^2} \quad (4.8)$$

Da das Cluster nur gleichartige Atome enthalten soll, kann man die Indizes bei den Parametern weglassen und schreiben: $a_k^{ij} = a_k$, $b_k^{ij} = b_k$. Setzt man dieses Potential in die Lösung 4.6 ein, so erhält man das folgende Ergebnis:

$$\tilde{u}(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, t) = \sum_{i=1}^N \sum_{j=i+1}^N \tilde{u}_{ij}(\vec{x}_i, \vec{x}_j, t) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^L a_k (1 + 8b_k t)^{-\frac{3}{2}} e^{-\frac{b_k r_{ij}^2}{1+8b_k t}} \quad (4.9)$$

Mit dieser analytischen Lösung der Diffusionsgleichung kann man nun den Algorithmus effizient implementieren.

Um das Lennard-Jones-Potential mit der Näherung 4.7 ausreichend genau darzustellen, genügt schon eine Summe aus zwei Gauß-Funktionen ($L = 2$). Trotzdem ist es (nach Scheraga [6]) sinnvoll, einen weiteren Summanden der Näherung hinzuzufügen, damit das modellierte Potential eine bestimmte Eigenschaft hat, die im folgenden näher erläutert wird.

Man stellt fest, daß das genäherte Potential $\tilde{u}_{ij}(r_{ij}, t)$ sich in Abhängigkeit der Parameter a_k und b_k für größer werdendes t verschieden verhält. Und zwar hängt das Verhalten vom Wert des Integrals

$$I_{ij} = \int \tilde{u}_{ij}(|\vec{x}|, t) d^3x = \pi^{\frac{3}{2}} \sum_{k=1}^L a_k \cdot b_k^{-\frac{3}{2}} \quad (4.10)$$

ab. Ist das Ergebnis nicht negativ, also $I_{ij} \geq 0$, so bewegt sich das Minimum der Funktion $\tilde{u}_{ij}(r_{ij}, t)$ mit größer werdendem t ebenfalls zu größeren Werten von r_{ij} hin. Man kann das daran erkennen, daß die Funktion für $r_{ij} = 0$ stets einen positiven Wert annimmt, da das Integral 4.10 unabhängig von t ist. Weil $\tilde{u}_{ij}(r_{ij}, t)$ für $r_{ij} \rightarrow \infty$ gegen null geht, muß dazwischen ein Minimum liegen. Für die Atome des Clusters würde das bedeuten, daß sie bei großem Glättungsparameter auch große Abstände voneinander einnehmen und nur noch sehr schwach miteinander interagieren würden. Das führt bei den Rechnungen meist zu großen numerischen Problemen und sollte daher möglichst vermieden werden.

Die Situation ändert sich jedoch vollkommen, wenn das Integral negativ wird, also $I_{ij} < 0$. Tritt dieser Fall ein, so existiert eine Zeit $t_{max} > 0$, ab der die Funktion $\tilde{u}_{ij}(r_{ij}, t)$ ihr Minimum bei $r_{ij} = 0$ und kein Maximum mehr hat. Das Potential wirkt dann ausschließlich anziehend auf die Atome, wodurch sie sich bewegen, bis sie sich alle auf demselben Punkt befinden. Diese Konfiguration kann man gut als Startposition für den Algorithmus verwenden, da man weiß, daß zu dieser Zeit kein weiteres Minimum existiert.

Wie oben schon erwähnt, reicht zur angemessenen Näherung des Lennard-Jones-Potentials eine Summe von nur zwei Gauß-Funktionen vollkommen aus. Mit den Parametern aus Tabelle 4.1 tritt dann aber eben die unerwünschte Situation ein, daß der Wert des Integrals I_{ij} kleiner null ist. Um das zu beheben, kann man zwei verschiedene Wege einschlagen. Man könnte die Höhe des gefitteten Potentials bei $r_{ij} = 0$ verringern, indem man mehr als zwei Summanden in Gleichung 4.7 benutzt, und so den Wert des Integrals verringern. Der einfachere Weg ist aber, der Summe eine dritte Gauß-Funktion hinzuzufügen, die eine große Breite und kleine, negative Höhe hat. Diese Modifikation verschiebt in den relevanten Bereichen das Potential nur nach unten, wodurch sich die Eigenschaften kaum ändern. Anschaulich kann man sich vorstellen, daß die dritte Gauß-Funktion, die für sich allein gesehen ausschließlich anziehend wirkt, bei großem Glättungsparameter das Cluster zusammenhält. Die Abbildungen 4.2 und 4.3 zeigen die Funktion \tilde{U}_{ij} mit und ohne diese Korrektur.

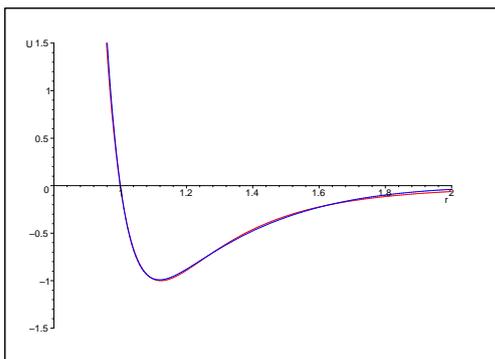


ABBILDUNG 4.2: An Lennard-Jones-Potential (—) gefittete Funktion \tilde{U}_{ij} (—) mit $L = 2$

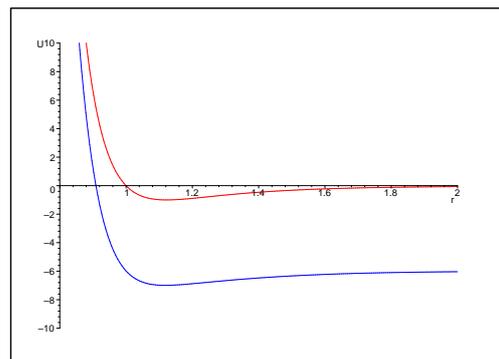


ABBILDUNG 4.3: An Lennard-Jones-Potential (—) gefittete Funktion \tilde{U}_{ij} (—) mit $L = 3$

Mit dem so gewählten Potential will man den Algorithmus zur Zeit t_{max} beginnen lassen, wo das Potential ausschließlich anziehend wirkt und alle Atome sich am selben Platz befinden. Um diese Zeit zu berechnen, betrachtet man die zweite Ableitung von

k	a_k	b_k
1	$1.7544 \cdot 10^4$	9.3197
2	-5.3556	1.2346
3	-6.0	$1.0 \cdot 10^{-4}$

TABELLE 4.1: Parameter von \tilde{U}_{ij}

$\tilde{u}_{ij}(r_{ij}, t)$ an der Stelle $r_{ij} = 0$:

$$\left. \frac{\partial^2 \tilde{u}_{ij}(r, t)}{\partial r^2} \right|_{r=0} = -2 \sum_{k=1}^L a_k b_k (1 + 8b_k t)^{-\frac{5}{2}} \quad (4.11)$$

Für kleine t ist die zweite Ableitung negativ und das Potential hat ein Maximum bei $r = 0$. Da für $t > t_{max}$ dort ein Minimum existieren soll, bestimmt man t_{max} durch

$$-2 \sum_{k=1}^L a_k b_k (1 + 8b_k t_{max})^{-\frac{5}{2}} = 0 \quad (4.12)$$

Die Lösung dieser nichtlinearen Gleichung ist $t_{max} \approx 32.4730$. Für alle Zeiten größer t_{max} hat das Gesamtpotential $\tilde{u}(\vec{x}_1, \vec{x}_2, \dots, t)$ nur noch ein Minimum bei $r_{ij} = 0 \quad \forall \quad i, j$. Man kann dann den Algorithmus bei einer Glättungszeit $t_0 > t_{max}$ in Schritt 1 beginnen.

Als nächste Frage stellt sich, wie man die Zeitschrittweiten Δt wählt. Einerseits sollten die Schrittweiten nicht zu groß sein, um Sprünge in andere Trajektorien zu vermeiden, andererseits will man den Algorithmus auch in einer angemessenen Zeit beenden, sollte also auch keine allzu kleinen Schrittweiten wählen. Scheraga [6] schlägt folgende Prozedur vor, da die Änderungen von \tilde{u} in der Nähe von t_{max} und $t = 0$ am größten sind:

- Für $t \geq t_s$ setze: $t_k = t_0 - \alpha k^2$ (mit $t_s = 1.0$ und $\alpha = 0.001$)
- Für $t < t_s$ setze: $t_k = t_s \left(1 - \frac{k-s}{Q-s}\right)^2$ (wobei Q die Gesamtzahl der Schritte ist, benutzt wurde $Q = s + 500$)

Mit diesen Vorarbeiten kann man den DEM-Algorithmus verwenden. Man minimiert zu jeder Zeit t_k das Potential \tilde{u} , bis man bei $t = 0$ angelangt ist. Um dann den Wert des Lennard-Jones-Potentials für die errechnete Konfiguration der Atome zu erhalten, ersetzt man zuletzt noch das Potential $\tilde{u}(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, t = 0)$ durch das ursprüngliche Potential $U(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N)$ und minimiert auch hier noch einmal, um den kleinen Fehler durch die Näherung der Funktion mittels Gauß-Funktionen nicht im Ergebnis zu haben.

In Tabelle 4.2 sind die Ergebnisse der Berechnungen für Cluster mit fünf bis 33 Atomen dargestellt. Für jede Zahl von Atomen wurden 100 Rechnungen mit verschiedenen Anfangskonfigurationen bei $t = t_{max}$ durchgeführt. Verwendet wurde dabei zum Vergleich der lokalen Minimierungsalgorithmen sowohl das Steepest-Descent- als auch das CG-Verfahren. Die Tabelle zeigt die Potentialwerte des jeweiligen globalen Minimums aus der Literatur [10], die Werte des tiefsten gefundenen Minimums, die prozentuale Erfolgsquote für das Finden dieses Minimums und die durchschnittliche Rechenzeit².

²Die Berechnungen wurden auf einem PentiumIII-Xeon Prozessor mit 866MHz und 1GB RAM durchgeführt.

# Atome	U_{global}	Steepest-Descent			CG		
		U_{min}	% Min	$t_{mean}[s]$	U_{min}	% Min	$t_{mean}[s]$
5	-9.1039	-9.1039	100	0.5976	-9.1039	100	0.4542
6	-12.7121	-12.7121	100	0.8224	-12.7121	100	0.6661
7	-16.5054	-16.5054	100	1.4057	-16.5054	100	1.1906
8	-19.8215	-19.7653	100	1.7688	-19.7653	100	1.5789
9	-24.1134	-23.2698	100	2.5513	-23.2698	100	2.3552
10	-28.4225	-26.7716	100	3.4282	-26.7716	100	3.2241
11	-32.7660	-32.7660	100	5.4308	-32.7660	100	5.3713
12	-37.9676	-33.5975	100	4.7691	-33.5975	100	5.1208
13	-44.3268	-44.3268	100	6.2979	-44.3268	100	6.7050
14	-47.8452	-47.8452	36	8.3891	-47.8452	100	9.2404
15	-52.3226	-50.1713	100	9.7397	-52.3226	100	11.8646
16	-56.8157	-56.8157	8	14.6232	-54.8465	100	14.5972
17	-61.3180	-58.5534	100	14.0955	-58.5534	100	14.1438
18	-66.5309	-66.5309	11	17.2589	-66.5309	88	18.9916
19	-72.6598	-68.1193	100	19.5684	-68.1193	100	19.6584
20	-77.1770	-77.1770	7	27.2388	-77.1770	10	29.0911
21	-81.6846	-81.6348	82	44.2526	-81.6348	50	31.9040
22	-86.8098	-86.0326	1	44.5693	-86.8098	4	36.2621
23	-92.8445	-92.8445	100	49.2486	-92.8445	100	38.5370
24	-97.3488	-97.3488	4	61.4411	-97.3488	21	46.7516
25	-102.3727	-100.1137	100	48.0893	-100.1137	87	48.9538
26	-108.3156	-108.3156	100	65.5952	-108.3156	100	48.7723
27	-112.8736	-111.3921	15	87.8001	-111.3921	97	65.5127
28	-117.8224	-117.2458	74	91.8810	-117.2458	81	68.2386
29	-123.5874	-123.5874	68	95.7551	-123.5874	83	71.9441
30	-128.2866	-122.8138	20	88.5458	-122.8138	100	77.7936
33	-144.8427	-142.4784	9	119.9773	-142.4784	97	97.3679

TABELLE 4.2: Mit der DEM gefundene Minima

Der direkte Vergleich der beiden Minimierungsverfahren zeigt den Vorteil des CG-Algorithmus. Er ist vor allem bei großen Clustern schneller und funktioniert sicherer als der einfache Steepest-Descent-Algorithmus. Trotzdem ist das Ergebnis für beide nicht zufriedenstellend, denn der Gesamtalgorithmus landet keineswegs immer im gleichen Minimum, obwohl es trotz der durchaus unterschiedlichen Startkonfigurationen bei $t = t_{max}$ nur ein Minimum gibt. Das kann entweder an einer zu großen Schrittweite bei dem Glättungsparameter oder an der numerischen Genauigkeit bei dem lokalen Minimierungsverfahren liegen, die stets gleich gewählt wurde. Das soll im Folgenden genauer am Beispiel des 14-Atom Clusters unter Verwendung des Steepest-Descent Verfahrens untersucht werden.

Dazu kann man sich zunächst einmal den Verlauf der in Kapitel 2.2.1 vorgestellten Geschwindigkeit v_T ansehen. Man erwartet verschiedene Graphen in Abhängigkeit vom Minimum, welches bei dem jeweiligen Rechnung gefunden wurde. Und tatsächlich sieht man in den Abbildungen 4.4 und 4.5 einen deutlichen Unterschied zwischen den beiden Verläufen. Es treten an völlig verschiedenen Stellen Sprünge auf, woraus man schließen

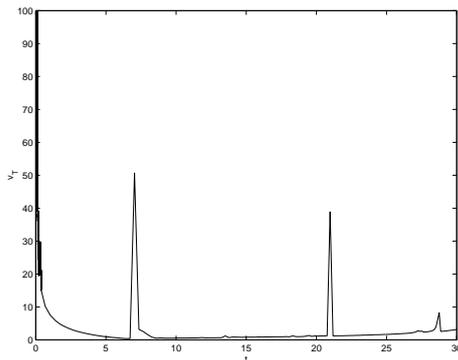


ABBILDUNG 4.4: v_T für das 14-Atom Cluster mit $U_{min} = -47.845157$

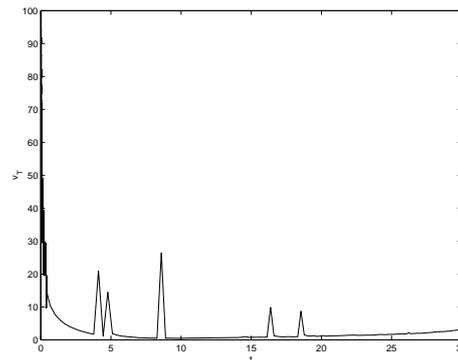


ABBILDUNG 4.5: v_T für das 14-Atom Cluster mit $U_{min} = -44.276342$

kann, daß beide Systeme auch völlig verschiedene Trajektorien bis zum ungeglätteten Potential verfolgt haben. Der erste deutliche Unterschied ist bei $t \approx 28.7$ zu erkennen, d.h. obwohl beide Systeme zur Zeit $t = t_{max}$ innerhalb der numerischen Genauigkeit im selben Minimum gelandet sind, schlagen sie zu einer anderen Zeit verschiedene Wege ein. Erhöht man nun die Anzahl der Schritte für t , macht also die Schrittweite kleiner, so ändert sich an diesem Verhalten nichts, wie Tabelle 4.3 zeigt. Der Grund muß also bei der

Q	U_{min}^1	U_{min}^2	$t_{mean}^1[s]$	$t_{mean}^2[s]$	$\#(U_{min}^1)/\#(U_{min}^2)$
577	-47.845157	-44.276342	8.5732	8.2856	36/64
1253	-47.845157	-44.276342	12.0676	11.8990	34/66
5564	-47.845157	-44.276342	23.2978	23.5711	27/73

TABELLE 4.3: Vergleich der Ergebnisse für das 14-Atom Cluster bei verschiedenen Zeitschrittweiten

numerischen Genauigkeit liegen. Wie oben beschrieben, wird als Abbruchbedingung für die lokale Minimierung unter anderem der Betrag des Gradienten benutzt. Und zwar wird abgebrochen, wenn

$$\max_{i \leq N} |\nabla_{\vec{x}_i} \tilde{u}(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, t)| < \epsilon_{\nabla} \quad (4.13)$$

gilt. Da sich die Energieskala durch die Glättung ändert, kann ϵ_{∇} nicht konstant gewählt werden, sondern muß dieser Änderung angepaßt werden. Für die Ergebnisse aus Tabelle 4.2 wurde folgende Regel benutzt:

$$\epsilon_{\nabla} = 10^{-6} \cdot (10^{-5})^{\frac{t}{t_{max}}} \quad (4.14)$$

Um nun zu überprüfen, ob eine Änderung bei der Berechnung von ϵ_{∇} das Verfahren eindeutiger macht, wurden nocheinmal für 14 Atome 100 Rechnungen mit

$$\epsilon_{\nabla} = 10^{-6} \cdot (10^{-7})^{\frac{t}{t_{max}}} \quad (4.15)$$

durchgeführt. Da die Genauigkeit bei $t = 0$ offensichtlich ausreicht - die gefundenen globalen Minima stimmen in genügend vielen Stellen mit den Vergleichswerten überein - wurde nur die zweite Zehnerpotenz verändert. Diese Änderung bedeutet natürlich auch

# Atome	U_{global}	U_{min}	% Min	$t_{mean}[s]$
14	-47.845157	-47.845157	100	34.8916
16	-56.815742	-56.815742	1	55.3456
18	-66.530949	-66.530949	52	77.4536
20	-77.177043	-77.177042	65	108.2488

TABELLE 4.4: Ergebnisse der DEM mit höhere numerischer Genauigkeit

eine erheblich längere Laufzeit, wie das Ergebnis zeigt (siehe Tabelle 4.4). Für das 14-Atom Cluster funktioniert das Verfahren dafür jetzt absolut eindeutig. Aber Rechnungen mit den gleichen Änderungen bei den anderen Atomzahlen, die vorher keine eindeutigen Ergebnisse lieferten, haben das Ergebnis in zwei Fällen zwar verbessert, aber sind noch nicht zufriedenstellend. In einem Fall scheint das Resultat sogar schlechter geworden zu sein. Man muß dabei allerdings beachten, daß das tiefere Minimum auch dasjenige sein kann, welches durch fehlende Genauigkeit oder zu große Zeitschrittweite eher zufällig gefunden wurde. Um über das Ergebnis bei 16 Atomen genauere Aussagen zu machen, muß man sich auch die Häufigkeit der anderen gefundenen Minima anschauen. Man sieht dann, daß 99 der 100 Läufe in drei anderen Potentialwerten landen, von denen wiederum 56 bei $U_{min} = -54.846542$ enden. Das läßt vermuten, daß das der Wert des richtigen Minimums ist, was man durch eine erneute Durchführung der Rechnung mit der erhöhten Anzahl von Zeitschritten $Q = 5564$ und der ursprünglichen Genauigkeit aus 4.14 bestätigen kann. Dann erscheint nämlich zu 100% dieses Minimum. In diesem Fall ist also eine kleinere Zeitschrittweite nötig gewesen, während die numerische Genauigkeit ausreichend war. Allgemein ist es demnach unmöglich die Parameter für alle Anzahlen von Atomen sinnvoll einheitlich zu setzen. Denn es spielt offenbar nicht nur eine Rolle, wieviele Atome das System hat, sondern die Güte des Verfahrens ist stark beeinflusst von nicht direkt erkennbaren Charakteristika der Potentialfläche.

4.2.1 Probleme mit dem Verfahren nach Scheraga

Wie man den obigen genaueren Betrachtungen entnehmen kann, funktioniert das von Scheraga [6] vorgeschlagene Verfahren nur in wenigen Ausnahmefällen zufriedenstellend. Es besitzt aber keineswegs die Eigenschaften eines numerischen Verfahrens, die für eine sichere Anwendung nötig sind. So sollte man die Parameter nicht an die jeweilige Größe des Problems anpassen müssen. Mit dieser Anpassung erreicht man dann zwar eine gewisse Eindeutigkeit des Ergebnisses, jedoch verschlechtert das oft wiederum das Resultat in Bezug auf die eigentliche Zielsetzung, das globale Minimum einer Funktion zu bestimmen. Ohne Vergleichswerte kann man dann keine Aussage über die Güte des Ergebnisses machen, was aber kaum als negativ zu bewerten ist, da ein solches Verfahren bis heute nicht existiert.

Die Probleme werden noch sichtbarer, wenn man sich etwas größeren Problemen zuwendet. Die Tabelle 4.2 wies für 30 und 33 Atome noch relativ gute Ergebnisse vor. Versucht man nun auch hier die Genauigkeit und die Zeitauflösung zu verbessern, so erhält man erschreckend schlechte Resultate, die manchmal sogar im positiven Potentialbereich liegen. Vor allem liefert jede neue Wahl der Parameter auch neue Ergebnisse. Das Problem ist, daß das Verfahren die Trajektorie vom stark geglätteten bis zum original Potential

nicht sicher genug verfolgen kann und auch nicht anzeigt, wenn das gar nicht möglich sein sollte. Denn Trajektorien können auch, wie schon in Kapitel 2.2.1 beschrieben, enden, bevor $t = 0$ erreicht worden ist. In solchen Fällen ist es vollkommen sinnlos, einen Startpunkt bei $t = t_{max}$ zu wählen. Startpunkte bei kleinerem t garantieren aber kein eindeutiges Minimum bei dieser Zeit, wodurch das Ergebnis von der Startkonfiguration abhängig wird.

Die Notwendigkeit, die Parameter an die jeweilige Clustergröße anpassen zu müssen, zeigt deutlich die Probleme, die entstehen, wenn das Verfahren keine Bewegungsgleichung für t aus sich heraus liefert. Jede Wahl eines Berechnungsschemas kann nur für wenige Fälle gut funktionieren und ist somit nicht allgemeingültig wählbar. Deshalb wurde im Zusammenhang dieser Arbeit auch versucht, die Notwendigkeit dieser Wahl von t bzw. Δt aus dem Verfahren zu eliminieren. Der Gedanke dabei war, daß durch die Anwendung der Diffusion die Potentialfunktion $\tilde{U}(\mathbf{x})$ mit $\mathbf{x} \in \mathfrak{R}^{3N}$ in den um eine Dimension höheren Raum \mathfrak{R}^{3N+1} eingebettet wird. Da die Lösung der Diffusionsgleichung das sogenannte Maximum-Minimum-Prinzip erfüllt (siehe [12]),

$$\inf_{\mathfrak{R}^{3N}} \tilde{U}(\mathbf{x}) \leq \tilde{u}(\mathbf{x}, t) \leq \sup_{\mathfrak{R}^{3N}} \tilde{U}(\mathbf{x}) \quad t \geq 0 \quad (4.16)$$

ist das globale Minimum \mathbf{x}_{min} der Funktion \tilde{U} auch immer das Minimum der mit dem Gauß-Kern gefalteten Funktion \tilde{u} an der Stelle $(\mathbf{x}_{min}, t = 0)$. Das heißt, man kann statt die Trajektorie $\mathbf{x}_{min}(t)$ eines Minimums zu verfolgen, indem man zu festen Zeiten $\tilde{u}|_{t=const}$ minimiert, auch eine lokale Minimierungsprozedur auf \tilde{u} unter Benutzung des vollen Gradienten $\nabla_{\mathbf{x},t}\tilde{u}$ anwenden. Zu beachten ist dabei nur, daß die Funktion nur für $t \geq 0$ definiert ist, aber keineswegs immer

$$\frac{\partial \tilde{u}}{\partial t} \Big|_{\mathbf{x}=\mathbf{x}_{min}, t=0} = 0 \quad (4.17)$$

gilt, so daß man die Minimierung abbrechen muß, wenn $t = 0$ erreicht worden ist, weil die Bedingung für ein Extremum

$$\nabla_{\mathbf{x},t}\tilde{u}(\mathbf{x}, t) \Big|_{\mathbf{x}=\mathbf{x}_{min}, t=t_{min}} = 0 \quad (4.18)$$

unter Umständen nie erreicht werden kann. Bei der Minimierung führt man dann Schritte gemäß

$$\begin{pmatrix} \mathbf{x}_k \\ t_k \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{k-1} \\ t_{k-1} \end{pmatrix} - \alpha \cdot C \nabla_{\mathbf{x},t}\tilde{u}(\mathbf{x}_{k-1}, t_{k-1}) \quad (4.19)$$

aus, wobei α die Schrittweite angibt und C eine $(N+1) \times (N+1)$ -Matrix ist, die die verschiedenen Skalen der räumlichen und zeitlichen Richtung ausgleicht. Eine interessante Möglichkeit dieses erweiterten Verfahrens sollte sein, daß es anzeigt, wenn eine Trajektorie bei $t \neq 0$ endet. Die Endpunkte der Trajektorien sind nämlich durch lokale Minima auf \tilde{u} bei $t \neq 0$ gegeben. Desweiteren sollte die erweiterte Suchrichtung durch Hinzunahme der Zeit in die Gradientenbildung eine eindeutigere Verfolgung einer Trajektorie ermöglichen.

Die Hoffnung war also, daß diese Prozedur bei entsprechender Wahl von C und genügend kleiner Schrittweite α eine Verbesserung der Ergebnisse der DEM liefern würde. Aber die Probleme blieben gegenüber dem ursprünglichen Verfahren dieselben. Man kann zwar für bestimmte Wahlen der Parameter, eindeutige Ergebnisse für bestimmte Cluster-Größen erreichen oder das Ende einer Trajektorie bei $t \neq 0$ bestätigen, aber eine allgemein funktionierende Wahl der Parameter ist auch mit dieser Änderung unmöglich. Das Problem

ist, daß die einfache Ableitung nach t keine gute Schrittweitenbestimmung für diesen Parameter darstellt, denn $\frac{\partial}{\partial t}\tilde{u}$ hängt stark nicht-linear von t ab, weswegen man immernoch gezwungen ist, eine Vorschrift in die Matrix C zu integrieren, die in Abhängigkeit von t die zeitliche Komponente des Gradienten sinnvoll anpaßt. Damit ist man aber wieder an dem Punkt, den man beim DEM Verfahren eigentlich verbessern wollte, so daß sich die Idee der erweiterten Gradientenbildung nicht bewährt hat.

Das Defizit der fehlenden Bewegungsgleichung wird allerdings durch die Annealing Algorithmen beseitigt, die im nächsten Abschnitt getestet werden sollen.

4.3 Anwendung des aGDA und des QMA

Die Verfahren des adiabatic Gaussian Density Annealing und des Quantum Mechanical Annealing haben zwar einen durchaus unterschiedlichen Ursprung, aber dennoch resultieren beide in Bewegungsgleichungen, die sich nur durch konstante Faktoren unterscheiden. Deswegen unterscheiden sich auch die Algorithmen nur wenig und man kann die Anwendung der beiden Methoden gut zusammen betrachten.

Auch der Vergleich mit der DEM hat gezeigt, daß sich die Verfahren sehr ähnlich sind, so daß man einige Überlegungen für die Anwendung auf das Lennard-Jones-Problem übernehmen kann. Man betrachtet wieder ein Cluster von Atomen, zwischen denen ein Potential der Form 4.5 wirkt. Beim aGDA benötigt man nun die Funktion $\langle U \rangle$ aus Gl. 2.47. Der Unterschied zur Lösung 2.11 bei der DEM besteht nur darin, daß die Breiten der Einteilchen-Gauß-Funktionen nun unterschiedlich sind, während man bei der DEM nur mit einer einzigen Breite t für alle Atome rechnet.

Um das Integral analytisch ausrechnen zu können, wird das Lennard-Jones-Potential wieder durch eine Summe von Gauß-Funktionen angenähert. Dann ergibt sich:

$$\begin{aligned}
\langle U \rangle &= \det^{-\frac{1}{2}}(2\pi\Lambda^2) \int \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^L a_k e^{-b_k(\bar{x}_i - \bar{x}_j)^2} e^{-\frac{1}{2}[\mathbf{x} - \mathbf{x}^0(\beta)]\Lambda^{-1}(\beta)[\mathbf{x} - \mathbf{x}^0(\beta)]} d\mathbf{x} \\
&= \left(\prod_{p=1}^N (2\pi\sigma_p)^{-\frac{3}{2}} \right) \int \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^L a_k e^{-b_k(\bar{x}_i - \bar{x}_j)^2} e^{-\sum_{s=1}^N \frac{(\bar{x}_s - \bar{x}_s^0)^2}{2\sigma_s}} dx_1^3 dx_2^3 \dots dx_N^3 \\
&= \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^L a_k (1 + 2b_k(\sigma_i + \sigma_j))^{-\frac{3}{2}} e^{-\frac{b_k(x_i^0 - x_j^0)^2}{1 + 2b_k(\sigma_i + \sigma_j)}} \quad (4.20)
\end{aligned}$$

Als Ergebnis ergibt sich also wieder 4.9 mit der Ersetzung $t = \frac{1}{4}(\sigma_i + \sigma_j)$. Die Parameter a_k und b_k werden aus Kapitel 4.2 übernommen. Allerdings ist für die hier betrachteten Annealing Algorithmen die Erweiterung der Gauß-Summe um den dritten Summanden nicht geeignet, um zu verhindern, daß die Atome sich ganz vom Cluster lösen. Denn übernimmt man auch diesen Summanden, so beeinflußt man damit die Krümmung der gemittelten Funktion $\langle U \rangle$, was beim DEM Verfahren keinen Einfluß hat, aber hier durchaus von Bedeutung ist, da die Breite der Gaußverteilung gemäß der Krümmung angepaßt wird. Man kann aber eine andere einfache Funktion verwenden, um das Cluster zusammenzuhalten,

die wie folgt definiert wird:

$$U^{confine} = \gamma \sum_{i=1}^N \sum_{j=i+1}^N r_{ij}^2 \quad \text{mit} \quad r_{ij} = |\vec{x}_i - \vec{x}_j| \quad (4.21)$$

Ein Wert von $\gamma = 5 \cdot 10^{-3}$ ist ausreichend, um die Atome am Cluster zu halten. Da die zweite Ableitung von $U^{confine}$ konstant und sehr klein ist, beeinflusst sie die Berechnung von Λ nur in den Bereichen, wo der Atomabstand sehr groß und somit das Lennard-Jones Potential sehr klein ist.

# Atome	U_{min}^{GDA}	global?	U_{min}^{QMA}	global?
5	-9.1039	+	-9.1039	+
6	-12.7121	+	-12.7121	+
7	-16.5054	+	-16.5054	+
8	-19.7653	-	-19.8215	+
9	-23.2698	-	-24.1134	+
10	-28.4225	+	-28.4225	+
11	-32.7660	+	-32.7660	+
12	-37.9676	+	-37.9676	+
13	-44.3268	+	-44.3268	+
14	-47.8452	+	-47.8452	+
15	-50.5336	-	-51.4180	-
16	-56.8157	+	-55.5593	-
17	-61.3071	-	-58.1726	-
18	-66.5309	+	-66.5309	+
19	-72.6598	+	-72.6598	+
20	-77.1770	+	-77.1770	+
21	-81.6846	+	-81.6846	+
22	-86.8098	+	-86.8098	+
23	-92.8445	+	-91.3278	-
24	-97.3488	+	-97.3488	+
25	-102.3727	+	-102.3727	+
26	-108.3156	+	-108.3156	+
27	-112.0829	-	-112.8255	-
28	-116.4915	-	-117.7780	-
29	-123.5874	+	-122.2913	-
30	-128.1816	-	-122.8137	-

TABELLE 4.5: Ergebnisse für aGDA und QMA

Mit dieser Vorbereitung können die Algorithmen des aGDA und QMA bereits implementiert werden. Da der aGDA Algorithmus aber einen Teil des QMA Algorithmus darstellt, nämlich den für $\hbar = 0$, kann man beide Methoden in einem Code implementieren. Als lokales Minimierungsverfahren wurde hier ein einfacher Steepest-Descent-Algorithmus verwendet. Tabelle 4.3 zeigt die Ergebnisse für Lennard-Jones-Cluster mit bis zu 30 Atomen. Für beide Algorithmen wurden für jede Clustergröße jeweils 100 Rechnungen durchgeführt, da die Ergebnisse von der Startkonfiguration abhängen, denn auch beim QMA

aGDA	QMA
$(\Lambda_0)_{ii} = 4.0$	$(\Lambda_0)_{ii} = 4.0$
$\lambda_{max} = 1.0 \cdot 10^{-4}$	$\hbar_{max} = 2.0$
	$\hbar_{min} = 1.0 \cdot 10^{-4}$
$\Delta\beta = 0.1$	$\Delta\hbar = 0.1\hbar_{k-1}$

TABELLE 4.6: Parameter für aGDA und QMA

wurde \hbar_{max} nicht so groß gewählt, daß $\langle U \rangle(\hbar_{max})$ konvex ist. Der Grund dafür ist das meist schlechtere Ergebnis, wenn man \hbar_{max} wesentlich größer wählt. Der tiefste Potentialwert aller gefundenen Konfigurationen wurde in die Tabelle eingetragen. Die sonstigen Parameter kann man Tabelle 4.3 entnehmen.

Die Verfahren zeigen beide durchaus gute Ergebnisse für kleine Cluster mit bis zu 26 Atomen. Der QMA Algorithmus ist, obwohl \hbar_{max} relativ klein gewählt wurde, wesentlich sicherer in seinem Ergebnis. Das bedeutet, die 100 Anwendungen auf die gleiche Clustergröße resultierten in höchstens drei verschiedenen Minimums-Konfigurationen, während es beim aGDA oft bis zu zehn verschiedene waren. Für größere Cluster liegen die gefundenen Minima zwar immernoch sehr tief, aber die globalen Minima werden von beiden Verfahren nicht mehr gefunden. Hier macht sich wieder die einfache Näherung durch eine Gauß-Verteilung bemerkbar, die nicht ausreicht, um die Wahrscheinlichkeitsverteilung etwas komplizierterer Cluster ausreichend darzustellen. Abhilfe würde hier mit Sicherheit eine bessere Näherung schaffen, die vor allem bei verschiedenen Temperaturen auch unterschiedliche Anzahlen von Gebieten mit großer Wahrscheinlichkeit darstellen können sollte. Shalloways [5] Ansatz mit verschiedenen Wave-Packets ist daher sehr vielversprechend, aber leider auch um vieles schwieriger zu realisieren, so daß selbst seine Anwendung auf Lennard-Jones Cluster bis jetzt nicht vollkommen realisiert werden konnte.

4.4 Minimierung von Lennard-Jones-Clustern mittels des EDP

Bei der Minimierung mittels des Effective Diffused Potentials wird mit einer Näherung an die freie Energie gerechnet und deren Minimum bei einer absteigenden Folge von Temperaturen verfolgt. Bei $T = 0$ stimmen freie und innere Energie überein, wodurch man das Ergebnis gut mit denen anderer Verfahren vergleichen kann. Man muß jedoch bedenken, daß bei endlichen Temperaturen nicht unbedingt die Konfiguration die wahrscheinlichste ist, die die geringste innere Energie hat, da durch den Einfluß der Entropie breite Minima gegenüber schmalen bevorzugt werden. Findet der Algorithmus also nicht das gleiche Minimum wie andere Verfahren, die nur mit der inneren Energie rechnen, so kann das bedeuten, daß das globale Minimum der Potentialfunktion bei endlichen Temperaturen kein Minimum der freien Energie ist. Das ist aber keineswegs ein Nachteil dieses Verfahrens, da alle physikalischen Systeme ebenfalls mit der größten Wahrscheinlichkeit die Konfiguration einnehmen, welche die geringste freie Energie hat.

Aber obwohl man nun mit einer Näherung an die freie Energie rechnet, ergeben sich nach der Implementierung der anderen Verfahren nur geringe Änderungen bzw. Erweite-

rungen der Algorithmen. Berechnet werden soll die freie Energie aus 2.86

$$F(\bar{\mathbf{x}}, \Lambda, T) = \det^{-\frac{1}{2}}(2\pi\Lambda) \int U(\mathbf{x}) e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})\Lambda^{-1}(\mathbf{x}-\bar{\mathbf{x}})} d\mathbf{x} - \frac{kT}{2} \left[3 \sum_{i=1}^N \ln(2\pi\sigma_i) + 3N \right] \quad (4.22)$$

Die Lösung des Integrals für das Lennard-Jones Potential ist schon aus Kap.4.3 bekannt. Es wird wieder die Näherung des Potentials durch zwei Gauß-Funktionen benutzt. Der Entropie-Anteil aus 4.22 kann direkt ausgewertet werden, so daß sich insgesamt folgendes ergibt:

$$F_{LJ}(\bar{\mathbf{x}}, \Lambda, T) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^2 \tilde{a}_k e^{-\tilde{b}_k(\bar{x}_i - \bar{x}_j)^2} - \frac{kT}{2} \left(3 \sum_{i=1}^N \ln(2\pi\sigma_i) + 3N \right) \quad (4.23)$$

mit

$$\tilde{a}_k = a_k (1 + 2b_k(\sigma_i + \sigma_j))^{-\frac{3}{2}} \quad (4.24)$$

$$\tilde{b}_k = b_k (1 + 2b_k(\sigma_i + \sigma_j))^{-1} \quad (4.25)$$

$$\bar{\mathbf{x}} = \begin{pmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_N \end{pmatrix} \quad (4.26)$$

Um das Cluster bei hohen Temperaturen trotzdem gebunden zu halten, ist wieder eine zusätzliches Potential nötig, welches das Cluster zu kleineren Atomabständen zwingt. Es wird wieder das einfache harmonische Potential

$$U^{confine} = \gamma \sum_{i=1}^N \sum_{j=i+1}^N |\bar{x}_i - \bar{x}_j|^2 \quad (4.27)$$

benutzt. Die Konstante γ muß groß genug gewählt werden, um das Lösen der Atome vom Cluster zu verhindern, und sie muß klein genug sein, um die Potentialstruktur nicht signifikant zu verändern. In allen Rechnungen erwies sich $\gamma = 0.02$ als eine gute Wahl. Der Beitrag dieses Potentials zur internen Energie U_{eff} berechnet sich zu

$$\det^{-\frac{1}{2}}(2\pi\Lambda) \int U^{confine}(\mathbf{x}) e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})\Lambda^{-1}(\mathbf{x}-\bar{\mathbf{x}})} d\mathbf{x} = \gamma \sum_{i=1}^N \sum_{j=i+1}^N |\bar{x}_i - \bar{x}_j|^2 + 3\gamma(N-1) \sum_{i=1}^N \sigma_i \quad (4.28)$$

Insgesamt ergibt sich also folgende Formel für die freie Energie eines Lennard-Jones-Clusters:

$$F_{LJ}(\bar{\mathbf{x}}, \Lambda, T) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^2 \tilde{a}_k e^{-\tilde{b}_k(\bar{x}_i - \bar{x}_j)^2} - \frac{kT}{2} \left(3 \sum_{i=1}^N \ln(2\pi\sigma_i) + 3N \right) \quad (4.29)$$

$$+ \gamma \sum_{i=1}^N \sum_{j=i+1}^N |\bar{x}_i - \bar{x}_j|^2 + 3\gamma(N-1) \sum_{i=1}^N \sigma_i \quad (4.30)$$

Damit kann der EDP-Algorithmus implementiert werden. Als lokales Minimierungsverfahren diene hier wieder ein einfacher Steepest-Descent-Algorithmus. Tabelle 4.4 zeigt die

Ergebnisse für Cluster mit bis zu 40 Atomen. Alle Rechnungen wurden bei einer Temperatur von entsprechend $kT_{max} = 3.5$ gestartet. Das geschah in Anlehnung an die Literatur [14], trotzdem konnten nicht alle Ergebnisse reproduziert werden. Das liegt daran, daß bei dieser Temperatur noch mehrere Minima der freien Energie vorhanden sind und das Ergebnis daher von der Startkonfiguration abhängt. Die Ergebnisse sind also keineswegs die besten, die man durch eine große Anzahl von Rechnungen erreichen könnte, zeigen aber, daß der Algorithmus stets tief liegende Minima findet.

# Atome	U_{global}	U_{min}	global?	$Min(U_{min})$	global?
7	-16.5054	-16.5054	+	-16.5054	+
8	-19.8215	-19.7653	-	-19.7653	-
9	-24.1134	-23.2698	-	-23.2698	-
10	-28.4225	-28.4225	+	-28.4225	+
11	-32.7660	-32.7660	+	-32.7660	+
12	-37.9676	-37.9676	+	-37.9676	+
13	-44.3268	-44.3268	+	-44.3268	+
14	-47.8452	-47.8452	+	-47.8452	+
15	-52.3226	-50.1713	-	-52.3226	+
16	-56.8157	-55.7783	-	-56.8157	+
17	-61.3180	-59.4303	-	-61.3180	+
18	-66.5309	-64.1350	-	-66.5309	+
19	-72.6598	-69.5416	-	-72.6598	+
20	-77.1770	-75.5892	-	-77.1770	+
21	-81.6846	-77.8357	-	-81.6846	+
22	-86.8098	-86.8098	+	-86.8098	+
23	-92.8445	-92.8445	+	-92.8445	+
24	-97.3488	-93.2811	-	-97.3488	+
25	-102.3727	-100.2629	-	-102.3727	+
26	-108.3156	-101.9846	-	-108.3156	+
27	-112.8735	-112.8255	-	-112.8255	-
28	-117.8224	-117.3446	-	-117.8224	+
29	-123.5874	-119.2993	-	-123.5874	+
30	-128.2866	-126.8746	-	-128.1816	-
31	-133.5864	-131.7206	-	-132.2184	-
32	-139.6355	-137.2454	-	-135.9581	-
33	-144.8427	-141.6785	-	-142.4784	-
34	-150.0445	-147.5889	-	-149.6721	-
35	-155.7566	-153.2152	-	-154.5210	-
36	-161.8254	-159.0590	-	-160.5094	-
37	-167.0337	-163.9518	-	-166.8226	-
38	-173.9284	-168.8194	-	-170.2837	-
39	-180.0332	-174.7076	-	-175.4776	-
40	-185.2498	-176.5944	-	-181.1579	-

TABELLE 4.7: Ergebnisse des EDP-Algorithmus

Wählt man die maximale Temperatur etwa zehn bis 20 mal höher, so existiert nur

noch ein einziges Minimum, so daß von jeder Startkonfiguration aus das gleiche Ergebnis berechnet wird. Allerdings verschlechtern sich die Ergebnisse dadurch gegenüber denen aus Tabelle 4.4 und der Algorithmus wird zusätzlich wesentlich langsamer, da die Berechnung der Minima bei hohen Temperaturen wesentlich aufwendiger ist.

Um eine Verbesserung des Verfahrens erreichen zu können, muß man sich zunächst die Ursache für die schlechten Ergebnisse klarmachen. Wie auch schon bei den meisten anderen Verfahren des Simulated Annealing in Temperature rechnet man mit einer sehr einfachen Näherung an die Wahrscheinlichkeitsverteilung des kanonischen Ensembles. Sie berücksichtigt nicht die Eigenschaft dieser Verteilung, bei kleinen Temperaturen in viele voneinander getrennte Gebiete großer Wahrscheinlichkeit unterteilt zu sein. Die verwendete Näherung funktioniert nur bei großen Temperaturen und bei $T = 0$ gut, dazwischen ist sie nicht ausreichend. Das wäre für den Zweck des Findens des globalen Energieminimums unwichtig, wenn die gute Näherung bei hoher Temperatur in eine gute Näherung bei $T = 0$ übergehen würde, aber das tut sie nicht in allen Fällen. Das liegt daran, daß der Algorithmus versucht, sich stetig von einem Minimum bei hoher Temperatur bis zu einem bei $T = 0$ zu bewegen. Entsteht aber unterhalb einer gewissen Temperatur ein neues Minimum, welches sich im Konfigurationsraum weit entfernt befindet, so kann der Algorithmus dies nicht berücksichtigen. Wollte man das mit einfachen Mitteln dem Verfahren hinzufügen, so könnte man bei jeder Temperatur den Konfigurationsraum nach neuen Minima absuchen, was aber einen nicht vertretbaren Aufwand bedeuten würde. Man kann jedoch einen Kompromiß eingehen, indem man eine Temperatur auswählt, bei der man den Konfigurationsraum genügend nach allen vorhandenen Minima durchsucht. Jedes dieser Minima nimmt man dann als Startkonfiguration für den EDP-Algorithmus. Wählt man dabei allerdings die Temperatur, bei der man die Suche durchführt, zu groß, so wird sich das Ergebnis kaum von dem unterscheiden, welches man bei dem ursprünglichen EDP-Algorithmus bei hohen Temperaturen erreicht. Wählt man die Temperatur hingegen zu klein, so wird der Aufwand für das Suchen und die Durchführung des Algorithmus für jedes gefundene Minimum zu groß. Es gilt also auch hier einen Kompromiß zu finden zwischen Genauigkeit und Aufwand.

Um den Algorithmus schneller zu machen, kann man zusätzlich noch den Produktsatz der Einteilchen-Gauß Funktionen so vereinfachen, daß für alle Teilchen

$$\sigma_i = \sigma \quad i = 1, \dots, N \quad (4.31)$$

gilt. Dann vereinfacht sich Gl.4.29 zu

$$F_{LJ}(\bar{\mathbf{x}}, \sigma, T) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{k=1}^2 \tilde{a}_k e^{-\tilde{b}_k (\bar{x}_i - \bar{x}_j)^2} - \frac{3NkT}{2} (\ln(2\pi\sigma) + 1) \quad (4.32)$$

$$+ \gamma \sum_{i=1}^N \sum_{j=i+1}^N |\bar{x}_i - \bar{x}_j|^2 + 3\gamma\sigma N(N-1) \quad (4.33)$$

Die Starttemperatur, bei der der gesamte Konfigurationsraum nach Minima durchsucht wird, muß der Zahl der Atome angepaßt werden. Für kleine N muß auch die Temperatur kleiner gewählt werden, um genügend Minima finden zu können. Tabelle 4.8 zeigt die benutzten Starttemperaturen in Abhängigkeit von der Zahl der Atome N . Bei der Suche wurde jeweils eine zufällige Startkonfiguration gewählt und dann lokal minimiert. Die

Minima wurden nur anhand der Werte ihrer freien Energie unterschieden. So wurden 50 Startkonfigurationen minimiert und die zehn mit den niedrigsten freien Energien als Startwerte für den EDP-Algorithmus verwendet. Tabelle 4.4 enthält in der fünften Spalte die Ergebnisse, die so erzielt wurden.

Atome	kT_{max}
7 – 17	0.5
18 – 24	3.5
25 – 29	5.0
30 – 40	10.0

TABELLE 4.8: Such-Temperaturen für den erweiterten EDP-Algorithmus

Wie man sieht, wurden die Ergebnisse durch die kleine Erweiterung erheblich verbessert. Die Schwierigkeit dabei bleibt jedoch, die richtige Temperatur zu wählen, bei der ein zu einem tiefen Minimum bei $T = 0$ korrespondierendes Minimum noch existiert. Man hat nur die Möglichkeit, verschiedenen Temperaturen für eine bestimmte Anzahl von Atomen auszuprobieren und sie dann auch für Systeme mit ähnlich vielen Atomen zu benutzen. Wie Tabelle 4.8 aber zeigt, ist der Bereich ziemlich klein, bei dem man den gleichen Wert von kT_{max} benutzen kann. Deshalb sind dem Verfahren auch mit dieser Erweiterung Grenzen gesetzt. Die globalen Minima der Cluster mit mehr als 30 Atomen sind für Verfahren mit Gauß-Näherungen an die Wahrscheinlichkeitsverteilung nur in besonders einfachen Ausnahmefällen zu finden. Generell wird die Struktur der Potentialfläche für die Cluster solcher Größen zu kompliziert, als daß eine so einfache Näherung funktionieren könnte. Eine Voraussetzung für das Funktionieren der Ansätze ist zum Beispiel, daß die Hierarchie aller Minima erhalten bleibt, also daß bei jeder Temperatur das gleiche Minimum das tiefste ist. Bei der Minimierung der verschiedenen Startkonfigurationen, die durch die Suche bei endlicher Temperatur gefunden wurden, sieht man deutlich die Verletzung dieser Hierarchie.

Wie auch noch genauer in Kapitel 4.6 beschrieben werden wird, existieren für die Minima von Lennard-Jones Clustern verschiedene dominierende Strukturen wie Icosaeder, fcc oder Decaeder (siehe [8]). Schaut man sich dann zum Beispiel ein Cluster mit einer bestimmten Anzahl von Atomen an, so gibt es lokale Minima mit niedrigen Potentialwerten für jede dieser Strukturen. Die zugehörigen Konfigurationen liegen auf der Potentialfläche aber räumlich getrennt, so daß nur eines davon - das breiteste - durch die genäherte Wahrscheinlichkeitsverteilung gut wiedergegeben wird. Ist das breiteste nicht gleichzeitig auch das tiefste Minimum, so hat der Algorithmus kaum eine Chance, das globale Minimum zu finden. Allerdings rechnet er mit einer Näherung an die freie Energie, so daß man nicht unbedingt erwarten kann, daß er das globale Potential-Minimum findet, wenn es bei endlichen Temperaturen durch den Entropieanteil gegenüber einem anderen benachteiligt ist. Es ist aber sehr schwierig, eine Aussage darüber zu machen, warum der Algorithmus ein bestimmtes Minimum bevorzugt, denn er minimiert nicht die reine potentielle Energie, benutzt aber auch nur eine Näherung der freien Energie. Trotzdem lokalisiert der EDP-Algorithmus die meisten globalen Minima mit bis zu 30 Atomen.

4.5 Minimierung mit dem Basin-Hopping-Algorithmus

Wales and Doye [8] haben mit dem Basin-Hopping-Algorithmus alle globalen Minima für Lennard-Jones-Cluster mit bis zu 110 Atomen gefunden. Wie schon beschrieben, ist er besonders leicht für alle Arten von Potentialen zu implementieren. Eine große Schwachstelle des Algorithmus stellt aber bei großen Zahlen von Atomen die lange Rechenzeit dar. Sie ist bedingt durch die häufige Benutzung eines Minimierungsalgorithmus, dessen Aufwand proportional zu N^2 (N ist die Zahl der Atome) steigt, wenn bei der Berechnung des Potentials und des Gradienten alle Paare von Atomen berücksichtigt werden. Besteht das Potential wie hier ausschließlich aus Lennard-Jones-Wechselwirkungen, kann man jedoch eine Technik anwenden, die vor allem in der Molekül-Dynamik benutzt wird, um den Aufwand drastisch zu reduzieren. Diese Methode ist also keineswegs neu, wird aber im Zusammenhang mit globalen Minimierungsverfahren nie erwähnt. Es handelt sich bei der Methode um die Einteilung des Simulationsgebietes in die sogenannte „linked-cell“ Struktur. Die Idee ist, daß man kurzreichweitige Potentiale nur auswerten muß, wenn der Abstand zweier Atome einen gewissen Wert nicht überschreitet. Der Fehler, den man dabei macht, ist nur äußerst gering, aber der Aufwand reduziert sich auf eine lineare Abhängigkeit von N . Das Lennard-Jones-Potential hat sein Minimum bei $r_{min} = 2^{\frac{1}{6}}\sigma$ und geht für größere r sehr schnell gegen null. Bei $r = 2.5$ hat dieses Potential für $\epsilon = \sigma = 1$ einen Wert von $U^{LJ}(r = 2.5) \approx -1.6 \cdot 10^{-2}$, bei $r = 5.0$ gilt nur noch $U^{LJ}(r = 5.0) \approx -2.5 \cdot 10^{-4}$. Das bedeutet, man kann in einem Bereich zwischen diesen beiden Werten das Potential abschneiden, ohne einen großen Fehler zu machen. Dazu geht man allgemein so vor:

Man wählt einen Radius r_{cut} , der den maximalen Abstand zweier Partikel angibt, zwischen denen das Potential noch berechnet werden soll. Dann teilt man das Simulationsgebiet in Würfel mit der Kantenlänge r_{cut} ein. Jeder dieser Würfel ist eine Zelle, der die Partikel zugeordnet werden, die in ihr liegen. Will man nun das Potential und dessen Ableitung berechnen, so braucht man nur Paare von Partikeln zu berücksichtigen, die entweder in der gleichen Zelle oder in benachbarten Zellen liegen. Man durchläuft also jede Zelle und berechnet das Potential zwischen den Partikeln in dieser und den Nachbarzellen. So ergibt sich ein sehr effizienter Algorithmus für die Potential- und Gradienten-Berechnung. Es bleibt noch zu überlegen, welchen Einfluß diese Veränderung des Potentials auf die Mi-

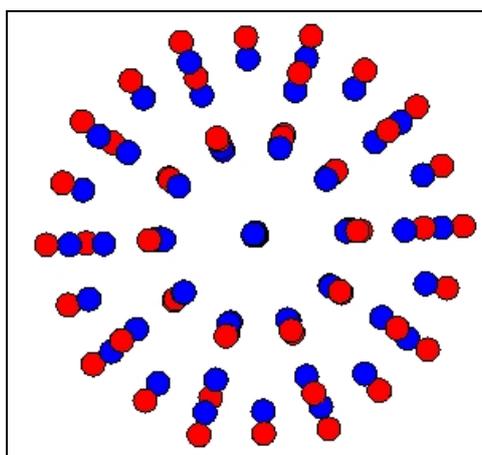


ABBILDUNG 4.6: Cluster mit (•) und ohne (•) r_{cut}

nima haben kann. Natürlich ändern sich die Werte und die Konfigurationen der Minima, aber durch Anwendung eines Minimierungsverfahrens, welches wieder alle $\frac{N(N-1)}{2}$ Partikelpaare berücksichtigt, wird jedes Minimum des abgeschnittenen Potentials in eines des originalen Potentials überführt. Dabei ändert sich der Energiewert in Abhängigkeit von der Gesamtzahl der Partikel durchaus deutlich, aber die Konfiguration bleibt bis auf eine durch die langreichweitigen Anziehungen bewirkte Veränderung der Dichte des Clusters die gleiche. Das läßt sich auch gut an folgendem Beispiel erkennen: Abbildung 4.6 zeigt das globale Minimum für 55 Atome, einmal mit dem linked-cell-Code minimiert und einmal mit einem N^2 -Code. Zum besseren Vergleich wurden die beiden Bilder übereinander gelegt. Man sieht, daß sich die Konfiguration des Clusters bis auf seine Dichte nicht verändert hat, wobei sich der Potentialwert deutlich von $U_{r_{cut}}^{LJ} = -277.2406$ auf $U_{min}^{LJ} = -279.2485$ verringert hat.

Mit diesen Veränderungen, die nur die Berechnung des Potentials und deren Ableitung betreffen, kann man den Basin-Hopping-Algorithmus sehr effizient implementieren. Als lokales Minimierungsverfahren wurde hier das CG-Verfahren benutzt. Für die Rechnungen wurden die Parameter gemäß Tabelle 4.9 gesetzt.

Parameter	Wert
k_{max}	15000
β	1.25
c	0.7
r_{cut}	2.5

TABELLE 4.9: Parameter für den Basin-Hopping Algorithmus

Zum Vergleich der Rechenzeiten des ursprünglichen Verfahrens mit denen des um die linked-cell-Struktur erweiterten sind noch jeweils drei Rechnungen mit unterschiedlichen Anzahlen von Atomen durchgeführt worden, wobei die Anzahl der Gesamtschritte auf $k_{max} = 100$ reduziert wurde. Wie in Tabelle 4.10 ersichtlich, ist Reduktion der Rechenzeit enorm. Während beim N^2 -Algorithmus die Rechenzeit offensichtlich von der Potentialauswertung dominiert wird, hängt die Laufzeit beim linked-cell-Code eher von der speziellen Form des Potentials des Clusters ab, so daß hier das Minimierungsverfahren im Durchschnitt auch bei kleineren Clustern länger brauchen kann als bei größeren.

# Atome	N^2	linked-cell
15	22.73s	24.37s
30	173.66s	21.48s
60	1021.76s	35.11s

TABELLE 4.10: Vergleich der Rechenzeiten

Mit dem so implementierten Algorithmus wurde für einen weiten Bereich von verschiedenen großen Lennard-Jones-Clustern die Minima berechnet. Die aus der Literatur (siehe [8] und [10]) bekannten globalen Minima für Cluster mit bis zu 60 Atomen wurden so gefunden. Für die Minimierung größerer Cluster wurde als Startkonfiguration jeweils das gefundene Minimum des um ein Atom kleineren Clusters genommen und ein Atom willkürlich

am Rand hinzugefügt. So wurden Minima für bis zu 160 Atomen berechnet. Die Zahlen für die Cluster bis 150 Atome sind hier nicht mehr aufgeführt, man kann sie in [10] nachsehen. Tabelle 4.11 enthält die Ergebnisse für die größeren Cluster, die nicht mehr mit Literaturwerten verglichen werden können, und daher zunächst nur die bis jetzt tiefsten erreichten Potentialwerte darstellen. Da der zeitliche Aufwand für den Algorithmus auch

# Atome	U_{min}
151	-899.1810
152	-905.4296
153	-911.7074
154	-917.8095
155	-923.9561
156	-931.3076
157	-936.9419
158	-943.4663
159	-950.0260
160	-957.1062
309	-2007.2190

TABELLE 4.11: *Potentialwerte neuer Minima*

für sehr große Cluster noch nicht zu groß ist, kann man auch noch Rechnungen mit mehr als 160 Atomen durchführen. Dann benötigt man im Allgemeinen jedoch auch eine wesentlich größere Anzahl von Monte-Carlo Schritten, um zu einem guten Ergebnis zu gelangen. Hier wurden deshalb für das Lennard-Jones-Cluster mit 309 Atomen, welches im Minimum ein vollständiges Mackay-Icosaeder bildet (siehe Kap.4.6), 100000 Schritte durchgeführt. Als Ergebnis wurde die richtige Konfiguration mit dem in Tabelle 4.11 angegebenen Potentialwert gefunden. Das zeigt, wie gut der Algorithmus auch noch bei größeren Atomzahlen funktioniert.

Der Basin-Hopping Algorithmus ist das einzige Verfahren, das sämtliche globalen Minima für Lennard-Jones-Cluster mit bis zu 110 Atomen gefunden hat, ohne Kenntnisse über die Struktur des Potentials oder vorher gefundene Minima zu benutzen. Außerdem ist er leicht zu implementieren, da man nur ein beliebiges lokales Minimierungsverfahren benötigt und die vereinfachte Potentialfläche mit einem Monte-Carlo-Verfahren absucht. Daher kann diese Methode für jede Anwendung der Minimierung angepaßt werden und wird auch im nächsten Kapitel auf das TIP4P-Potential angewendet.

4.6 Bilder einiger Minima

Im Folgenden sind einige Bilder der Minima von Lennard-Jones-Cluster zu sehen. Alle Cluster mit mehr als 25 Atomen wurden mit der oben beschriebenen Erweiterung des Basin-Hopping-Algorithmus berechnet. Kleinere Cluster resultieren aus Berechnungen mit dem DEM Algorithmus.

Besonders stabile und damit auch relativ einfach zu findende Strukturen entstehen bei den sogenannten magischen Zahlen für Icosaeder. Die Zahl der Atome hängt von der

Anzahl der Schalen N_s ab und läßt sich mit folgender Formel berechnen:

$$N = \frac{1}{3}(10N_s^3 + 15N_s^2 + 11N_s + 3) \quad (4.34)$$

Das ergibt die magischen Zahlen $N_m = 1, 13, 55, 147, 306, \dots$. Cluster mit diesen Größen bilden in ihrem globalen Energieminimum die sogenannten Mackay-Icosaeder (siehe Abbildung 4.7 bis 4.12). Diese Konfiguration für 309 Atome wurden mit der oben beschriebenen Variante des Basin-Hopping nach Durchführung von 100000 Monte-Carlo Schritten gefunden.

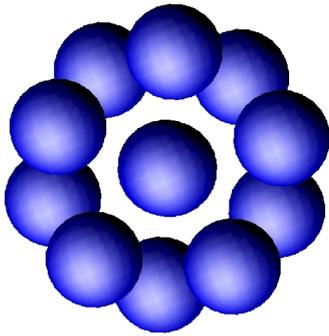


ABBILDUNG 4.7: *Mackay-Icosaeder mit 13 Atomen*

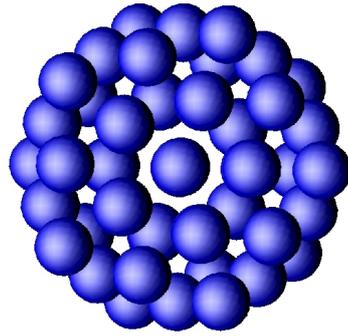


ABBILDUNG 4.8: *Mackay-Icosaeder mit 55 Atomen*

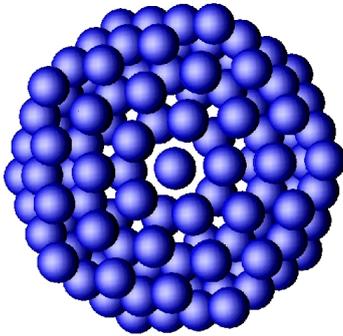


ABBILDUNG 4.9: *Mackay-Icosaeder mit 147 Atomen*

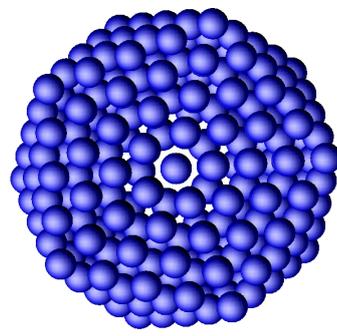


ABBILDUNG 4.10: *Mackay-Icosaeder mit 309 Atomen*

Das DEM Verfahren nach Scheraga [6] findet für das 12 Atom Cluster die Konfiguration 4.13 mit einer Energie von $U^{12} = -33.5975$, welches die Struktur des 13 Atom Mackay-Icosaeder ohne das Zentralatom darstellt. Das tatsächliche Minimum ist aber die Konfiguration 4.14 mit der Energie $U_{min}^{12} = -37.9676$. Hier fehlt dem vollständigen Mackay-Icosaeder ein Atom in der ersten Schale. Das läßt darauf schließen, daß bei der DEM durch die Benutzung der speziellen Startbedingungen symmetrische Konfigurationen bevorzugt werden.

Die Strukturen der meisten Minima basieren auf dem Icosaeder, aber zum Beispiel für $N = 38$ entsteht eine vollkommen andere Struktur. Abbildung 4.15 zeigt das Minimum mit $U_{min}^{38} = -173.9284$, welches ein fcc-basiertes abgeschnittenes Oktaeder darstellt (siehe

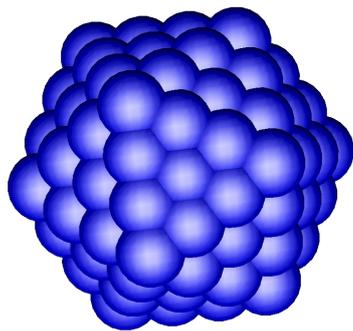


ABBILDUNG 4.11: *Mackay-Icosaeder mit 147 Atomen (andere Darstellung)*

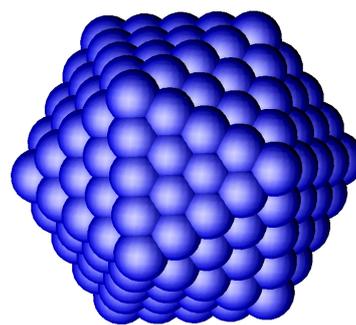


ABBILDUNG 4.12: *Mackay-Icosaeder mit 309 Atomen (andere Darstellung)*

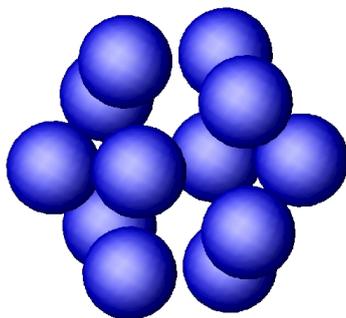


ABBILDUNG 4.13: *Minimum mit 12 Atomen*

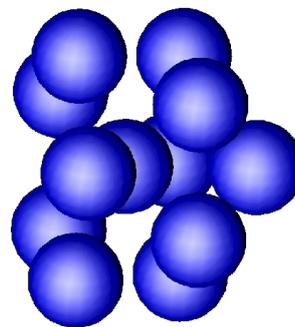


ABBILDUNG 4.14: *Globales Minimum mit 12 Atomen*

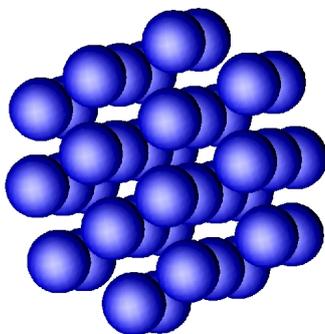


ABBILDUNG 4.15: *Globales Minimum mit 38 Atomen*

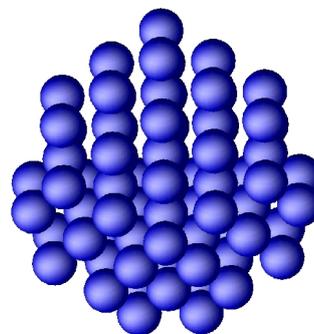
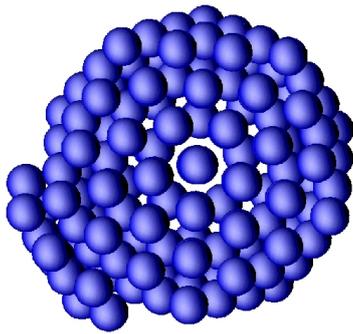
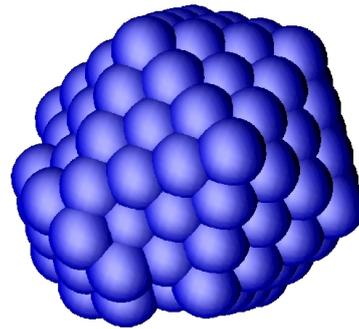


ABBILDUNG 4.16: *Globales Minimum mit 77 Atomen*

[8]). Für $N = 75, 76, 77, 102, 103, 104$ basieren die Strukturen der globalen Minima auf einem Decaeder. Abbildung 4.16 zeigt ein Beispiel für eine solche Konfiguration.

Die gefundenen Minima für Cluster mit mehr als 150 Atomen aus Kapitel 4.5 basieren alle auf der Icosaeder-Struktur. Ihre Konfiguration entspricht der des vollständigen 147-Atom-Icosaeder, wobei sich die restlichen Atome in der unvollständigen vierten Schale befinden. Als Beispiel ist in Abbildung 4.17 und 4.18 das Cluster mit 160 Atomen in zwei verschiedenen Darstellungen abgebildet.

ABBILDUNG 4.17: *Minimum mit 160 Atomen*ABBILDUNG 4.18: *Minimum mit 160 Atomen*

4.7 Zusammenfassung

In diesem Kapitel wurden die meisten der in Kapitel 2 vorgestellten Verfahren auf einfache atomare Cluster mit Lennard-Jones Potential angewendet. Obwohl solche Cluster, die nur aus einer Atomsorte bestehen, deren Wechselwirkung durch ein einfaches Zweikörperpotential beschrieben wird, ein sehr einfaches Modell darstellen, ist ihre Optimierung bereits eine große Herausforderung. Die hohe Zahl an lokalen Minima auf der Potentialfläche und die hohe Dimension des Problems beschränken die Verfahren auf sehr einfache Näherungen und relativ geringe Schrittzahlen. Die Anwendung hat gezeigt, daß sich die Algorithmen in zwei Klassen unterteilen lassen. Die glättenden Verfahren (DEM, GDA, QMA, EDP) funktionieren im Moment nur bei sehr kleinen Atomzahlen zufriedenstellend und finden sie schon 30 Atomen kein globales Minimum mehr. Sie unterscheiden sich vor allem durch ihre Rechenzeiten, die in Tabelle 4.12 für einige Cluster angegeben sind. Neben dem Verfahren

	# Atome		
	10	20	30
DEM			
SD	3.43s	27.24s	88.55s
CG	3.22s	29.09s	77.79s
GDA			
SD	2.20s	30.51s	126.71s
QMA			
SD	8.45s	62.98s	178.12s
EDP			
SD	8.38s	367.77s	1397.92s

TABELLE 4.12: *Rechenzeiten der glättenden Verfahren*

sind in der Tabelle auch die verwendeten lokalen Minimierungsverfahren angegeben. Man sieht, daß DEM und GDA etwa den gleichen Rechenaufwand besitzen, während der QMA-Algorithmus bereits die doppelte Rechenzeit benötigt, da er von einem zusätzlichen Faktor (\hbar) abhängt, wodurch merklich mehr Minimierungsschritte durchgeführt werden müssen. Beim EDP-Algorithmus steigt der Rechenaufwand sogar um eine ganze Ordnung, weil bei ihm zum Potential der zusätzliche Entropieterm hinzukommt. Er minimiert im Gegensatz

zu allen anderen Verfahren die freie Energie, wodurch ein Vergleich schwierig ist. Durch die kleine Verbesserung bei der Wahl der Startkonfiguration, hat das EDP-Verfahren die meisten globalen Minima gefunden, die sich damit innerhalb der Näherung auch als die thermodynamisch bevorzugten Grundzustände erwiesen haben.

Die insgesamt ernüchternden Ergebnisse der glättenden Verfahren beruhen, wie schon mehrfach erwähnt, auf der simplen Näherung durch den Einteilchenansatz. Dieser Ansatz ist korrekt nur für freie Teilchen, die untereinander nicht interagieren können. Deshalb stellt er auch eine gute Näherung für hohe Temperaturen dar, wo die potentielle gegenüber der kinetischen Energie kaum eine Rolle spielt und die Atome sich wie freie Teilchen bewegen. Bei niedrigeren Temperaturen jedoch ist der Ansatz nicht mehr ausreichend, denn durch die Wirkung des Potentials wird der Konfigurationsraum in viele Gebiete mit hoher und niedriger Wahrscheinlichkeit getrennt. Die Beschreibung dieser Eigenschaft kann ein Einteilchenansatz nicht leisten, so daß alle Verfahren scheitern müssen, sobald es mehrere Konfigurationen mit hoher Wahrscheinlichkeit gibt. Die erfolgversprechendste Möglichkeit zur Verbesserung der Algorithmen liegt also in der Verwendung eines besseren Ansatzes, wobei das Problem entsteht, daß aus den meisten Ansätzen ein nicht analytisch berechenbares Potential resultiert.

Die zweite Klasse von Algorithmen benutzt keine geglättete Potentialfunktion. Der hier verwendete Basin-Hopping-Algorithmus benutzt ausschließlich das unveränderte Lennard-Jones-Potential und berechnet den aktuellen Potentialwert mithilfe eines lokalen Minimierungsverfahrens. Die resultierende Transformation hat Ähnlichkeit mit der durch die nicht-lineare Diffusionsgleichung erzeugten Treppenfunktion, besitzt aber den entscheidenden Vorteil, daß bei ihr die hohen Energiebarrieren fehlen. Die Rechenzeit hängt beim Basin-Hopping-Algorithmus sehr stark von der Gesamtzahl der Schritte ab, so daß er bei kleinen Clustern durchaus mit den anderen Verfahren konkurrieren kann, weil eine geringe Anzahl von Monte-Carlo Schritten genügt, um die globalen Minima zu lokalisieren. Bei größeren Problemen muß man die Anzahl der Schritte jedoch drastisch erhöhen, so daß ein Vergleich der Rechenzeit mit den anderen Verfahren eigentlich wenig sinnvoll ist. Im Rahmen dieser Arbeit wurde die Auswertung des Potentials mithilfe der linked-cell Struktur durchgeführt, wodurch die Abhängigkeit des Aufwandes von der Anzahl der Atome auf eine lineare Beziehung reduziert werden konnte. Dadurch kann der Algorithmus auch auf relativ große Cluster mit bis zu einigen Hundert Atomen in akzeptabler Zeit angewendet werden. In Kapitel 4.5 wurden so Minima für Cluster mit 15 bis 160 und 309 Atomen berechnet. Wales und Doye [8] haben mit ihrem ursprünglichen Verfahren sämtliche aus der Literatur bekannten globalen Minima gefunden. Daran erkennt man die momentane Überlegenheit der Algorithmen, die zwar einen hohen Aufwand haben, da sie einen großen Teil des Konfigurationsraumes durchsuchen, dafür aber keine Fehler durch ungenügenden Näherungen machen. Dieser Vorteil wird auch im nächsten Kapitel deutlich werden, wo Cluster aus Wassermolekülen optimiert werden sollen.

Kapitel 5

Optimierung von Wasser-Clustern

5.1 Das TIP4P Potential

Es existiert eine große Anzahl von Modellen für Wassermoleküle, die entwickelt wurden, um die Eigenschaften von Wasser genauer zu untersuchen. Die Idee ist, daß das Modell unbekannte Eigenschaften vorhersagen kann, wenn man es möglichst gut an bekannte Eigenschaften anpaßt. Allerdings scheint es kaum möglich zu sein, alle Eigenschaften von Wasser durch ein einfaches Modell zu beschreiben. Mit einfachem Modell ist in diesem Zusammenhang statisch gemeint, alle Parameter sind also konstant. Bessere Übereinstimmungen kann man durch dynamische Modelle erhalten, bei denen zum Beispiel eine abstandsabhängige Elektronendichte benutzt wird oder polarisierbare Moleküle durch variable Atomabstände (SWFLEX-AI [17]) erzeugt werden. Die einfachen Modelle werden daher meist speziell an einen oder wenige Parameter angepaßt, in denen sie dann besonders gut mit den physikalischen Werten übereinstimmen. Für die Energie-Minimierung wird hier das TIP4P-Potential [11] benutzt, welches eine durchschnittlich gute Übereinstimmung des Dipolmoments und der Diffusionskonstante bietet. Die Temperaturabhängigkeit der Dichte wird durch dieses Modell allerdings eher schlecht wiedergegeben.

Wie der Name schon andeutet, besteht ein Wassermolekül bei Verwendung des TIP4P-Modells aus vier Einheiten, den drei dem realen Molekül entsprechenden Atomen und einem Hilfspunkt, der die negative Ladung trägt. Die positiven Ladungen befinden sich an den Orten der beiden Wasserstoffatome. Demzufolge können diese drei Punkte durch ein Coulomb-Potential interagieren, während auf das Sauerstoffatom ausschließlich ein Lennard-Jones-Potential wirkt. Alle Bindungen und Winkel innerhalb eines Moleküls werden im TIP4P-Modell als starr angenommen, so daß sich jedes Molekül als starrer Körper nur in seinen drei Translations- und drei Rotations-Freiheitsgraden bewegen kann. Das hat den Vorteil, daß die Potentiale nur zwischen Atomen verschiedener Moleküle berechnet werden müssen. Außerdem führen nicht starre Modelle, die die intramolekularen Bindungen zum Beispiel durch harmonische Potentiale berücksichtigen, zu numerischen Problemen bei der Minimierung. Die Energien dieser Bindungen sind nämlich schon bei geringen Abweichungen sehr hoch, so daß die Schrittweiten extrem klein gewählt werden müssen. Da man deswegen davon ausgehen kann, daß sich bei einer minimierten Konfiguration sowieso alle intramolekularen Bindungen und Winkel nicht von ihren Normalwerten unterscheiden, ist es bei einem Wasser-Modell sinnvoll, die Bindungen als starr zu betrachten. Dann muß

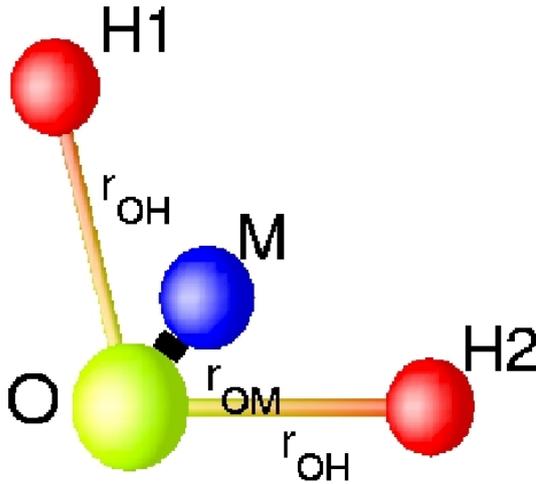


ABBILDUNG 5.1: TIP4P-Modell

TIP4P		
r_{OH}	[Å]	0.9572
r_{OM}	[Å]	0.15
θ_{HOH}	[°]	104.52
θ_{HOM}	[°]	52.26
q_H	[e]	0.52
q_M	[e]	-1.04
ϵ_O	[kcal/mol]	0.155
σ_O	[Å]	3.15365
g	[kcalÅ/mol e ²]	331.842

ABBILDUNG 5.2: TIP4P-Parameter [11]

man allerdings einen geringen Mehraufwand bei der Berechnung der Ableitungen in den Rotationsfreiheitsgraden des Moleküls in Kauf nehmen.

Den genauen Aufbau eines Wassermoleküls und die dazugehörigen Parameter des TIP4P-Modells kann man Abbildung 5.1 und der Tabelle 5.2 entnehmen. Die Abstände r_{OH} und der Winkel θ_{HOH} wurden dabei an die experimentellen Daten in der Gasphase angepaßt. Das resultierende Dipolmoment liegt etwas unter dem experimentellen Wert von Wasser und seine größte Dichte hat das Modell bei -25°C , also weit unter dem physikalischen Wert von Wasser. Das Modell bietet also eher eine durchschnittlich gute Beschreibung von echtem Wasser, ist dafür aber mit moderatem numerischen Aufwand zu benutzen. Das Gesamtpotential eines aus N Molekülen bestehenden Wasser-Clusters sieht dann wie folgt aus:

$$U(\mathbf{x}) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{n_i, m_j \in \{H1, H2, M\}} g \frac{q_i^{n_i} q_j^{m_j}}{r_{ij}^{n_i m_j}} \quad (5.1)$$

$$+ \sum_{i=1}^N \sum_{j=i+1}^N 4\epsilon_{ij}^{OO} \left(\left(\frac{\sigma_{ij}^{OO}}{r_{ij}^{OO}} \right)^{12} - \left(\frac{\sigma_{ij}^{OO}}{r_{ij}^{OO}} \right)^6 \right) \quad (5.2)$$

Dabei gilt wieder die Beziehung $r_{ij} = |\vec{x}_i - \vec{x}_j|$ und die ϵ_{ij} und σ_{ij} werden nach den Lorentz-Berthelot-Mischungsregeln berechnet.

5.1.1 Bewegung eines Moleküls als starrer Körper

Die Beschreibung eines Moleküls als starrer Körper ist nicht ganz trivial, daher wird in diesem Kapitel genauer eingegangen auf die Darstellung eines Moleküls und der Ableitung des Potentials nach den äußeren Koordinaten.

Die oben angegebenen Parameter für die Abstände und Winkel zwischen den Atomen eines Moleküls beziehen sich auf das molekülinterne Koordinatensystem. Die Minimierung soll aber in den äußeren Koordinaten stattfinden. Dazu wird jedem Molekül ein inneres

Koordinatensystem zugeordnet, welches hier willkürlich seinen Ursprung im Sauerstoff-Atom haben soll. Alle Atome sollen in der x-y-Ebene liegen und die Verbindung O-H1 beschreibt die x-Achse. Die Positionen der Atome in diesem System werden als \vec{x}_I^k mit $k \in \{O, H1, H2, M\}$ bezeichnet. Der Ursprung des inneren Systems liegt im äußeren Koordinatensystem an der Stelle \vec{R}_i , die Orientierung des Moleküls wird durch drei Euler-Winkel $\vec{\phi}_i$ angegeben. Der $(3 \cdot 4N)$ -dimensionale Vektor \mathbf{x} stellt aber die Koordinaten aller Atome im äußeren System dar, so daß man zwischen beiden Darstellungen transformieren muß, um das Potential berechnen zu können. Um zum Beispiel \vec{x}_i^M zu erhalten, muß man folgendes berechnen:

$$\vec{x}_i^M = \vec{R}_i + D^{-1}(\vec{\phi}_i) \vec{x}_I^M \quad (5.3)$$

Die Matrix D^{-1} setzt sich aus drei Drehungen zusammen und sieht insgesamt wie folgt aus:

$$D^{-1}(\phi^1, \phi^2, \phi^3) = \begin{pmatrix} \cos(\phi^1) \cos(\phi^3) - \sin(\phi^1) \cos(\phi^2) \sin(\phi^3) & -\cos(\phi^1) \sin(\phi^3) - \sin(\phi^1) \cos(\phi^2) \cos(\phi^3) & \sin(\phi^1) \sin(\phi^2) \\ \sin(\phi^1) \cos(\phi^3) + \cos(\phi^1) \cos(\phi^2) \sin(\phi^3) & -\sin(\phi^1) \sin(\phi^3) + \cos(\phi^1) \cos(\phi^2) \cos(\phi^3) & -\cos(\phi^1) \sin(\phi^2) \\ \sin(\phi^2) \sin(\phi^3) & \sin(\phi^2) \cos(\phi^3) & \cos(\phi^2) \end{pmatrix} \quad (5.4)$$

Will man dann den Gradienten $\nabla_{\vec{r}_i, \vec{\phi}_i}$ von $U(\mathbf{x})$ bilden, so treten beim Nachdifferenzieren die Terme

$$\frac{\partial}{\partial \vec{R}_i} |\vec{x}_i^n - \vec{x}_j^m| = \frac{\vec{r}_{ij}^{nm}}{|\vec{r}_{ij}^{nm}|} \quad (5.5)$$

$$\frac{\partial}{\partial \vec{\phi}_i} |\vec{x}_i^n - \vec{x}_j^m| = \frac{\partial}{\partial \vec{\phi}_i} \left(D^{-1}(\vec{\phi}_i) \vec{x}_I^n \right)^T \cdot \frac{\vec{r}_{ij}^{nm}}{|\vec{r}_{ij}^{nm}|} \quad (5.6)$$

mit $n, m \in \{O, H1, H2, M\}$ auf, die sich mit Hilfe von Gl.5.3 und 5.4 berechnen lassen. Die so entstehenden Formeln sehen zwar sehr umständlich aus, sind aber numerisch wesentlich vorteilhafter als die Beschreibung der internen Bindungen durch harmonische Potentiale.

5.2 Anwendung der DEM auf Wasser

Das TIP4P-Potential setzt sich aus einem Coulomb- und einem Lennard-Jones-Anteil zusammen.

$$U(\mathbf{x}) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{n_i, m_j \in \{H1, H2, M\}} g \frac{q_i^{n_i} q_j^{m_j}}{r_{ij}^{n_i m_j}} \quad (5.7)$$

$$+ \sum_{i=1}^N \sum_{j=i+1}^N 4\epsilon_{ij}^{OO} \left(\left(\frac{\sigma_{ij}^{OO}}{r_{ij}^{OO}} \right)^{12} - \left(\frac{\sigma_{ij}^{OO}}{r_{ij}^{OO}} \right)^6 \right) \quad (5.8)$$

$$= \sum_{i=1}^N \sum_{j=i+1}^N \sum_{n_i, m_j \in \{H1, H2, M\}} U_{ij, n_i m_j}^C + \sum_{i=1}^N \sum_{j=i+1}^N U_{ij}^{LJ} \quad (5.9)$$

Um die Lösung der Diffusionsgleichung 2.9 analytisch berechnen zu können, muß man mit diesem Potential das Integral

$$u(\mathbf{x}, t) = (4\pi t)^{-\frac{4N3}{2}} \int U(\mathbf{y}) e^{-\frac{1}{4t} \sum_{i=1}^N \sum_{n_i \in \{H1, H2, M\}} (\vec{x}_i^{n_i} - \vec{y}_i^{n_i})^2} d\mathbf{x} \quad (5.10)$$

lösen können. Da jedes Wassermolekül in dem Modell vier Teilchen beinhaltet, muß die Integration über $3 \cdot 4N$ Dimensionen durchgeführt werden. Die Lösung für den Lennard-Jones-Anteil ist im Prinzip schon aus dem vorherigen Kapitel bekannt und kann übernommen werden. Da ausschließlich die Sauerstoffatome durch ein Lennard-Jones-Potential interagieren kann man folgende Vereinfachung machen:

$$\epsilon_{ij}^{\text{OO}} = \epsilon_i^{\text{O}} \quad \text{und} \quad \sigma_{ij}^{\text{OO}} = \sigma_i^{\text{O}} \quad (5.11)$$

Deswegen ist es wieder möglich, für alle vorkommenden Lennard-Jones-Paare nur eine Näherung durch Gauß-Funktionen durchzuführen. Allerdings sind die Werte ϵ_{O} und σ_{O} jetzt verschieden von eins, so daß die Parameter neu berechnet werden (siehe Tabelle 5.1). Alternativ dazu könnte man auch die Parameter für die auf eins skalierten Werte von ϵ

k	a_k	b_k
1	436.1110	0.8954
2	-0.2124	0.1266

TABELLE 5.1: Parameter von \tilde{U}_{ij}^{LJ}

und σ benutzen, müßte dann aber jeweils die Ersetzung $\tilde{r}_{ij}^{\text{OO}} = r_{ij}^{\text{OO}}/\sigma_i^{\text{O}}$ machen und die Summe mit ϵ_i^{O} multiplizieren. Eine zusätzliche Gauß-Funktion wie beim reinen Lennard-Jones-Problem ist hier jedoch nicht notwendig, da das Cluster schon durch die anziehend wirkenden Coulombkräfte zusammengehalten wird. Bei einem Test mit einem solchen Zusatzterm wirkte sich dieser sogar nachteilig aus, da er den Lennard-Jones-Anteil dominanter machte und so völlig andere Minima bevorzugte, als das reine TIP4P-Potential. Der Lennard-Jones-Anteil wird dann folgendermaßen dargestellt:

$$U_{ij}^{LJ} = 4\epsilon_i^{\text{O}} \left(\left(\frac{\sigma_i^{\text{O}}}{r_{ij}^{\text{OO}}} \right)^{12} - \left(\frac{\sigma_i^{\text{O}}}{r_{ij}^{\text{OO}}} \right)^6 \right) \approx \sum_{k=1}^2 a_k e^{-b_k r_{ij}^{\text{OO}}} = \tilde{U}_{ij}^{LJ} \quad (5.12)$$

Für diesen Anteil ergibt sich wieder die schon bekannte Lösung

$$\tilde{u}_{ij}^{LJ}(\vec{x}_i^{\text{O}}, \vec{x}_j^{\text{O}}, t) = \sum_{k=1}^2 a_k (1 + 8b_k t)^{-\frac{3}{2}} e^{-\frac{b_k r_{ij}^{\text{OO}}}{1+8b_k t}} \quad (5.13)$$

Der Coulomb-Anteil des Potentials muß nicht durch eine Summe von Gauß-Funktionen genähert werden, denn für die Faltung von $1/r$ (mit $r = |\vec{x}|$) gilt folgende Formel:

$$(4\pi t)^{-\frac{3}{2}} \int \frac{1}{r} e^{-\frac{r^2}{4t}} d^3x = \frac{1}{2r} \left(\operatorname{erfc} \left(-\frac{r}{\sqrt{8t}} \right) - \operatorname{erfc} \left(\frac{r}{\sqrt{8t}} \right) \right) \quad (5.14)$$

Als Lösung ergibt sich die Differenz von zwei komplementären Fehlerfunktionen, die wie folgt definiert sind:

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x) = 1 - \frac{2}{\sqrt{\pi}} \int_0^x e^{-y^2} dy \quad (5.15)$$

Damit ergibt sich dann für den Coulomb-Anteil des TIP4P-Potentials

$$\tilde{u}_{ij,n_i m_j}^C = g \frac{q_i^{n_i} q_j^{m_j}}{2r_{ij}^{n_i m_j}} \left(\operatorname{erfc} \left(-\frac{r_{ij}^{n_i m_j}}{\sqrt{8t}} \right) - \operatorname{erfc} \left(\frac{r_{ij}^{n_i m_j}}{\sqrt{8t}} \right) \right) \quad (5.16)$$

und die Gesamtlösung hat die Form

$$\tilde{u}(\mathbf{x}, t) = \sum_{i=1}^N \sum_{j=i+1}^N \sum_{n_i, m_j \in \{H1, H2, M\}} \tilde{u}_{ij, n_i m_j}^C(\mathbf{x}, t) + \sum_{i=1}^N \sum_{j=i+1}^N \tilde{u}_{ij}^{LJ}(\mathbf{x}, t) \quad (5.17)$$

Mit diesen Vorbereitungen kann der DEM-Algorithmus auf Cluster aus Wassermolekülen angewendet werden. Allerdings macht die Betrachtung von Translations- und Rotationsfreiheitsgraden eine Änderung bei den lokalen Minimierungsverfahren notwendig. Hier wurde ein einfaches Steepest-Descent-Verfahren mit Armijo-Schrittweitensteuerung verwendet (siehe Kap.3.1.1). Wie oben beschrieben wird der Gradient in den äußeren Variablen \vec{R}_i und $\vec{\phi}_i$ des Moleküls berechnet. Führt man dann die Minimierungsschritte in beiden Variablen gleichzeitig aus, so bekommt man Schwierigkeiten, ein geeignetes Abbruchkriterium für die lokale Minimierungsprozedur zu finden. Eigentlich würde man verlangen, daß der Betrag des Gradienten unter einen gewissen Wert sinkt, so daß die Bedingung für ein Extremum oder einen Sattelpunkt erfüllt wird. Nun stellt man aber fest, daß in einigen Fällen die Werte des Gradienten in den Rotationsfreiheitsgraden deutlich verschieden von null sind, obwohl jede weitere Veränderung in Richtung des negativen Gradienten der zugehörigen Winkel einen Anstieg des Potentials zur Folge hat. Das ist eine Folge der Darstellung durch Euler-Winkel.

Bei der Darstellung durch Euler-Winkel wird die Orientierung im Raum durch drei aufeinanderfolgende Drehungen um verschiedene Achsen beschrieben. Dabei muß die Reihenfolge der Drehungen stets die gleiche bleiben. Der Nachteil dieser Darstellung ist, daß eine kleine Veränderung der Orientierung des beschriebenen Moleküls zu großen Änderungen der drei Winkel führen kann, was zu numerischen Problemen führen kann. Außerdem ist das Problem des sogenannten „gimbal lock“ bei Euler-Winkeln bekannt. So bezeichnet man den Effekt, daß bei bestimmten Drehungen, zum Beispiel wenn $\phi_2 = 0$ gilt, zwei der drei Drehachsen parallel liegen, wodurch man einen Freiheitsgrad verliert. Treten solche Probleme auf, so muß man trotzdem die Minimierung beenden und eine nicht optimale Orientierung eines oder mehrerer Moleküle in Kauf nehmen. Um aber dadurch nicht auch die translatorische Bewegung zu beeinflussen, kann man die Minimierung in einen Translations- und einen Rotationsanteil aufspalten, indem man zum Beispiel immer abwechselnd einen Schritt des einen und des anderen Anteils durchführt. Den Rotationsanteil bricht man dann auch ab, wenn der Gradient zwar noch ungleich null ist, aber die Schrittweite unter einen bestimmten Wert gefallen ist. Beheben könnte man diese Probleme durch die Benutzung von anderen Darstellungen für die Orientierung im Raum. Quaternionen zum Beispiel beschreiben jede Position durch die Drehung um eine Achse. Man muß dann zwar mit einer Variable mehr umgehen (drei für die Achse, eine für den Winkel), erhält dadurch aber eine wesentlich stabilere Darstellung.

Tabelle 5.2 zeigt die besten erreichten Potentialwerte für Wasser-Cluster mit bis zu zehn Molekülen, wobei als lokales Minimierungsverfahren ein einfacher Steepest-Descent-Algorithmus diente. Die Ergebnisse ließen sich deutlich verbessern, wenn man statt wie es Scheraga [7] vorschlägt bei großen Glättungszeit t zu beginnen, eine Suche nach der Konfiguration mit dem niedrigsten Potential bei relativ kleinem t durchführte und diese Konfiguration dann als Anfangswert für den DEM-Algorithmus verwendete. Dabei wurde $t_{max} = 2.0$ gewählt und 100 verschiedene Anfangskonfigurationen lokal minimiert.

Offensichtlich unterscheiden sich auch die Potentialwerte der gefundenen globalen Minima von denen aus [9]. Da die Konfigurationen der Minima aber übereinstimmen, ist der

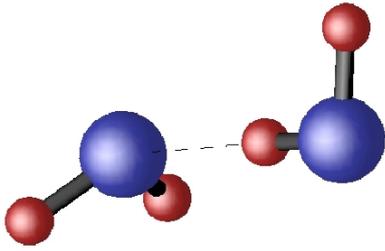


ABBILDUNG 5.3: 2 Wasser Moleküle

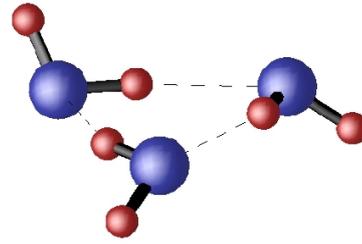


ABBILDUNG 5.4: 3 Wasser Moleküle

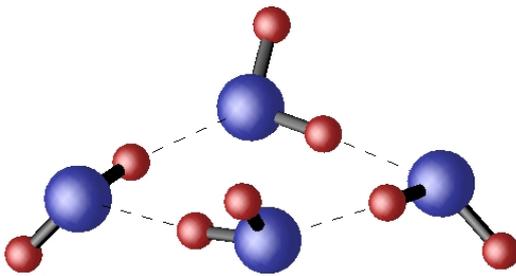


ABBILDUNG 5.5: 4 Wasser Moleküle

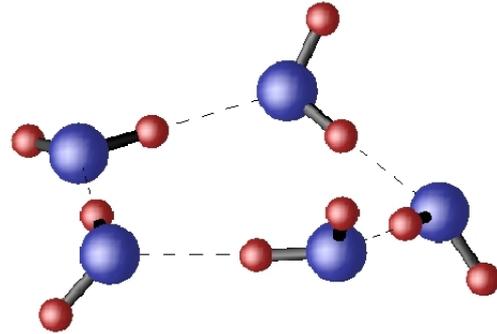


ABBILDUNG 5.6: 5 Wasser Moleküle

Grund wahrscheinlich eine geringe Abweichung in irgendeinem der verwendeten TIP4P-Parameter. Mit der hier verwendeten Methode wurden die globalen Minima der Cluster mit bis zu fünf Molekülen gefunden. Scheraga berichtet in [7] ebenfalls, daß er Cluster mit bis zu fünf Molekülen erfolgreich minimieren konnte. Obwohl er ein anderes Potentialmodell (MCY) für die Wassermoleküle verwendet, stimmen die angegebenen minimalen Konfigurationen mit denen aus [9] überein. Das Ergebnis für das Cluster mit sechs Molekülen weicht nur sehr wenig von dem Minimum ab, welches mit dem Basin-Hopping-Algorithmus gefunden wurde (siehe Kap.5.3), in der Konfiguration ist dieser Unterschied allerdings nicht zu erkennen.

Das gefundene Minimum mit 10 Molekülen ist offensichtlich kein tiefes Minimum, da

#	U_{global} [kJ/mol]	U_{DEM} [kJ/mol]	global?
2	-26.0876	-26.0817	+
3	-69.9939	-69.9774	+
4	-116.5904	-116.5637	+
5	-152.1090	-152.0746	+
6	-197.7805	-197.1480	-
7	-243.5724	-238.5964	-
8	-305.5183	-296.0275	-
9	-344.4352	-313.4166	-
10	-391.0227	-343.3072	-

TABELLE 5.2: Ergebnisse des DEM-Algorithmus (U_{global} aus [9])

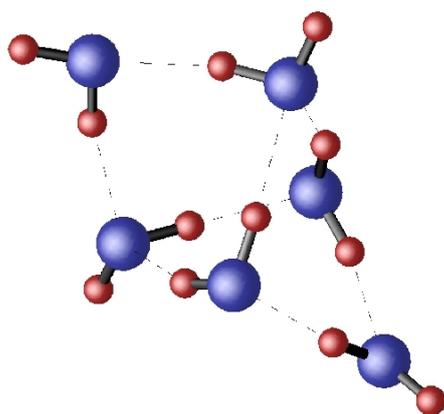


ABBILDUNG 5.7: 6 Wasser Moleküle

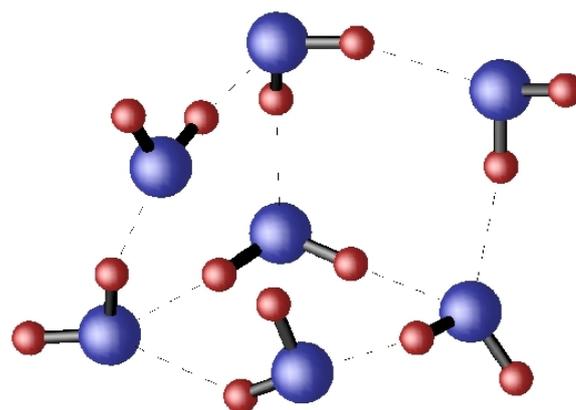


ABBILDUNG 5.8: 7 Wasser Moleküle

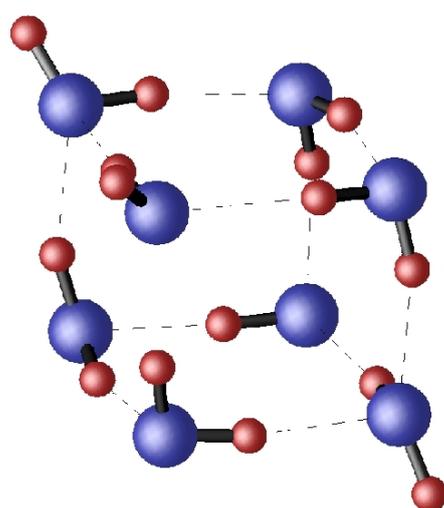


ABBILDUNG 5.9: 8 Wasser Moleküle

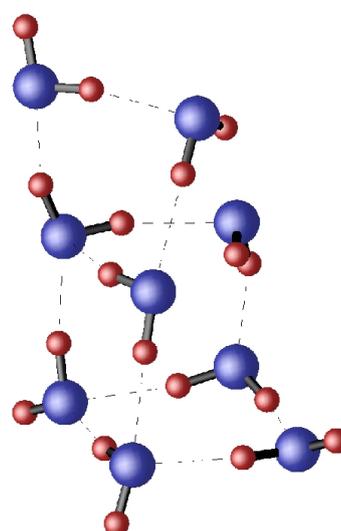


ABBILDUNG 5.10: 9 Wasser Moleküle

alle Moleküle etwa in einer Ebene liegen. Interessant ist insbesondere das gefundene Minimum mit acht Molekülen. Seine Struktur sieht dem tatsächlichen globalen Minimum mit deutlich niedrigerem Potential sehr ähnlich (siehe Abbildung 5.13 in Kapitel 5.3). Die Sauerstoffatome befinden sich auch relativ zueinander in denselben Positionen, nur die relative Lage der Wasserstoffatome ist anders. Hieran kann man erkennen, welche zusätzlichen Schwierigkeiten durch die Betrachtung von Molekülen gegenüber einfachen atomaren Clustern hinzukommen. Die Einbeziehung von Rotationsfreiheitsgraden stellt eine große Herausforderung an Minimierungsalgorithmen dar. Es kann nämlich eine Situation auftreten, die es dem Algorithmus sogar schon bei nur zwei Wassermolekülen unmöglich macht, ein Energieminimum zu finden, denn liegen die Ebenen der beiden Moleküle irgendetwas einmal parallel zueinander, so wird diese relative Position mit großer Wahrscheinlichkeit beibehalten, da alle Komponenten des Gradienten, die eine Drehung senkrecht zu dieser Ebene zur Folge hätten, null sind. Diese Konfiguration von zwei Molekülen stellt aber kein

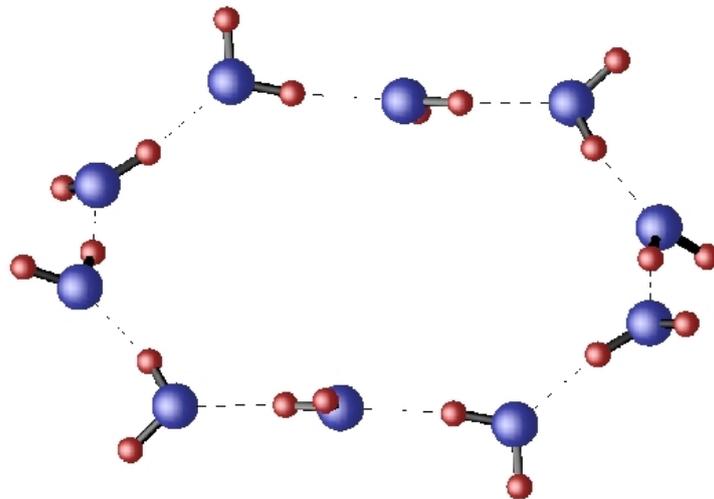


ABBILDUNG 5.11: 10 Wasser Moleküle

Energieminimum dar. Ähnliche Situationen können für beliebige Anzahlen von Molekülen auftreten. Bei den atomaren Clustern spielt dieses Problem keine Rolle, weil die einzelnen Atome sich leichter von Sattelpunkten und Maxima der Energiefläche wegbewegen können, während bei den Wassermolekülen noch der Nachteil der schwachen Darstellung durch Euler-Winkel hinzukommt. Eine andere Darstellung wäre hier also sicherlich von Vorteil (siehe auch Kapitel 5.4).

5.3 Anwendung des Basin-Hopping auf Wasser

Auch bei der Optimierung von Wasser-Clustern ist der Basin-Hopping Algorithmus allen anderen Verfahren weit überlegen. Sein einfacher Aufbau verschafft ihm im Fall molekularer Cluster den Vorteil, besser mit den zusätzlichen Freiheitsgraden umgehen zu können. Zwar treten bei ihm prinzipiell dieselben Probleme zum Beispiel bei der Benutzung der Euler-Winkel auf, aber durch die zufällige Änderung der Positionen und Winkel in jedem Monte-Carlo Schritt werden problematische Konfigurationen immer schnell ersetzt oder gar nicht erst akzeptiert. Im Detail wurde der Basin-Hopping-Algorithmus für die TIP4P-Cluster wie folgt implementiert:

Die Positionen und Orientierungen der Moleküle wurden wieder durch ihre äußeren Koordinaten \vec{R}_i und $\vec{\phi}_i$ repräsentiert. Daher minimierte der Steepest-Descent-Algorithmus das Cluster auch jeweils in diesen Koordinaten. Wie auch in [9] berichtet, erwies es sich allerdings als vorteilhaft, die Monte-Carlo Schritte getrennt in den Translationen und den Rotationen durchzuführen. Das bedeutet, es wurden immer abwechselnd 60 Schritte gemacht, in denen \vec{R}_i um maximal c_{trans} verschoben wurde, und 60 Schritte, in denen die Winkel $\vec{\phi}_i$ um maximal c_{rot} verändert wurden. Die Wahl der einzelnen Parameter kann aus Tabelle 5.3 entnommen werden. So wurde der Algorithmus auf Wasser-Cluster mit bis zu 17 Molekülen angewendet. Die aus der Literatur (siehe [9]) bekannten Minima wurden für bis zu 14 Moleküle gefunden. Da die Potentialwerte wie oben schon erwähnt trotzdem von denen aus der Literatur abweichen, sind sie in Tabelle 5.4 angegeben. Im Vergleich

Parameter	Wert
k_{max}	20000
β [kJ/mol]	0.5
c_{trans}	1.0
c_{rot}	0.5

TABELLE 5.3: Parameter für den Basin-Hopping Algorithmus

#	$U_{min}^{[9]}$ [kJ/mol]	U_{min}^{TIP4P} [kJ/mol]	global?
2	-26.0867	-26.0817	+
3	-69.9939	-69.9774	+
4	-116.5904	-116.5637	+
5	-152.1090	-152.0746	+
6	-197.7805	-197.1480	+
7	-243.5724	-243.4861	+
8	-305.5183	-305.3297	+
9	-344.4352	-344.2340	+
10	-391.0227	-390.7075	+
11	-431.4891	-431.1773	+
12	-492.9082	-488.3070	-
13	-532.9705	-531.3885	-
14	-582.9911	-579.0956	-
15	-628.3723	-621.1847	-
16	-681.1925	-664.5827	-
17	-723.8077	-707.6357	-

TABELLE 5.4: Ergebnisse des Basin-Hopping Algorithmus

zur DEM findet der Basin-Hopping Algorithmus deutlich bessere Minima und bis zu 11 Molekülen stimmen die Minima auch mit denen aus [9] überein. Für die Cluster mit 12 bis 14 Molekülen sind die Potentialwerte zwar deutlich niedriger als die Referenzwerte, aber trotzdem stimmen die Konfigurationen der Cluster ziemlich gut überein. Das liegt an der leicht abweichenden Orientierung der Wasserstoffatome, deren optimale Lage nicht gefunden wurde. Da hier die Gesamtzahl der Schritte wesentlich geringer war als die in [9] angegebene, sind für die größeren Cluster keine besseren Minima gefunden worden. Die Gesamtleistung gemessen an der Zahl der Atome, vier pro Molekül, ist im Vergleich zur Optimierung von Lennard-Jones Clustern deutlich schlechter. Die durchschnittliche Zeit für die Durchführung von 100 Monte-Carlo Schritten beträgt bei vier Molekülen bzw. 16 Atomen bereits $t_4 = 223s$ und bei der doppelten Anzahl $t_8 = 482s$. Obwohl bei der Potentialauswertung weniger Paare berücksichtigt werden und das Gesamtsystem auch weniger Freiheitsgrade hat, steigt die Rechenzeit gegenüber der Lennard-Jones-Cluster deutlich an. Das liegt vor allem an den Rotationsfreiheitsgraden der Moleküle, die bei der Minimierung eine sehr kleine Schrittweite erfordern und dadurch den Rechenaufwand für jede lokale Minimierung erhöhen. Deswegen wäre auch für das Basin-Hopping eine effizientere Methode für die räumliche Darstellung der Moleküle von Vorteil, so daß mehr Monte-Carlo Schritte

durchgeführt werden könnten. Wales und Hodges [9] berichten allerdings auch, daß sie bei der Optimierung der TIP4P-Cluster wesentlich mehr (100000) Monte-Carlo Schritte benötigt haben, als bei der Minimierung der Lennard-Jones Cluster. Das bedeutet, daß die Potentialfläche, in den Koordinaten \vec{R}_i und $\vec{\phi}_i$ betrachtet, bereits für wenige Moleküle im Vergleich zu den atomaren Clustern eine wesentlich schwierigere Struktur hat. Diese Eigenschaft könnte man vielleicht ebenfalls durch die Betrachtung des Potentials in einem anderen Koordinatensystem verbessern. Trotzdem hat sich gezeigt, daß der Basin-Hopping Algorithmus auch mit der ungünstig gewählten Darstellung und schwierigeren Potentialen relativ gut umgehen kann.

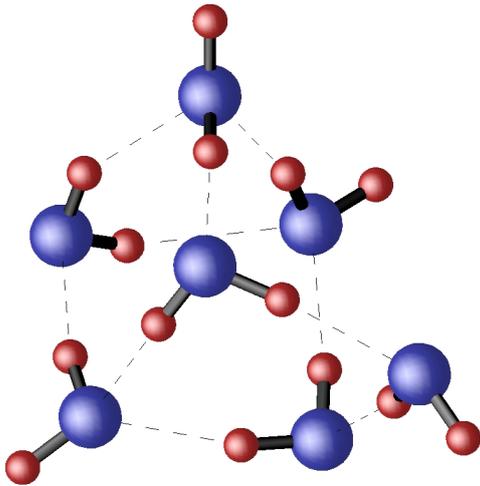


ABBILDUNG 5.12: 7 Wassermoleküle

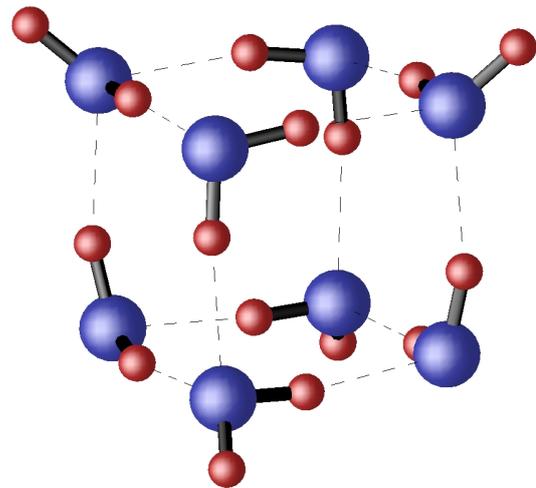


ABBILDUNG 5.13: 8 Wassermoleküle

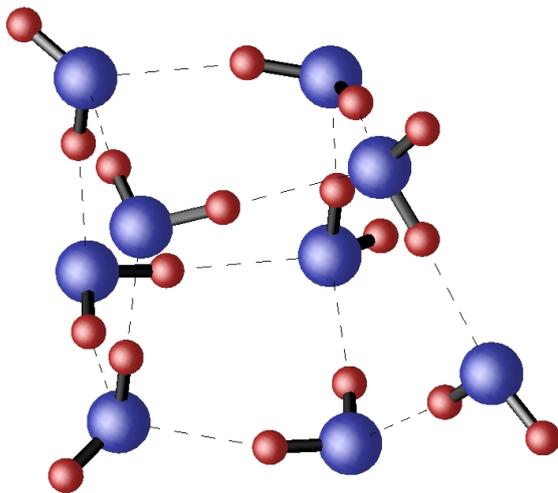


ABBILDUNG 5.14: 9 Wassermoleküle

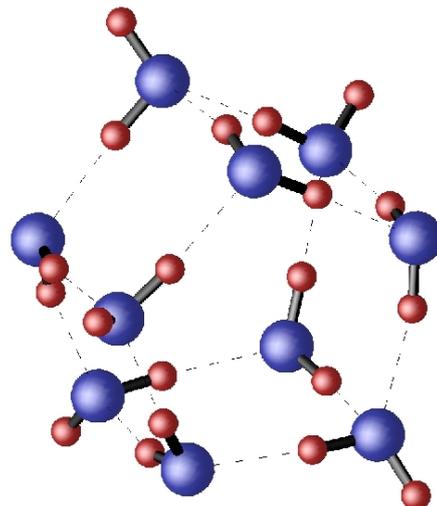


ABBILDUNG 5.15: 10 Wassermoleküle

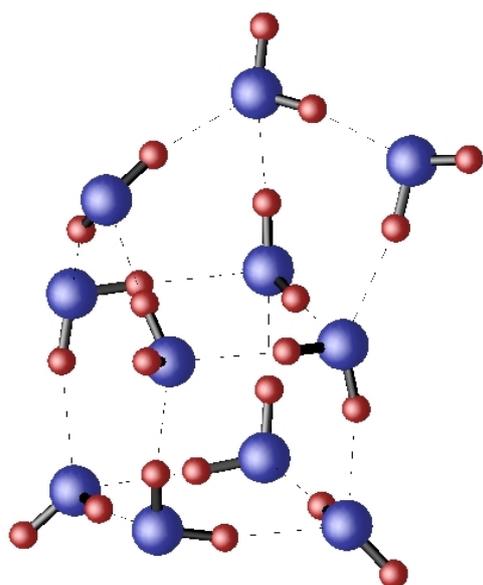


ABBILDUNG 5.16: 11 Wassermoleküle

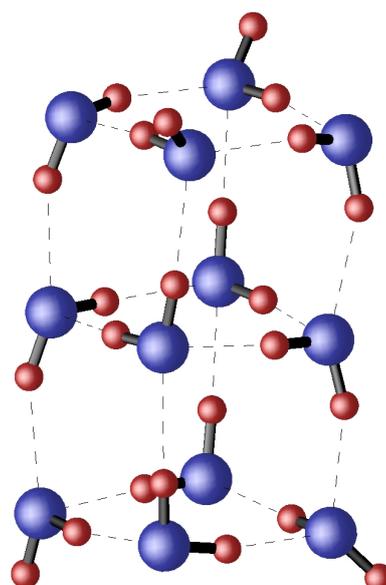


ABBILDUNG 5.17: 12 Wassermoleküle

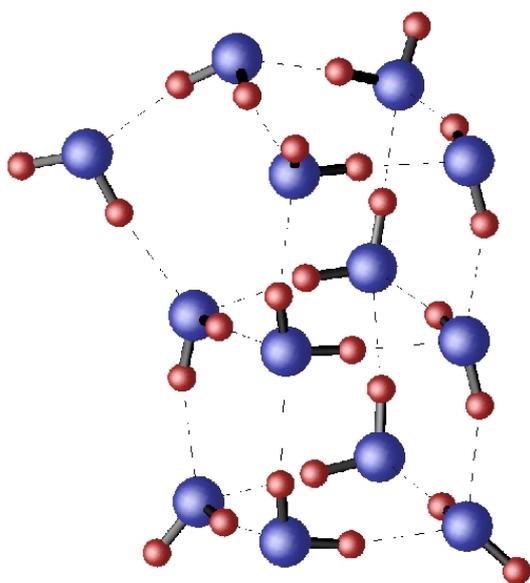


ABBILDUNG 5.18: 13 Wassermoleküle

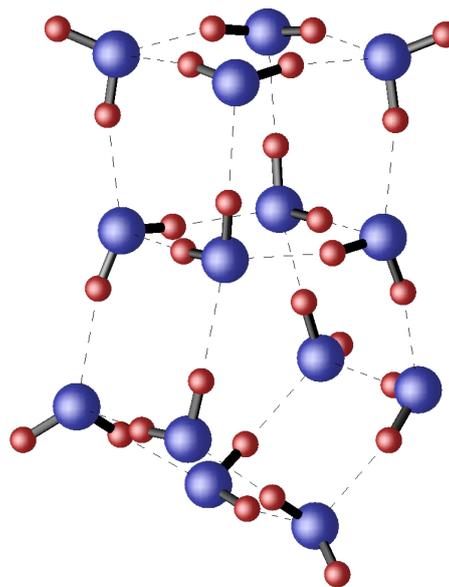


ABBILDUNG 5.19: 14 Wassermoleküle

5.4 Zusammenfassung

Aus der Anwendung zweier Verfahren auf Cluster aus Molekülen ist ersichtlich geworden, daß die Optimierung komplexerer Strukturen wesentlich schwieriger ist als bei einfachen ungebundenen atomaren Clustern. Die Betrachtung von Wasser bietet dafür ein besonders gutes Beispiel, weil Simulationen mit Wassermodellen in allen Bereichen des wissenschaftlichen Rechnens durchgeführt werden. Deswegen existieren viele Potentialmodelle für Wassermoleküle, die alle nur bestimmte physikalische Eigenschaften gut wiedergeben. Das

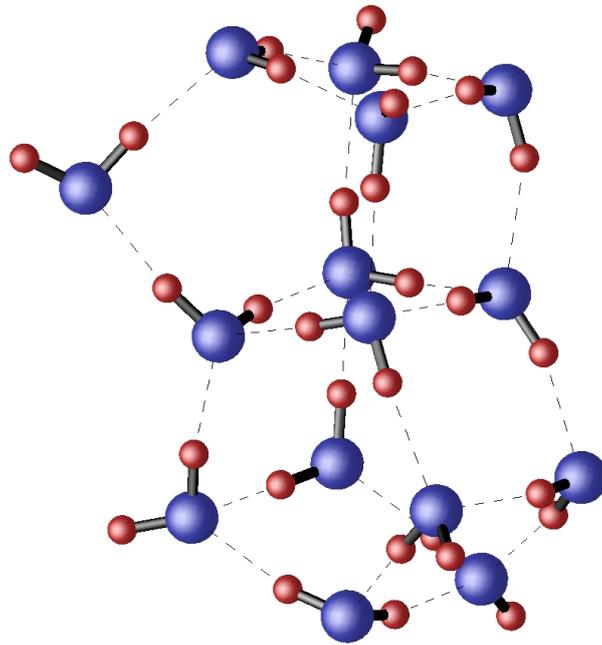


ABBILDUNG 5.20: 15 Wassermoleküle

zeigt, daß es selbst für so einfache Moleküle noch nicht möglich ist, ein einfaches Potential anzugeben, welches alle Eigenschaften von „echtem“ Wasser zufriedenstellend wiedergibt. Dazu zählt vor allem auch, daß die Bindungen innerhalb eines Moleküls eigentlich nicht starr sind, wie es das hier benutzte TIP4P-Potential annimmt. Aus dieser Annahme resultiert die Betrachtung des Moleküls als starrer Körper, der dann in der Literatur meist mit Euler-Winkeln beschrieben wird ([7],[9]). Die Euler-Winkel bringen aber erhebliche Probleme bei der Minimierung mit sich und sollten daher besser durch andere Darstellungen wie zum Beispiel Quaternionen ersetzt werden. Eine andere Möglichkeit die intramolekularen Bindungen zu berücksichtigen ist, sie durch Potentiale darzustellen, die schon bei kleinen Abweichungen von der Ruhelage große Energien haben und das Molekül so zusammenhalten. Aber auch dieses Modell führt zu Problemen, denn die Bindungen müßten eigentlich quantenmechanisch betrachtet werden, weil man Vibrationen zwischen zwei gebundenen Atomen nicht mit beliebiger Energie anregen kann. Das führt bei moleküldynamischen Anwendungen zu Fehlern in der Temperaturmessung, da dort schon Vibrationsenergien auftreten, bevor sie eigentlich erlaubt wären. Im Zusammenhang mit den Optimierungsverfahren bringen solche Bindungspotentiale aber eher andere Schwierigkeiten mit sich. Da diese Potentiale bei kleinen Abweichungen die größten Energien liefern, bestimmen sie die maximale Schrittweite, die man durchführen kann. Das macht die Minimierung vor allem bei der Verwendung von geglätteten (ungebundenen) Potentialen extrem langsam. Man will diese gebundenen Potentiale aber auch nicht mitglätten, weil das die Bindungen leicht zerstören kann und somit zu unsinnigen Konfigurationen führt. Deshalb wählt man bei kleinen Molekülen das kleinere Übel der Euler-Winkel, während man bei großen Molekülen wie Proteinen die Darstellung durch Bindungspotentiale wählt.

Bei den einfachen Wassermolekülen ist der Basin-Hopping-Algorithmus dem glättenden

Verfahren wieder weit überlegen, weil er durch die Verwendung von zufälligen Monte-Carlo Schritten besser mit den Problemen durch die Rotationen umgehen kann. Die DEM findet mit ihrer ursprünglichen Anwendung nur selten die globalen Minima der Cluster mit bis zu fünf Molekülen. Nur durch die größere Einbeziehung eines Zufallsmomentes durch die Suche nach der besten Startkonfiguration läßt sich dieses Ergebnis etwas verbessern. Um das Verfahren sinnvoll auf Molekül-Cluster anwenden zu können, muß eine bessere Darstellung der Moleküle gefunden werden.

Kapitel 6

Optimierung von Proteinen

6.1 Einleitung

Eine große Herausforderung an Optimierungsalgorithmen stellt die Anwendung auf biologische Moleküle dar. Die Vorhersage der dreidimensionalen Struktur von Proteinen ist ein weit verbreitetes Forschungsgebiet, weil sie ein wichtiger Bestandteil von lebenden Zellen sind und auch im menschlichen Körper wichtige Funktionen übernehmen. Man unterscheidet bei ihnen zwischen der primären (α), sekundären (β) und tertiären (γ) Struktur, die durch die Abfolge der Residuen aus Aminosäuren, ihre lokale Struktur und durch die vollständige räumliche Struktur gegeben sind. Das Ziel der Untersuchungen ist es, den Zusammenhang zwischen dem primären Aufbau und der dreidimensionalen Konfiguration eines Proteins zu verstehen, denn die Kenntnis dieses Zusammenhangs würde die Vorhersage seiner chemischen und biologischen Eigenschaften erlauben. Man wäre dann zum Beispiel besser in der Lage, die Daten aus dem „human genome project“ zu interpretieren, Krankheitserreger zu verstehen und medizinisch wirksame Arzneimittel herzustellen. Aus diesen Gründen ist die Optimierung von Proteinen ein wichtiges Anwendungsgebiet von globalen Minimierungsverfahren.

6.2 Das CHARMM-Potential

Die Minimierung der potentiellen Energie zur Vorhersage der tertiären Struktur eines Proteins macht nur Sinn, wenn das verwendete Potential in der Lage ist, das physikalische System gut genug zu beschreiben. Im Moment ist man aber noch weit davon entfernt, für solch große Systeme Potentiale benutzen zu können, die zum Beispiel aus quantenmechanischen Berechnungen resultieren und somit eine theoretisch begründete Form haben. Stattdessen muß man sich mit empirischen Potentialfunktionen zufriedengeben, die die Eigenschaften des Systems nur in engen Bereichen richtig wiedergeben können. Deswegen ist die Übertragung der Ergebnisse auf reale Moleküle noch mit großer Vorsicht zu genießen.

Für die Verwendung des Potentials ist es zunächst wichtig anzugeben, wie die Lage der einzelnen Atome des Moleküls beschrieben werden sollen. Die Angabe der Position des i -ten Atoms durch einen dreidimensionalen Vektor \vec{x}_i reicht zwar prinzipiell aus, aber aus ihr werden weitere Größen berechnet, die die Struktur des Moleküls und die Wirkungsweise des Potentials anschaulicher machen, weil die Potentialanteile direkt aus ihnen berechnet

werden. Diese Größen sollen kurz beschrieben werden:

Wenn die Atome i und j gebunden sind, haben sie Bindungsvektor und -länge

$$\vec{r} = \vec{x}_i - \vec{x}_j \quad r = |\vec{r}| \quad (6.1)$$

Zwei benachbarte Bindungen zwischen den Atomen i , j und k schließen den Bindungswinkel $\theta = \angle(i - j - k)$ mit

$$\cos \theta = \frac{\vec{r}_{ij} \vec{r}_{jk}}{|\vec{r}_{ij}| |\vec{r}_{jk}|} \quad (6.2)$$

ein. Der Winkel zwischen den beiden Ebenen, die durch die benachbarten Atome (i, j, k) und (j, k, l) gegeben sind, heißt Diederwinkel Φ und berechnet sich nach

$$\cos \Phi = \frac{(r_{ij} \times r_{jk}) \cdot (r_{jk} \times r_{kl})}{|r_{ij} \times r_{jk}| |r_{jk} \times r_{kl}|} \quad (6.3)$$

Durch die Angabe aller Bindungslängen, Winkel und Diederwinkel ist die Struktur des Moleküls eigentlich bestimmt, aber das System reagiert dann sehr empfindlich auf kleine Änderungen in den Winkeln. Deswegen führt man zur Stabilisation noch einen weiteren Winkel ein, der zwischen den zwei Ebenen liegt, die durch die Atome (i, j, k) und (j, k, l) gegeben sind, wenn (j, k, l) jeweils mit i gebunden sind. Die vier Atome (i, j, k, l) formen also ein Tetraeder. Man nennt diesen Winkel die uneigentliche Torsion Ψ . Alle Bindungen und Winkel sind natürlich translations- und rotationsinvariant. Bei Spiegelung vertauschen sich allerdings die Vorzeichen der Diederwinkel, so daß die Angabe dieser Vorzeichen die Chiralität vorgibt.

Mit diesen Vorbereitungen können die verschiedenen Anteile des CHARMM-Potentials angegeben werden:

CHARMM-Potential

$$U = U^{LJ} + U^C + U^{bonded} \quad (6.4)$$

$$U^{bonded} = U^b + U^\theta + U^\Phi + U^\Psi \quad (6.5)$$

$$U^b = \sum k_b (r - r^0)^2 \quad (6.6)$$

$$U^\theta = \sum k_\theta (\theta - \theta^0)^2 \quad (6.7)$$

$$U^\Phi = \sum k_\Phi (1 + \cos(n\Phi - \delta)) \quad (6.8)$$

$$U^\Psi = \sum k_\Psi (\Psi - \Psi^0)^2 \quad (6.9)$$

Das Potential setzt sich aus den gebundenen und den ungebundenen Anteilen zusammen. Die gebundenen Kräfte wirken immer auf die Atome, die nach obiger Beschreibung auch zu der Berechnung der relevanten Größen benutzt wurden. Bis auf die Diederwinkel werden für alle gebundenen Potentiale harmonische Ansätze verwendet, die die Größen auf kleine Abweichungen von den Gleichgewichtswerten beschränken. Die Form ist so gewählt, weil unter normalen Bedingungen, also nicht zu hohen Temperaturen, die Bindungslängen und die Winkel sehr steif sind und nur geringe Abweichungen von etwa 0.2Å bzw. 2° auftreten (siehe [21]). Das Diederpotential, welches die Torsion der Atome in einer Kette

bestimmt, ist periodisch im Winkel Φ und sehr glatt, so daß schon bei niedrigen Temperaturen Verdrehungen möglich sind. Der Parameter n gibt dabei die Periodizität an und die Phase δ bestimmt das Minimum der Funktion. Die uneigentliche Torsion dient wie oben schon erwähnt nur zur Stabilisation. Der ungebundenen Anteil besteht aus den schon hinreichend bekannten Lennard-Jones und Coulomb Potentialen, die zwischen allen Paaren von Atomen berechnet werden, die, wenn sie in einer Kette liegen, mindestens vier bzw. beim Lennard-Jones Potential in einigen Ausnahmen drei Bindungen voneinander entfernt sind. Die Wahl der Exponenten im Lennard-Jones Anteil geschieht aus den in Kapitel 4 genannten Gründen, wobei das CHARMM-Modell noch weitere Terme der Form $\left(\frac{\sigma}{r}\right)^E$ mit anderen Exponenten E beinhaltet, die hier nicht berücksichtigt werden. Die Wahl der ungebundenen Anteile beeinflusst allerdings die Lage der Minima und es ist bis jetzt nicht klar, welche Potentiale das richtige Verhalten wiedergeben.

Sämtliche in den Potentialanteilen vorkommenden Parameter sind in der Regel durch die Namen der an der Berechnung beteiligten Atome gegeben, unter Umständen auch abhängig von der chemischen Gruppe, in der sie sich befinden. Die zugrundeliegende Annahme, daß man diese Parameter unabhängig von dem Molekül, das man betrachtet, wählen kann, ist umstritten, aber bis jetzt die einzige Möglichkeit, ein Potentialmodell für eine große Gruppe von Molekülen anzugeben.

Ein Protein ist in dem CHARMM-Modell durch die Angabe seiner primären Struktur (zum Beispiel aus der *Protein Data Bank* [23]) und der zu den Atomnamen gehörigen Parameter (siehe [24]) ausreichend beschrieben, um Simulationen damit durchführen zu können.

6.3 Das Minimierungsverfahren

Die komplizierte Form des verwendeten Potentials macht eine Anwendung der in Kapitel 2 vorgestellten Verfahren in ihrer ursprünglichen Variante vollkommen unpraktikabel. So würde zum Beispiel die Glättung des vollständigen CHARMM-Potentials mithilfe der thermodynamischen Mittelung aus dem Gaussian Density Annealing dazu führen, daß die Bindungskräfte schwächer würden und deshalb die Bindungen nicht aufrechterhalten werden würden. Das Ergebnis sind vollkommen unrealistische Konfigurationen bei hohen Temperaturen, die auch durch Abkühlung nicht wieder in sinnvolle Strukturen des Moleküls übergehen. Auch die Anwendung des Basin-Hopping-Algorithmus ist bei der Größe der Proteine kaum möglich, da der Rechenaufwand für jeden Monte-Carlo Schritt enorm ist. Das liegt an der Form der gebundenen Potentialanteile, die bereits bei kleinen Abweichungen von den Gleichgewichtswerten sehr starke Kräfte zur Folge hat, so daß nur sehr kleine Schrittweiten für den Minimierungsalgorithmus in Frage kommen.

Für die Minimierung von Proteinen wurde in dieser Arbeit ein Ansatz verwendet, der auf folgenden Eigenschaften des CHARMM-Modells basiert: Wie oben schon erwähnt sind die Abweichungen von den Gleichgewichtswerten bei den Bindungslängen und Winkeln unter normalen Bedingungen nicht zu großer Temperaturen nur sehr gering. Deshalb betrachtet man diese Potentiale als Realisierung der Zwangsbedingungen, durch die die primäre Struktur des Moleküls aufrechterhalten wird. Die Potentialanteile U^b , U^θ und U^Ψ bleiben daher unverändert. Durch die Torsionswinkel Φ wird die lokale Struktur des Moleküls bestimmt, weswegen dieser Anteil eine wichtige Rolle bei der Minimierung einnimmt. Da

der Wert von n , der die Periodizität angibt, in den meisten Fällen eins ist, kann dieser Potentialanteil als ausreichend glatt angesehen werden und wird deshalb ebenfalls nicht verändert.

Die ungebundenen Kräfte bestimmen die globale Struktur des Moleküls und verhindern normalerweise die Minimierung der lokalen Struktur. Deshalb sollte als erster Schritt eine Minimierung durchgeführt werden, bei der Lennard-Jones- und Coulomb-Anteil unberücksichtigt bleiben, damit ein Minimum von U^{bonded} gefunden werden kann. Nach diesem Schritt werden wieder alle Kraftanteile berücksichtigt, wobei die ungebundenen zunächst geglättet werden, damit das Molekül nicht direkt in ein energetisch hohes Minimum fällt. Die Glättung des Lennard-Jones Potentials wird in Anlehnung der DEM mittels der Faltung mit einem Gauß-Kern der Breite t vorgenommen:

$$u^{LJ}(\mathbf{x}, t) = (2\pi t)^{-\frac{3N}{2}} \int U^{LJ}(\mathbf{y}) e^{-\frac{(\mathbf{y}-\mathbf{x})^2}{4t}} d\mathbf{y} \quad (6.10)$$

Für die Lösung wird U^{LJ} wieder durch zwei Gauß-Funktionen angenähert, deren Parameter a_k und b_k für $k = 1, 2$ aus Tabelle 4.1 entnommen werden können. Allerdings werden bei Proteinen Kräfte zwischen vielen verschiedenen Atomarten berechnet, so daß die Werte von ϵ_{ij} und σ_{ij} aus

$$U_{ij}^{LJ}(\vec{x}_i, \vec{x}_j) = 4\epsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right) \quad (6.11)$$

für jedes unterschiedliche Paar von Atomen neu bestimmt werden müssen. Die Näherung durch Gauß-Funktionen muß an die jeweiligen Werte angepaßt werden. Um aber nicht für jedes mögliche Paar von Atomarten, eigene Parameter a_k und b_k benutzen zu müssen, kann man so vorgehen, daß man das Potential wie folgt berechnet:

$$\tilde{U}_{ij}^{LJ} = \epsilon_{ij} \sum_{k=1}^L a_k e^{-b_k r_{ij}^2} \quad \text{mit} \quad r'_{ij} = \frac{r_{ij}}{\sigma_{ij}} \quad (6.12)$$

ϵ_{ij} gibt lediglich die tiefe des Minimums an und kann daher multipliziert werden, σ_{ij} bestimmt die örtliche Skala und wird daher durch die Ersetzung von r_{ij} durch r'_{ij} berücksichtigt. Insgesamt ergibt sich so das geglättete Lennard-Jones Potential

$$\tilde{u}_{ij}^{LJ} = \epsilon_{ij} \sum_{k=1}^L a_k (1 + 8b_k t)^{-\frac{3}{2}} e^{-\frac{b_k r_{ij}^2}{1+8b_k t}} \quad \text{mit} \quad r'_{ij} = \frac{r_{ij}}{\sigma_{ij}} \quad (6.13)$$

Dabei ergibt sich nun das Problem, daß durch die Glättung die örtliche Skala stark verändert wird. In Abbildung 6.1 sieht man das geglättete Lennard-Jones Potential zwischen zwei Atomen für verschiedene Glättungszeiten dargestellt. Bei $t = 2$ wirkt das Potential schon bis zu einem Abstand von ca. $r = 4$ stark abstoßend. Um bei der Auswertung der Potentiale aber Zeit zu sparen, sollen die kurzreichweitigen Kräfte ab einem gewissen Abstand der Atome nicht mehr ausgewertet werden, so daß nicht alle N^2 Paare berücksichtigt werden müssen. Deshalb ist es wichtig, daß das Potential immer oberhalb eines festen Abstandes ohne großen Fehler abgeschnitten werden kann. Um die örtliche Skala von \tilde{u}^{LJ} weitgehend unabhängig von t zu machen, kann man daher setzen:

$$\hat{u}_{ij}^{LJ} = \epsilon_{ij} \sum_{k=1}^L a_k (1 + 8b_k t)^{-\frac{3}{2}} e^{-\frac{b_k \hat{r}_{ij}^2}{1+8b_k t}} \quad \text{mit} \quad \hat{r}_{ij} = \frac{r_{ij}}{\sigma_{ij} \cdot (1 + 8b_1 t)} \quad (6.14)$$

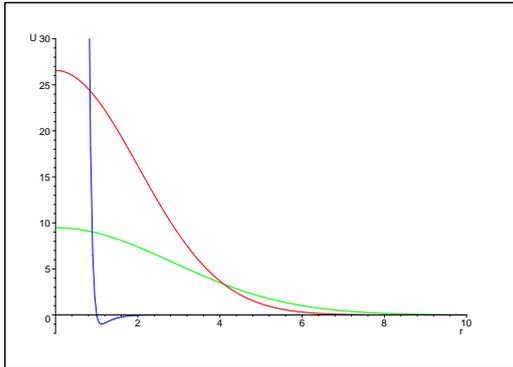


ABBILDUNG 6.1: \tilde{u}_{ij}^{LJ} für $t = 0$ (—), $t = 1$ (—) und $t = 2$ (—)

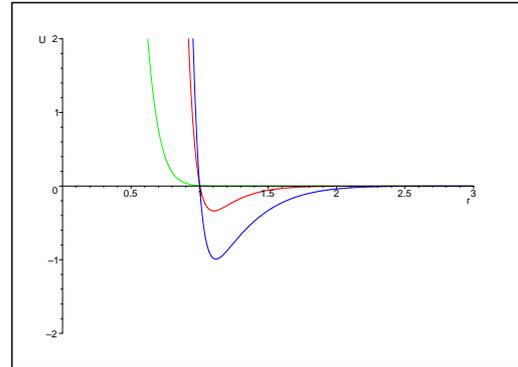


ABBILDUNG 6.2: \hat{u}_{ij}^{LJ} für $t = 0$ (—), $t = 0.01$ (—) und $t = 0.5$ (—)

Dadurch bleibt die Breite der ersten Gaußfunktion konstant, so daß der abstoßende Bereich nicht größer wird. Den Effekt sieht man in Abbildung 6.2. Die Funktion wird zwar immer noch geglättet, der abstoßend wirkende Bereich wird aber beibehalten oder sogar kleiner. So kann der Lennard-Jones Anteil zu jeder Zeit ohne großen Fehler mit einem konstanten Abschneideradius r_{cut} ausgewertet werden.

Für die Glättung des Coulomb-Anteils könnte man ebenfalls eine Faltung mit dem Gauß-Kern vornehmen, aber bei der Betrachtung großer Systeme wie Proteinen muß man wie oben beachten, daß eine schnelle Methode zur Auswertung des Potentials benutzt werden soll. Bei dem hier benutzten Programm wird die sogenannte P^3M -Methode von Hockney und Eastwood [26] verwendet. Dazu wird der langreichweitige $\frac{1}{r}$ Anteil in einen kurzreichweitigen und einen langreichweitigen aufgespalten:

$$\frac{1}{r} = \frac{1}{r} (1 - \operatorname{erf}(Gr) + \operatorname{erf}(Gr)) \quad (6.15)$$

$$= \frac{1}{r} (\operatorname{erfc}(Gr) + \operatorname{erf}(Gr)) \quad (6.16)$$

Dabei treten wieder die schon in Kapitel 5.2 beschriebene Fehlerfunktion und die komplementäre Fehlerfunktion auf. Diese Form des Potentials erhält man als Lösung der Poisson-Gleichung, wenn man in der durch Delta-Distributionen dargestellten Ladungsdichte für jedes Atom eine Gauß-Verteilung mit entgegengesetzter Ladung und eine mit gleicher Ladung hinzuaddiert, deren Mittelpunkte am Ort des Atoms liegen. Der Parameter G gibt die Breite dieser Verteilungen an und bestimmt die Reichweite des kurzreichweitigen Anteils. In Abbildung 6.3 sind die beiden Anteile zusammen mit der resultierenden $\frac{1}{r}$ -Funktion dargestellt. Der kurzreichweitige Anteil kann zusammen mit dem Lennard-Jones Potential ausgewertet werden und wird wie dieser ab einem festen Abstand r_{cut} vernachlässigt, während der langreichweitige Anteil auf einem Gitter gelöst wird (siehe [26]). Da dieser Teil des Potentials hinreichend glatt ist, bleibt er auch in dem hier benutzten Minimierungsalgorithmus unverändert. Der kurzreichweitige Anteil wird um einen von der Glättungszeit t abhängigen Faktor unterdrückt:

$$u_{ij}^{Ckr} = \lambda(t) \cdot U_{ij}^{Ckr} = \lambda(t) \cdot \frac{q_i q_j}{4\pi\epsilon_0} \operatorname{erfc}(Gr) \quad (6.17)$$

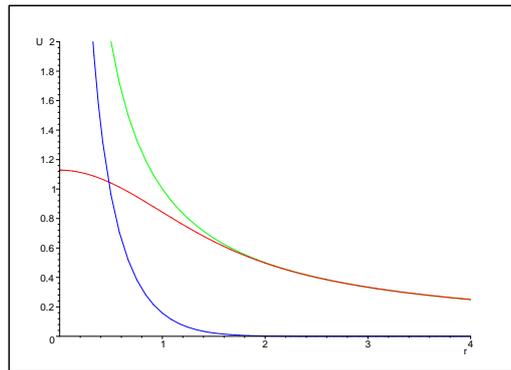


ABBILDUNG 6.3: Die Funktionen $\frac{\text{erfc}(Gr)}{r}$ (—), $\frac{\text{erf}(Gr)}{r}$ (—) und $\frac{1}{r}$ (—) mit $G = 1$

Der Nachteil der Auswertung durch die P^3M -Methode ist, daß sie eigentlich nur für periodische Probleme funktioniert. Hier soll aber ein einzelnes Molekül minimiert werden, so daß ein Fehler bei der Berechnung des langreichweitigen Coulomb-Potentials entsteht. Dieser Fehler wird durch die Vergrößerung der Simulationsbox verkleinert und dann vernachlässigt, da er mindestens um einen Faktor 100 kleiner ist als die gesamte Coulombenergie.

Damit kann der Gesamtalgorithmus zur Minimierung von Proteinen formuliert werden:

Protein-Minimierung

Schritt 1: Von einer zufälligen (aber sinnvollen) Anfangskonfiguration aus minimiere zunächst U^{bonded} , während U^{LJ} und U^{C} unberücksichtigt bleiben.

Schritt 2: Setze $t = 1.0$ und minimiere

$$u(t) = U^{\text{bonded}} + U^{\text{C}_{lr}} + \lambda(t) \cdot U^{\text{C}_{kr}} + \hat{u}^{\text{LJ}}(t) \quad (6.18)$$

mit

$$\lambda(t) = (1 - t)^2 \quad (6.19)$$

Schritt 3: Falls $t > 10^{-4}$ setze $t = \frac{t}{1.5}$ und gehe zu **Schritt 2**, sonst zu **Schritt 4**

Schritt 4: Minimiere

$$U = U^{\text{bonded}} + U^{\text{C}} + U^{\text{LJ}} \quad (6.20)$$

Für die Implementation dieses Algorithmus wurde als Grundlage ein bereits vorhandenes Programm benutzt, welches von Jan Hamaekers und Ralf Wildenhues¹ zur moleküldynamischen Simulation entwickelt wurde und auch ein lokales Minimierungsverfahren beinhaltet, das mit Conjugierten Gradienten arbeitet. In dieses Programm wurde der oben beschriebene Algorithmus integriert. Im nächsten Kapitel wird die Anwendung dieses Algorithmus auf BPTI beschrieben.

¹Abteilung Wissenschaftliches Rechnen und Numerische Simulation des Instituts für Angewandte Mathematik der Universität Bonn

6.4 Anwendung auf BPTI

BPTI (Bovine Pancreatic Trypsin Inhibitor) ist mit 861 Atomen eines der kleinsten Proteine und kommt im Nukleus der Zellen vor. Seine Aufgabe ist die Regelung der Proteinsynthese in der Zelle durch die Blockierung des Enzyms Trypsin. Abbildung 6.4 zeigt seine Konformation im kristallinen Zustand wie man sie aus der *Protein Data Bank* [23] erhält. Dieser Zustand stellt natürlich ein Energieminimum dar, ist aber nicht mit dem

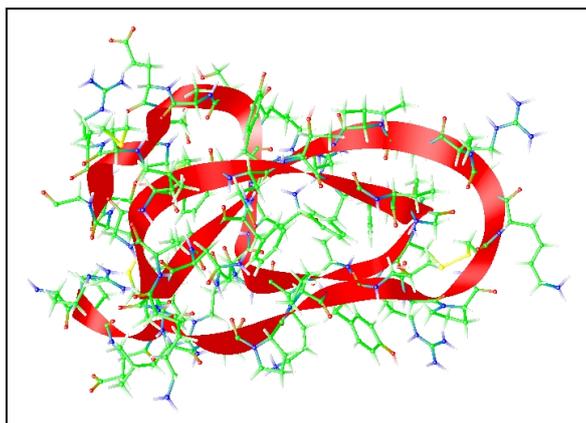


ABBILDUNG 6.4: BPTI im kristallinen Zustand

Grundzustand in einer wässrigen Umgebung identisch. Das Ziel der Minimierung sollte es sein, diesen Grundzustand zu finden, um Aussagen über die Konfiguration in der biologischen Umgebung des Proteins machen zu können. Man beschränkt sich aber meist noch darauf, nur das reine Molekül zu optimieren, da die Hinzunahme von Wasser das Problem erheblich erschwert. Eine andere Möglichkeit, die Effekte des umgebenden Wassers zu berücksichtigen, wäre die Hinzunahme eines weiteren Potentialanteils wie in [25], worauf hier aber zunächst verzichtet werden soll.

Um einen zufälligen Anfangszustand zu erzeugen, wurde ausgehend von der kristallinen Konfiguration eine moleküldynamische Rechnung bei einer konstanten Temperatur von etwa $T = 1500K$ durchgeführt. Bei dieser Temperatur brechen die Bindungen des Moleküls eigentlich, aber da das CHARMM-Modell nicht für diese Temperaturbereiche entwickelt wurde, bleibt das Molekül bei dieser Simulation trotzdem stabil. Die Temperatur wurde so hoch gewählt, um möglichst schnell einen entfalteten Zustand des Proteins zu erhalten. Abbildung 6.5 zeigt diesen Zustand, der dann als Anfangskonfiguration für den Minimierungsalgorithmus diente.

Abbildung 6.6 zeigt die Konfiguration nach der Minimierung der gebundenen Potentiale. Diese Minimierung wurde allerdings nach 50000 Schritten und 11h und 55min Rechenzeit beendet, ohne daß das Abbruchkriterium von

$$\max_{i \leq N} |\nabla_{\vec{x}_i} U| < 10^{-6} \quad (6.21)$$

erreicht wurde. Dieser Abbruch war nötig, weil die Torsionsenergie nur sehr langsam gegen ihr Minimum konvergiert, wie man in Abbildung 6.7 gut erkennen kann. Der Grund liegt in den sehr kleinen Schrittweiten, die durch die hohen Energien von U^b und U^θ bei kleinen

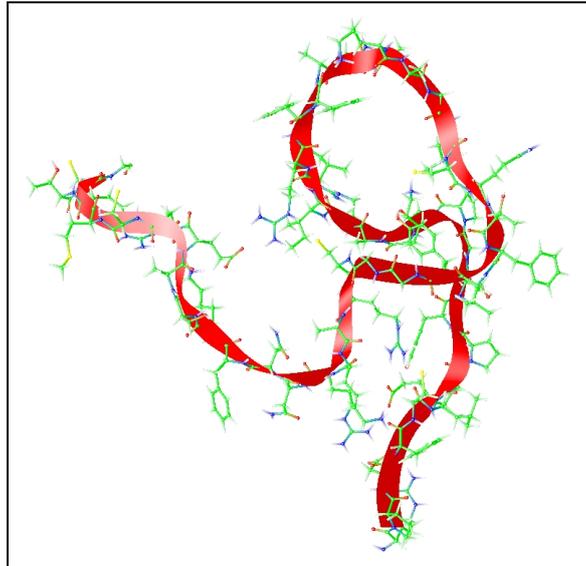
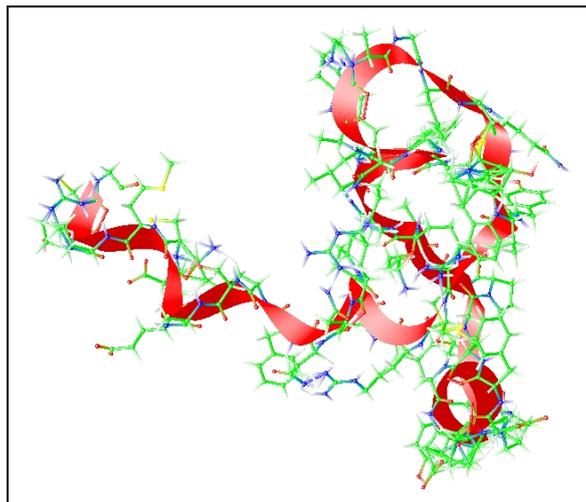


ABBILDUNG 6.5: Anfangszustand

ABBILDUNG 6.6: Minimum von U^{bonded}

Abweichungen von den Equilibriumswerten notwendig sind. Trotzdem sieht man, wie sich die lokale Struktur an die durch das Minimum der Torsionsenergie vorgegebene anpaßt und eine deutliche Spiralförmung entsteht. Die Energie der uneigentlichen Torsion ist etwa um einen Faktor 100 kleiner und spielt somit für die Struktur genau wie die anderen beiden Anteile der Bindungslängen und Winkel keine große Rolle.

Das Endergebnis der BPTI-Minimierung ist in Abbildung 6.8 zu sehen. Die lokale spiralförmige Struktur ist dabei wieder etwas vermindert worden, während sich die globale Struktur an die Lennard-Jones- und Coulomb-Energie angepaßt hat. Dadurch haben sich vor allem die unrealistisch kleinen Abstände der äußeren Atome normalisiert. Die gezeigte Konfiguration hat eine potentielle Energie von $U = -6.7559 \cdot 10^4 \text{kcal/mol}$. Diese Energie

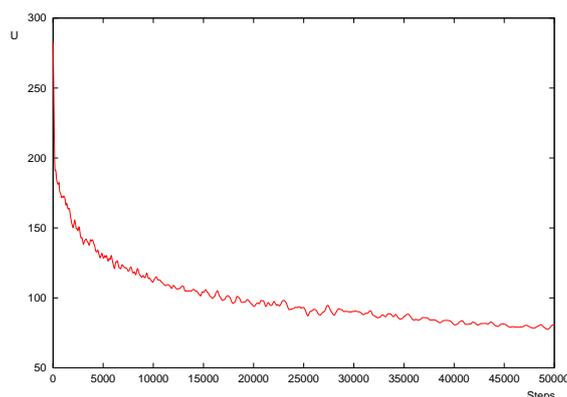


ABBILDUNG 6.7: Verlauf von U^{Φ} [kcal/mol]

ist allerdings mit einem großen Fehler behaftet, da der Parameter G für die P^3M -Methode nicht ideal gewählt wurde. Zum Vergleich wurde die Anfangskonfiguration 6.5 mit 20000

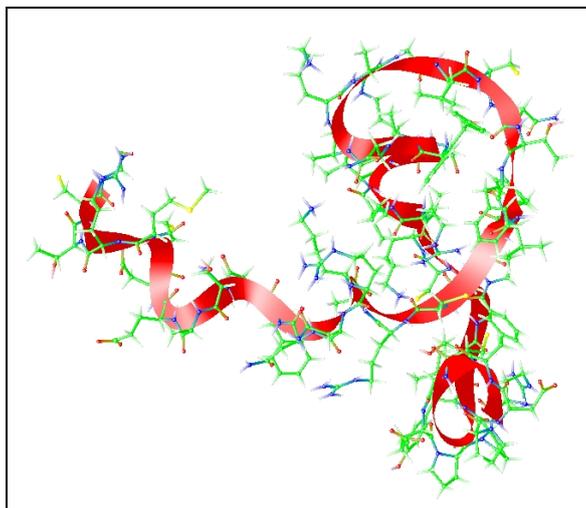


ABBILDUNG 6.8: Minimum von U

Schritten nur lokal minimiert, so daß die Konfiguration in Abbildung 6.9 mit der Energie $U = -6.8473 \cdot 10^4$ kcal/mol entstand. Hier hat sich die Struktur der Anfangskonfiguration kaum geändert. Der Vergleich mit dem Potentialwert des durch den vorgestellten Algorithmus berechneten Minimums überrascht etwas, da seine potentielle Energie höher liegt. Das liegt wahrscheinlich daran, daß keine der lokalen Minimierungen durchgeführt werden konnte, bis das Abbruchkriterium für den Betrag des Gradienten erreicht war. Der Grund ist wie oben schon beschrieben, die äußerst schlechte Konvergenz bei der Minimierung. Daher wurden bei der Minimierung von U jeweils nur 5000 Schritte durchgeführt, bevor das CG-Verfahren abgebrochen wurde. Zum Zeitpunkt des Abbruchs lag der Betrag des Gradienten bei etwa 10^{-3} . Da dieser Wert aber über eine lange Zeit konstant blieb, wäre eine Durchführung von wesentlich mehr Schritten notwendig gewesen. Diese Tatsache stellt ein großes Problem bei der Minimierung von komplexen Molekülen dar, die

durch das CHARMM-Potential beschrieben werden. Der enorme Rechenaufwand macht Proteine für Verfahren mit geglätteten Potentialfunktionen sehr unattraktiv. Hinzu kommt noch, daß die Glättung der ungebundenen Kraftanteile nicht in der Lage ist, die globale Struktur des Moleküls wesentlich zu ändern. Das zeigt, daß der vorgestellte Algorithmus

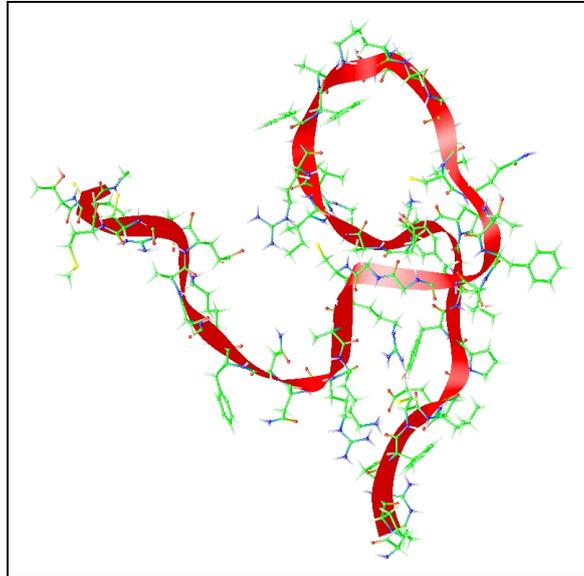


ABBILDUNG 6.9: *Minimum von U durch lokales Minimieren*

selbst dann nicht ausreichend ist, um die globale Struktur des Proteins zu optimieren, wenn man die Minimierungen bis zum Erreichen des Abbruchkriteriums durchführt. Zur Verbesserung wäre zum Beispiel eine Einbeziehung von vielen Anfangskonfigurationen möglich, die bereits so ausgewählt wurden, daß sie in energetisch sinnvollen Bereichen liegen. Solche Generierungsphasen für Anfangskonfigurationen sind in vielen Methoden zur Vorhersage der tertiären Struktur von Proteinen enthalten (siehe zum Beispiel [25]), denn das komplexe Zusammenspiel zwischen den lokalen Bindungen der Atome und der globalen Wechselwirkung der ungebundenen Atome macht es dem Minimierungsalgorithmus sehr schwer, vollkommen andere Punkte im Konformationsraum zu erreichen. Ein Ausweg könnte sein, die Positionen der Atome nicht nur durch ihre Ortsvektoren zu speichern, aus denen dann alle anderen Parameter berechnet werden, sondern umgekehrt das Molekül durch die Angabe aller Bindungslängen, Winkel und Diederwinkel zu bestimmen. Dieser Wechsel des Koordinatensystems macht es wesentlich leichter, die lokale Struktur zu minimieren, da viele der Variablen unabhängig voneinander sind, solange die ungebundenen Kräfte ausgeschaltet sind. Die Berechnung dieser ungebundenen Anteile wird allerdings durch die Benutzung der lokalen Variablen erschwert, was aber durch die Verbesserung der Minimierungseigenschaften durchaus in Kauf genommen werden könnte.

6.5 Zusammenfassung

Die Optimierung von biologischen Molekülen wie Proteinen stellt zur Zeit ein wichtiges Gebiet der Forschung dar, weswegen es eine Vielzahl von Ansätzen zur Lösung dieses

Problems gibt. Dabei ist aber stets zu beachten, daß selbst die Modelle, durch die die Proteine dargestellt werden, noch immer am Anfang ihrer Entwicklung stehen. Die gebundenen Potentialanteile des hier benutzten CHARMM-Modells resultieren aus einer Taylorentwicklung der Abweichung um die Equilibriumswerte (siehe [21]) und beschreiben daher das physikalische Verhalten nur in erster Näherung. Man kann deswegen nicht erwarten, daß das Modell alle physikalischen Eigenschaften des Moleküls gut genug wiedergibt, um genaue Vorhersagen aus den numerischen Berechnungen gewinnen zu können. Trotzdem ist die Entwicklung von Optimierungsverfahren eine wichtige Aufgabe, da jede Verbesserung der Modelle und der Verfahren beim Verständnis der biologischen Vorgänge helfen kann.

Die hier vorgestellte Methode zur Energieminimierung von Proteinen ist allein nicht ausreichend, um gute Kandidaten für energetisch tief liegende Konfigurationen zu liefern. Sie leidet vor allem unter dem enormen Rechenaufwand, der durch die schlechte Konvergenz der Minimierungsverfahren bei dem hier benutzten Potentialmodell begründet ist. Verbessert werden könnte das Verfahren durch eine Darstellung des Moleküls in den lokalen Variablen der Bindungen, Winkel und Diederwinkel, wodurch das Problem der im Moment schlechten Konvergenz behoben werden könnte. Eine andere Verbesserungsmöglichkeit könnte eine Analyse der verschiedenen Skalen des Gesamtpotentials liefern, die dann zum Beispiel durch die Verwendung eines Diffusionstensors D bei der Anwendung der DEM berücksichtigt werden könnte. So wäre eine unterschiedlich starke Glättung der verschiedenen Potentialanteile möglich, um die gebundenen Kräfte sinnvoll in die DEM miteinbeziehen zu können. Außerdem könnte ein Einfrieren der Bindungspotentiale von Vorteil sein, die im Moment durch ihre hohe Vibrationsfrequenz die sehr kleine Schrittweite bei der Minimierung erzwingen.

Kapitel 7

Zusammenfassung und Ausblick

Globale Minimierungsalgorithmen besitzen ein breites Spektrum von Anwendungen, weshalb eine große Anzahl von Ansätzen für solche Algorithmen entwickelt wurde. Die in dieser Arbeit betrachteten Verfahren benutzen alle eine durch eine Integraltransformation geglättete Potentialfunktion, welche das Resultat ganz unterschiedlicher Überlegungen ist. Bei der DEM resultiert die Transformation aus der Benutzung der Diffusionsgleichung, deren bekannte glättenden Eigenschaften die Motivation für ihre Benutzung darstellt. Bei den anderen Verfahren, wie QMA, GDA und EDP, steht am Anfang jedoch eine physikalische Beschreibung des Problems. Die generelle Ähnlichkeit aller Verfahren folgt aus der Benutzung der gleichen Annahme, daß das System näherungsweise durch voneinander unabhängige Atome beschrieben werden kann. Diese Annahme führt in den unterschiedlichen Zusammenhängen, wie der quantenmechanischen oder klassischen Betrachtung bzw. der Betrachtung der inneren oder freien Energie, zu ebenfalls unterschiedlichen Algorithmen, die aber dennoch bei näherer Betrachtung große Ähnlichkeit aufweisen. Die DEM läßt sich ebenfalls in diesem Zusammenhang sehen, auch wenn ihr durch den fehlenden physikalischen Hintergrund die wichtige Vorschrift zur Wahl der Glättungszeit fehlt. Ihr Funktionieren kann man als Folge der Tatsache verstehen, daß das mit der Einteilchen-Näherung thermodynamisch gemittelte Potential eine Lösung der Diffusionsgleichung darstellt, wenn man alle Einteilchenverteilungen auf dieselbe Breite beschränkt.

Eine Ausnahme in der Betrachtung stellt die Transformation des Potentials im Basin-Hopping-Algorithmus dar. Hier wird das Potential nicht geglättet, sondern so vereinfacht, daß nur Potentialwerte der Minima angenommen werden können. Diese Transformation würde den Minimierungsalgorithmus sehr mächtig machen, wenn sie analytisch ausgeführt werden könnte. Durch die Notwendigkeit der numerischen Durchführung beschränkt sich der Erfolg des Verfahrens auf einfache Probleme, bei denen der Rechenaufwand durch gute Konvergenz bei der Minimierungsprozedur akzeptabel bleibt.

Die Ähnlichkeit aller Verfahren bis auf den Basin-Hopping-Algorithmus erkennt man auch daran, daß ihre Resultate bei der Anwendung auf Lennard-Jones-Cluster beinahe identisch waren. Sie konnten zwar durch kleine Änderungen der ursprünglichen Algorithmen in der Regel verbessert werden, aber bei der Minimierung von Clustern mit mehr als 30 Atomen versagen alle Verfahren vollkommen, wenn man sich auf die Aufgabenstellung der globalen Minimierung bezieht. Eine Verbesserung der Verfahren kann man am offensichtlichsten erreichen, wenn man die in ihnen benutzten Näherungen so modifiziert, daß in den relevanten Bereichen eine gute Übereinstimmung mit der Zielfunktion erreicht

werden kann. Sind die Näherungen nämlich so schlecht, daß sie die für die Aufgabenstellung wichtigen Eigenschaften nicht wiedergeben können, so ist es offensichtlich, daß ihre Verbesserung sich stark auf die Qualität der Ergebnisse auswirken wird. Das Gaussian-Density-Annealing ist ein gutes Beispiel dafür, da die Gibbs/Boltzmann-Verteilung in Abhängigkeit von der Temperatur in viele Gebiete mit hoher Wahrscheinlichkeit unterteilt ist, was durch eine einfache Gauß-Funktion nicht wiedergegeben werden kann. Hier würde bereits eine Summe von Gauß-Verteilungen, wie sie bei der Packet-Annealing-Method benutzt wird, die Näherung entscheidend verbessern, führt aber zu großen Problemen bei der Berechenbarkeit der auftretenden Integrale und wurde daher bis heute nicht vollständig implementiert. Andere Möglichkeiten der Verbesserung liegen in der Verwendung eines neuen Ansatzes für die Lösung der Schrödingergleichung mit nicht-verschwindendem Potential oder in der Verallgemeinerung der in der DEM benutzten Diffusionsgleichung. Man muß aber auch dabei stets auf die gute Berechenbarkeit der resultierenden Potentialtransformationen und Bewegungsgleichungen achten, da ansonsten der Rechenaufwand schnell inakzeptabel wird.

Ein weiteres Gebiet wichtiger zukünftiger Forschung wird es sein, die vorgestellten Algorithmen auf komplexere Systeme wie zum Beispiel Proteine anzuwenden. Das ist in ihrer ursprünglich entwickelten Form nicht ohne weiteres möglich, da sie Zwangsbedingungen wie Bindungen in Molekülen, in welcher Form sie auch realisiert werden, bis jetzt nur schlecht berücksichtigen können. Hier könnten Methoden wie Multiskalenanalyse oder die Benutzung eines vorteilhafteren Koordinatensystems weitere Fortschritte bringen.

Auch die Potential-Modelle selber müssen weiterentwickelt werden, um den Resultaten der Minimierungsalgorithmen bessere Vorhersagekraft liefern zu können. Vielleicht gelingt sogar eine befriedigende Darstellung von Molekülen durch vereinfachte, sogenannte minimale Potentiale, bei denen nicht alle Atome des Moleküls zur Wechselwirkung beitragen, wodurch das Problem stark vereinfacht würde. Allerdings werden genauere Beschreibungen eher durch komplexere Modelle geliefert werden, so daß es eine stete Herausforderung darstellt, die existierenden Verfahren auf diese Modelle anwenden zu können.

Anhang A

Herleitung der Bewegungsgleichungen

A.1 Bewegungsgleichungen des GDA

Die thermodynamischen Mittelwerte berechnen sich klassisch durch folgendes Integral:

$$\langle A \rangle = \int A(\mathbf{x}) p_B(\mathbf{x}, \beta) d\mathbf{x} \quad (1.1)$$

mit

$$p_B = \frac{e^{-\beta U(\mathbf{x})}}{\int e^{-\beta U(\mathbf{x})} d\mathbf{x}} \quad (1.2)$$

Dann gilt für die Abhängigkeit von $\langle A \rangle$ von β offensichtlich folgendes:

$$\frac{d}{d\beta} \langle A \rangle = -\langle U A \rangle + \langle U \rangle \langle A \rangle \quad (1.3)$$

Setzt man dort die Erwartungswerte

$$\mathbf{x}^0 = \langle \mathbf{x} \rangle \quad (1.4)$$

$$\Lambda_{ii} = \langle (\vec{x}_i - \vec{x}_i^0)^2 \rangle \quad (1.5)$$

ein und wählt für die Wahrscheinlichkeitsverteilung den Ansatz,

$$p_{GDA}(\mathbf{x}, \beta) = \det^{-\frac{1}{2}}(2\pi\Lambda) e^{-\frac{1}{2}(\mathbf{x}-\mathbf{x}^0)\Lambda^{-1}(\mathbf{x}-\mathbf{x}^0)} \quad (1.6)$$

so erhält man die Bewegungsgleichungen:

$$\frac{d}{d\beta} \mathbf{x}^0 = -\langle U \mathbf{x} \rangle + \langle U \rangle \langle \mathbf{x} \rangle \quad (1.7)$$

$$= -\int U \mathbf{x} p_{GDA} d\mathbf{x} + \int U \mathbf{x}^0 p_{GDA} d\mathbf{x} \quad (1.8)$$

$$= -\int U(\mathbf{x} - \mathbf{x}^0) p_{GDA} d\mathbf{x} \quad (1.9)$$

$$= -\Lambda \nabla_{\mathbf{x}^0} \langle U \rangle \quad (1.10)$$

$$\frac{d}{d\beta}\Lambda = -\langle U(\mathbf{x} - \mathbf{x}^0)^2 \rangle + \langle U \rangle \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \quad (1.11)$$

$$= -\int U(\mathbf{x} - \mathbf{x}^0)^2 p_{GDA} d\mathbf{x} + \int U\Lambda p_{GDA} d\mathbf{x} \quad (1.12)$$

$$= -\int U [(\mathbf{x} - \mathbf{x}^0)^2 - \Lambda] p_{GDA} d\mathbf{x} \quad (1.13)$$

$$= -\Lambda^2 \nabla_{\mathbf{x}^0}^2 \langle U \rangle \quad (1.14)$$

A.2 Bewegungsgleichungen des QMA

Ausgangspunkt ist die Schrödingergleichung in imaginärer Zeit:

$$\frac{\partial}{\partial \tau} \Psi(\mathbf{x}, \tau) = H\Psi(\mathbf{x}, \tau) \quad \text{mit} \quad H = \frac{\hbar^2}{2m} \nabla^2 - U(\mathbf{x}) \quad (1.15)$$

Ist der Hamiltonoperator nicht explizit zeitabhängig, so läßt sich die Lösung in einen zeitabhängigen und einen nicht-zeitabhängigen Teil aufspalten:

$$|\Psi\rangle = e^{-H\tau} |\Phi\rangle \quad \text{bzw.} \quad \Psi(\mathbf{x}, \tau) = e^{-H\tau} \Phi(\mathbf{x}) \quad (1.16)$$

Dann gilt für die zeitliche Entwicklung des Erwartungswertes eines Operators folgendes:

$$\frac{d}{d\tau} \langle A \rangle = \frac{d}{d\tau} \frac{\langle \Psi | A | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \frac{d}{d\tau} \frac{\langle \Phi | e^{-H\tau} A e^{-H\tau} | \Phi \rangle}{\langle \Phi | e^{-2H\tau} | \Phi \rangle} \quad (1.17)$$

$$= -\langle HA + AH \rangle + 2\langle A \rangle \langle H \rangle \quad (1.18)$$

Setzt man dort die Erwartungswerte

$$\mathbf{x}^0 = \langle \mathbf{x} \rangle \quad (1.19)$$

$$\Lambda_{ii} = \langle (\vec{x}_i - \vec{x}_i^0)^2 \rangle \quad (1.20)$$

ein und wählt für die Wellenfunktion den Ansatz der Gaussian-Wave-Packets,

$$\Psi(\mathbf{x}, \tau) = \det^{-\frac{1}{4}}(2\pi\Lambda) e^{-\frac{1}{4}(\mathbf{x} - \mathbf{x}^0)\Lambda^{-1}(\mathbf{x} - \mathbf{x}^0)} \quad (1.21)$$

so erhält man die Bewegungsgleichungen:

$$\frac{d}{d\tau} \mathbf{x}^0 = -\langle H\mathbf{x} + \mathbf{x}H \rangle + 2\langle \mathbf{x} \rangle \langle H \rangle \quad (1.22)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* (\nabla^2 \mathbf{x} + \mathbf{x} \nabla^2 - 2\mathbf{x}^0 \nabla^2) \Psi d\mathbf{x} - 2\langle (\mathbf{x} - \mathbf{x}^0)U \rangle \quad (1.23)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* (2\nabla + 2\mathbf{x} \nabla^2 - 2\mathbf{x}^0 \nabla^2) \Psi d\mathbf{x} - 2\langle (\mathbf{x} - \mathbf{x}^0)U \rangle \quad (1.24)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* \left(-2\frac{\mathbf{x} - \mathbf{x}^0}{\Lambda} - \frac{(\mathbf{x} - \mathbf{x}^0)^3}{2\Lambda^2} \right) \Psi d\mathbf{x} - 2\langle (\mathbf{x} - \mathbf{x}^0)U \rangle \quad (1.25)$$

$$= 0 - 2\langle (\mathbf{x} - \mathbf{x}^0)U \rangle \quad (1.26)$$

$$= -2\Lambda \nabla_{\mathbf{x}^0} \langle U \rangle \quad (1.27)$$

$$\frac{d}{d\tau}\Lambda = -\langle H(\mathbf{x} - \mathbf{x}^0)^2 + (\mathbf{x} - \mathbf{x}^0)^2 H \rangle + 2\langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle H \rangle \quad (1.28)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* (\nabla^2 (\mathbf{x} - \mathbf{x}^0)^2 + (\mathbf{x} - \mathbf{x}^0)^2 \nabla^2 - 2\Lambda \nabla^2) \Psi \, d\mathbf{x} \quad (1.29)$$

$$- 2 (\langle (\mathbf{x} - \mathbf{x}^0)^2 U \rangle - \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle U \rangle) \quad (1.30)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* (2 + 2(\mathbf{x} - \mathbf{x}^0) \nabla + 2(\mathbf{x} - \mathbf{x}^0)^2 \nabla^2 - 2\Lambda \nabla^2) \Psi \, d\mathbf{x} \quad (1.31)$$

$$- 2 (\langle (\mathbf{x} - \mathbf{x}^0)^2 U \rangle - \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle U \rangle) \quad (1.32)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* \left(2 - 2 \frac{(\mathbf{x} - \mathbf{x}^0)^2}{\Lambda} + \frac{(\mathbf{x} - \mathbf{x}^0)^4}{2\Lambda^2} + 1 - \Lambda \frac{(\mathbf{x} - \mathbf{x}^0)^2}{2\Lambda^2} \right) \Psi \, d\mathbf{x} \quad (1.33)$$

$$- 2 (\langle (\mathbf{x} - \mathbf{x}^0)^2 U \rangle - \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle U \rangle) \quad (1.34)$$

$$= \frac{\hbar^2}{2m} \int \Psi^* \left(2 - 2 + \frac{1}{2} + 1 - \frac{1}{2} \right) \Psi \, d\mathbf{x} \quad (1.35)$$

$$- 2 (\langle (\mathbf{x} - \mathbf{x}^0)^2 U \rangle - \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle U \rangle) \quad (1.36)$$

$$= \frac{\hbar^2}{2m} - 2 (\langle (\mathbf{x} - \mathbf{x}^0)^2 U \rangle - \langle (\mathbf{x} - \mathbf{x}^0)^2 \rangle \langle U \rangle) \quad (1.37)$$

$$= \frac{\hbar^2}{2m} - 2\Lambda^2 \nabla_{\mathbf{x}^0}^2 \langle U \rangle \quad (1.38)$$

Literaturverzeichnis

- [1] J.E. Straub, *Optimization Techniques with Applications to Proteins*, in: New Developments in Theoretical Studies of Proteins, edited by Ron Elber, World Scientific, Singapore (1996), pp. 137-196.
- [2] I. Andricioaei, J.E. Straub, *Finding the needle in the haystack: algorithms for global optimization*, Computers in Physics 10 (1996), 449.
- [3] P. Amara, D. Hsu and J.E. Straub, *Global energy minimum searches using an approximate solution of the imaginary time Schroedinger equation*, J. Phys. Chem. 97 (1993), pp. 6715-6721.
- [4] I. Andricioaei, J.E. Straub, *Generalized simulated annealing algorithms using tsallis statistics: Application to conformational optimization of a tetrapeptide*, Physical Review E (1996), 53:R3055-R3058
- [5] D. Shalloway, *Variable-Scale Coarse-Graining in Macromolecular Global Optimization*, L. Biegler, T. Coleman, A. R. Conn, and F. Santosa, eds. Springer, New York (1997), pp. 135-161.
- [6] J. Kostrowicki, L. Piela, B. J. Cherayil, H.A. Scheraga, *Performance of the diffusion equation method in searches for optimum structures of clusters of Lennard-Jones atoms*, J. Phys. Chem. 95 (1991), pp. 4113-4119.
- [7] R.J. Wawak, M.M. Wimmer, H.A. Scheraga, *An application of the diffusion equation method of global optimization to water clusters*, J. Phys. Chem. 96 (1992), pp. 5138-5145.
- [8] D.J. Wales and J.P.K. Doye, *Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing Up to 110 Atoms*, Chem. Phys. Lett. 269 (1997), pp. 408-412.
- [9] D.J. Wales and M.P. Hodges, *Global minima of water clusters $(H_2O)_n, n \leq 21$, described by an empirical potential*, Chem. Phys. Lett., 286 (1998), pp. 65-72.
- [10] <http://www-wales.ch.cam.ac.uk/CCD.html>
- [11] M.W. Mahoney, W.L. Jorgensen, *A five-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions*, J. Chem. Phys., 112 (2000), 8910.

- [12] J. Weickert, *Anisotropic diffusion in image processing*, Teubner Verlag, Stuttgart, 1997
- [13] H. Vershelde, S. Schelstraete, J. Vandekerckhove, J.L. Vershelde, *An effective potential for calculating free energies. 1. General concepts and approximations*, J. Chem. Phys., 106 (1997), 1556.
- [14] S. Schelstraete, H. Vershelde, *Finding minimum energy configurations of Lennard-Jones clusters using an effective potential*, J. Phys. Chem. A101 (1997), 310.
- [15] Z. Wu, *The effective energy transformation scheme as a general continuation approach to global optimization with application to molecular conformation*, SIAM J. Optimization, 1993.
- [16] M. S. Gockenbach, A. J. Kearsley, W. W. Symes, *An Infeasible Point Method for Minimizing the Lennard-Jones Potential*, Technical Report 93-47, Department of Computational & Applied Mathematics, Rice University, Houston, TX 77005, December 1993.
- [17] M. Chaplin, *Water structure and behavior*, <http://www.sbu.ac.uk/water/index.html>.
- [18] K. Huang, *Statistical Mechanics*, John Wiley & Sons, New York, 1987.
- [19] J.E. Dennis, R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia, 1996.
- [20] R. Fletcher, C.H. Reeves, *Function minimization by conjugate gradients*, Comput.J., 7 (1964).
- [21] A. Neumaier, *Molecular Modeling of Proteins and Mathematical Prediction of Protein Structure*, SIAM Review, 39(1997):407-460.
- [22] M.R. Hoare, *Structure and dynamics of simple microclusters*, Adv. Chem. Phys. 40 (1979), pp. 49-135.
- [23] Research Collaboratory for Structural Bioinformatics PDB, <http://www.rcsb.org/pdb/>.
- [24] A.D. MacKerell, C. Brooks, L. Brooks, L. Nilsson, B. Roux, Y. Won, M. Karplus, The Encyclopedia of Computational Chemistry, chapter *CHARMM: The Energy Function and Its Parameterization with an Overview of the Program*, John Wiley & Sons, (1998), pp 271-277.
- [25] A. Azmi, R.H. Byrd, E. Eskow, R.B. Schnabel, S. Crivelli, *Predicting Protein Tertiary Structure using a Global Optimization Algorithm with Smoothing*, in C.A. Floudas, P.M. Pardalos, *Optimization in Computational Chemistry and Molecular Biology*, Kluwer Academic Publishers (2000).
- [26] R.W. Hockney and J.W. Eastwood, *Computer Simulation Using Particles*, Adam Hilger, Philadelphia (1988).