

Mehrzentren-Integrationsverfahren für Molekülstruktur-Rechnungen

Inaugural-Dissertation

zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

im Fachbereich Physik
der Universität Gesamthochschule Kassel

vorgelegt von
Dipl.-Phys. Andreas Heitmann
aus Wolfhagen

Kassel, den 05.06.2000

Als Dissertation vom Fachbereich Physik

angenommen am: 09.06.2000

Erster Gutachter: Prof. Dr. B. Fricke

Zweiter Gutachter: Priv. Doz. Dr. S. Fritzsche

Tag der mündlichen Prüfung: 03.07.2000

Inhaltsverzeichnis

1. Einleitung	3
2. Theorie der relativistischen Molekülstruktur	7
2.1. Die Dirac-Gleichung	7
2.2. Die Born-Oppenheimer-Näherung	8
2.3. Dichtefunktionaltheorie	9
2.4. Die Kohn-Sham-Gleichungen	10
2.5. Die Lokaldichte-Näherung	11
2.6. Nichtlokale Dichtefunktionale	12
2.7. Relativistische Dichtefunktionale	12
2.8. Basisentwicklung der KS-Orbitale	13
2.9. Symmetrieorbitale	14
2.10. Der Dichtefit	17
2.11. Die Frozen-Core-Näherung	19
3. Das Mehrzentren-Integrationsverfahren	21
3.1. Die Zellenfunktionen	22
3.1.1. Das Verfahren von Baerends	23
3.1.2. Die Zellenfunktion von Becke	23
3.1.3. Modifikationen der Becke-Funktion	25
3.1.4. Voronoi-Zellen	25
3.1.5. Die Zellenfunktionen von Delley	26
3.1.6. Die Zellenfunktionen von Stratmann et al.	26
3.2. Die Koordinatentransformationen	27
3.2.1. Kugelkoordinaten	28
3.2.2. Die Radialtransformation	28
3.2.3. Radialtransformationen für nicht-relativistische Wellenfunktionen	29
3.2.4. Radialtransformation für relativistische Wellenfunktionen	30
3.3. Aufteilung des dreidimensionalen Integrals	31
3.4. Numerische Integrationsregeln	32
3.4.1. Eindimensionale numerische Integrationsregeln	33
3.4.2. Zweidimensionale numerische Integrationsregeln für die Kugeloberfläche	38
3.4.3. Automatische Integration	40
3.5. Ausnutzung der Molekülsymmetrie	45
3.6. MULIN — Das Mehrzentren-Integrationsprogramm	46

4. Testrechnungen	49
4.1. Atomare Rechnungen: Rf	49
4.2. Dimere: Au ₂	53
4.2.1. Die Referenzrechnung	53
4.2.2. Vergleichsrechnungen für konstanten Abstand	56
4.3. Kleine Moleküle: RfCl ₄	64
4.3.1. Die Referenzrechnung	64
4.3.2. Vergleichsrechnungen für konstanten Abstand	64
4.4. Cluster: Au@Au ₉	69
5. Zusammenfassung und Ausblick	75
A. Physikalische Konstanten und Definitionen	79
B. Details zu den Rechnungen	80
B.0.1. Trapezregel für periodische Integranden	80
C. Das Integrationsprogramm	81
C.1. Beschreibung der Eingabe-Datei	81
C.1.1. Eingabe	81
C.1.2. Erzeugung der numerischen Integrationsregel	82
C.1.3. Ausgabe	83
C.1.4. Diagnose und Fehlersuche	84
C.1.5. Interpolation der numerischen Radialfunktionen	85
C.1.6. Zellenfunktion	85
C.1.7. Symmetrie	87
C.1.8. Parameter der Integrationszentren	87
C.1.9. Transformation der intrinsischen Koordinaten auf Kugelkoordinaten	89
C.1.10. Numerische Integrationsregeln in den intrinsischen Koordinaten	91
C.1.11. Anzahl der Stützstellen der numerischen Integrationsregel	92
C.1.12. Variation der Zellengröße bei heteronuklearen Systemen	93
C.1.13. Testfunktionen	93
C.2. Eingabe-Datei für atomare Rechnungen	95
C.3. Eingabe-Datei für lineare Moleküle	96
C.4. Eingabe-Datei für dreidimensionale Moleküle	97
Literaturverzeichnis	99

1. Einleitung

Wie in vielen Bereichen der Naturwissenschaft hat sich die enorm gestiegene Leistungsfähigkeit verfügbarer Rechnersysteme auch auf die Möglichkeiten der theoretischen Beschreibung der Eigenschaften von Molekülen und Festkörpern ausgewirkt. In der Chemie und Biologie hat die Entwicklung von Software zur Berechnung von Moleküleigenschaften den Computer zu einem gängigen Werkzeug zur Vorhersage und Modellierung gemacht. Die Nanostrukturwissenschaften benutzen Molekulardynamik-Programme zur zeitabhängigen Simulation großer Systeme und in der Kristallographie werden Berechnungen der Bandstruktur von Kristallen erfolgreich eingesetzt.

Ein bisher nicht befriedigend gelöstes Problem ist jedoch das Skalenverhalten quantenmechanischer Methoden. Molekulardynamik-Simulationen berücksichtigen die quantenmechanischen Eigenschaften der Atome meist nur durch die verwendeten Potentiale, die im Verlauf der Rechnung jedoch nicht variiert werden. Die Elektronenstruktur wird zugunsten der Rechenzeit vollkommen vernachlässigt.

Auf der anderen Seite können atomare *ab-initio* Berechnungen auch die Spektren angeregter Atome sehr gut reproduzieren, benötigen dafür allerdings schon beträchtliche Rechnerkapazitäten.

Dazwischen liegt das Gebiet der Molekülphysik und der Kristallstruktur. Eine quantenmechanische Beschreibung der elektronischen Struktur erfordert neue mathematische Methoden, um den hohen Rechenaufwand zu begrenzen. Diese Entwicklungen wurden im Jahre 1998 mit dem Nobelpreis für Chemie honoriert. Er wurde an Walter Kohn für die Entwicklung der Dichtefunktionaltheorie (DFT) vergeben, die auch die Grundlage unserer Programme bildet, und an John Pople für die Algorithmen zur praktischen Umsetzung der DFT in Computerprogrammen.

Die exakte Lösung des Vielteilchen-Problems ist schon für einfachste Systeme unzugänglich. Die Dichtefunktionaltheorie bildet eine Alternative zu den herkömmlichen Lösungsverfahren wie Multikonfigurations-, CI- oder Coupled-Cluster-Rechnungen. Beide Vorgehensweisen sind im Prinzip exakt, bei den Multikonfigurations-Methoden nähert man sich der exakten Lösung durch systematische Vergrößerung der Basissätze, bei der DFT versucht man, das universelle Energiefunktional anzunähern, welches die Beiträge der Vielteilchen-Austausch- und Korrelations-Wechselwirkung enthält. Während bei Multikonfigurations-Rechnungen die Zahl der benötigten Basisfunktionen sehr schnell ansteigt und die Größe der berechenbaren Systeme begrenzt, konnte für die DFT gezeigt werden, dass durch die Verwendung moderner Dichtefunktionale [1, 2] eine hohe Genauigkeit bei der Berechnung von Grundzustands-Eigenschaften erreicht werden kann, wobei die Dichtefunktionale auch den Bereich großer Moleküle mit schweren Kernen abdecken können [3].

Die Dichtefunktionaltheorie überführt die molekulare Vielteilchen-Wellenfunktion in ein

äquivalentes Produkt aus Einteilchen-Wellenfunktionen. Um diese Molekülorbitale darzustellen, wählt man eine endliche Basis aus numerischen oder analytischen atomaren Orbitalen. Die so angenäherte Lösung der Dirac-Gleichung wird dadurch in eine Matrix-Eigenwert-Gleichung überführt [4].

Die Elemente der Matrix müssen durch die Integration über den \mathbb{R}^3 berechnet werden. Die spezielle Form der Integranden lässt dabei zu, dass ein Teil der Mehrzentren-Integrale analytisch ausgewertet werden könnte, falls die atomaren Orbitale als durch Gauß-Funktionen (GTO) dargestellt werden. Molekülstruktur-Programme wie GAUSSIAN [5] nutzen diese Eigenschaft der GTOs aus. Der Vorteil der einfachen Berechnung von Integralen von Gauß-Funktionen wird jedoch mit umfangreichen Basissätzen erkaufte, da insbesondere in Kernnähe solche Funktionen das singuläre Verhalten der Lösung der Dirac-Gleichung nicht wiedergeben. Für viele Integrale, insbesondere für die Matrixelemente der Austausch- und Korrelations-Potentiale lassen sich außerdem mit GTOs keine geschlossenen Ausdrücke angeben. Hier ist es in jedem Fall notwendig, eine numerische Integration durchzuführen.

Das in unserer Arbeitsgruppe entwickelte Molekülstrukturprogramm RELMOS erhebt den Anspruch, auch für schwere Elemente gute Ergebnisse für Molekülgeometrie und Bindungsenergien zu liefern. Statt die relativistische Struktur der Dirac-Gleichung nur mit Hilfe von Spin-Bahn-Korrekturen zur Schrödinger-Gleichung oder in der skalar-relativistischen Näherung zu behandeln, arbeitet es mit vierkomponentigen Spinor-Wellenfunktionen.

Die Dirac-Gleichung wird dabei mit Hilfe der relativistischen Lokaldichte-Näherung (RLDA) vereinfacht, und in einer numerischen Basis von atomaren Orbitalen dargestellt. Da die Basisfunktionen ebenfalls voll relativistisch berechnet werden, sind sie dem Problem gut angepasst, man kommt mit einer geringen Anzahl von Basisfunktionen aus [6].

Die Rechenzeit des Verfahrens skaliert etwa mit der dritten Potenz der Zahl der beteiligten, aktiven Elektronen. Beim Übergang zu größeren Systemen wächst die benötigte Rechenzeit sehr schnell an. Verschiedene Ansätze der Beschleunigung der Berechnungen wurden bereits in dem Programm implementiert, darunter die Frozen-Core-Näherung [7] und die Embedding-Methode für große Cluster [8].

Der größte Teil der Rechenzeit des Programms wird dafür benötigt, die Werte der Matrixelemente zu bestimmen, also numerische Integrationen über den \mathbb{R}^3 auszuführen. Die große Zahl der Matrixelemente erlaubt es dabei nicht, für jedes Integral eine eigene Integrationsregel aufzustellen. Wir verwenden daher eine gemeinsame Regel für alle vorkommenden Integrale. Diese Integrationsregel wird vor dem Start der eigentlichen Rechnung durch das Programm von Baerends et al. [9] aufgestellt. Der Algorithmus wurde zuerst im Amsterdamer Dichtefunktional Programm ADF [10] implementiert und ersetzte das ursprüngliche Diophantine-Verfahren [11, 12]. Das Baerends-Verfahren erzeugt eine verhältnismäßig hohe Anzahl von Integrationspunkten im Vergleich zu neueren Methoden, die in der Literatur vorgestellt werden.

Da die Anzahl der zu berechnenden Matrixelemente für eine Rechnung von der numerischen Integrationsregel unabhängig ist, die Rechenzeit aber fast linear in der Zahl der Funktionsauswertungen für die Berechnung der Integranden ist, skaliert die Rechenzeit unseres Programms ebenfalls linear in der Zahl der Integrationspunkte. Eine Einsparung von Integrationspunkten bei gleicher numerischer Genauigkeit führt also direkt zu einer besseren Handhabbarkeit von Rechnungen an größeren Molekülen.

Aus dieser Problematik leitet sich folgende Aufgabenstellung ab: zunächst sollen die bis-

her aus der Literatur bekannten Methoden der Mehrzentren-Integration für unser Programm implementiert werden. Dazu ist es notwendig, eine Software zu erstellen, die die vielfältigen numerischen und analytischen Verfahren [13–24] in einer möglichst allgemeinen Art und Weise umsetzt. Die nächste Aufgabe ist dann, aus diesen unterschiedlichen Verfahren dasjenige auszuwählen, welches für unsere Problemstellungen die besten Ergebnisse liefert. Abschließend soll die neue Methode an verschiedenen typischen Systemen getestet werden.

Die Arbeit ist im wesentlichen in drei Abschnitte aufgeteilt. In einem ersten Kapitel werden die theoretischen Grundlagen unseres Molekülstrukturprogramms dargestellt. Das nächste Kapitel beschreibt die neue Methode der Mehrzentren-Integration und im letzten Kapitel werden Rechnungen mit der bisherigen Integrationsmethode und der neuen Integration ausgeführt, um deren Handhabung in praktischen Anwendungen zu zeigen und einen Vergleich der beiden Verfahren zu ermöglichen.

2. Theorie der relativistischen Molekülstruktur

In diesem Kapitel wird gezeigt, wie durch verschiedene Vereinfachungen die Dirac-Gleichung in eine Form gebracht wird, die eine näherungsweise Berechnung der Lösungen mit dem Computer erlaubt. Alle in diesem Kapitel vorgestellten Methoden sind in unserem Programm zur relativistischen Molekülstrukturberechnung RELMOS implementiert.

2.1. Die Dirac-Gleichung

Die Fragestellung nach der quantenmechanischen, relativistischen Molekülstruktur führt zunächst auf die zeitunabhängige Dirac-Gleichung [25]

$$\hat{H}\Psi = E\Psi. \quad (2.1)$$

Die molekulare Wellenfunktion Ψ_0 hängt sowohl von den elektronischen Koordinaten r_i , als auch von den Kernkoordinaten R_a ab. Der relativistische Hamilton-Operator \hat{H} setzt sich zusammen aus der kinetischen Energie der Elektronen und Kerne und den verschiedenen Wechselwirkungspotentialen:

$$\hat{H} = \sum_a \hat{T}_a + \sum_i \hat{t}_i + \hat{V}_{ee} + \hat{V}_{eK} + \hat{V}_{KK} \quad (2.2)$$

Dabei ist $\hat{T}_a(R_a)$ der Operator der kinetischen Energie des a -ten Kerns, $\hat{t}_i(r_i)$ der entsprechende Operator für das i -te Elektron. Sie lauten in der relativistischen Darstellung:

$$\begin{aligned} \hat{t}_i &= \boldsymbol{\alpha} \cdot \hat{\mathbf{p}}_i + c^2 m_e \boldsymbol{\beta} \\ \hat{T}_a &= \boldsymbol{\alpha} \cdot \hat{\mathbf{p}}_a + c^2 M_a \boldsymbol{\beta}, \end{aligned} \quad (2.3)$$

wobei $\boldsymbol{\alpha}$ und $\boldsymbol{\beta}$ die bekannten Dirac-Matrizen, \mathbf{p} der dreidimensionale Impulsoperator und m_e bzw. M_a die Masse des entsprechenden Teilchens sind.

Die Potentiale werden durch die elektrostatische Coulomb-Wechselwirkung dargestellt. Im einzelnen ist dies der Operator der Kern-Kern-Abstoßung

$$V_{KK} = \frac{1}{2} \sum_{\substack{a,b \\ a \neq b}} \frac{Z_a Z_b}{|R_a - R_b|}, \quad (2.4)$$

der Kern-Elektron-Anziehung

$$V_{eK} = - \sum_a \sum_i \frac{Z_a}{|\mathbf{R}_a - \mathbf{r}_i|} \quad (2.5)$$

und der Elektron-Elektron-Abstoßung

$$V_{ee} = \frac{1}{2} \sum_{\substack{i, j \\ i \neq j}} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}, \quad (2.6)$$

bei der eine Doppelsumation durch den Vorfaktor $1/2$ wieder kompensiert wird.

Die Energieeigenwerte des relativistischen Hamilton-Operators sind nicht nach unten beschränkt [26]. Für positive Kernladungen existieren diskrete gebundene Elektronenzustände mit positiver Energie sowie kontinuierliche Spektren ungebundener Zustände mit $E > m_e c^2$ und $E < -m_e c^2$. Eine mit den Beobachtungen verträgliche Interpretation der Energiezustände liefert die Dirac'sche Löcher-Theorie [27]. Dabei wird angenommen, dass im Vakuumzustand alle Elektronenzustände mit negativer Energie besetzt und die mit positiver Energie unbesetzt sind.

Statt alle Zustände zu betrachten, bei denen der Erwartungswert des Ladungsoperators die Zahl der Elektronen im System liefert, wird nur der Unterraum der Zustände ohne virtuelle Elektron-Positron-Anregungen benutzt (sog. No-Pair-Näherung [28–30]). Bei unseren molekularen Rechnungen wird die No-Pair-Näherung implizit angewandt, indem atomare No-Pair-Zustände als Basisfunktionen benutzt werden [6]. In atomaren Rechnungen können auch mit Hilfe von Projektionsoperatoren die Zustände negativer Energie durch die Wahl der Randbedingungen herausprojiziert werden.

Die Lösung Ψ_0 der Dirac-Gleichung zur niedrigsten Energie E_0 wird Grundzustandswellenfunktion genannt und ist Gegenstand der weiteren Überlegungen.

2.2. Die Born-Oppenheimer-Näherung

Die Born-Oppenheimer-Näherung [31] separiert die Kernbewegung von der Elektronenbewegung. Dabei werden die Kerne als klassische geladene Teilchen betrachtet, in deren elektrostatischem Potential sich die Elektronen bewegen. Die Kernkoordinaten \mathbf{R}_a sind fixiert, so dass sie nur noch parametrisch im Hamilton-Operator (Gleichungen 2.4 und (2.5)) und in dessen Eigenwerten $E(\mathbf{R}_1, \dots, \mathbf{R}_N)$ auftreten. Dadurch vereinfacht sich die Dirac-Gleichung für das Molekül erheblich.

Die Näherung wird durch das Verhältnis der Masse der Kerne zur Elektronenmasse gerechtfertigt, der durch den Parameter $\xi = (m_e/M_a)^{1/4}$ ausgedrückt werden kann. Für typische zweiatomige Systeme in der Nähe des Grundzustandes liegt ξ in der Größenordnung 0.1, die Korrektur der Potentialenergieflächen skaliert mit ξ^4 , liegt also in der Größenordnung 10^{-4} .

Die Bedeutung der Born-Oppenheimer-Näherung liegt in der Möglichkeit, die Molekülstruktur durch die Minima der Energiefunktion $E(\mathbf{R}_a)$ beschreiben zu können. In den letzten

Jahren ist in unserem Molekülstrukturprogramm durch die gleichzeitige Berechnung der analytischen Energiegradienten $\partial E(\mathbf{R}_1, \dots, \mathbf{R}_N)/\partial \mathbf{R}_i$ die Suche und Charakterisierung der Minima von E erheblich erleichtert worden [32].

2.3. Dichtefunktionaltheorie

Die entscheidende Vereinfachung der Schrödinger bzw. Dirac-Gleichung wird durch die Dichtefunktionaltheorie (DFT) erreicht. Mit ihr ist es möglich, die Vielteilchen-Gleichung in eine effektive Einteilchen-Gleichung zu überführen.

Die Grundlage der DFT bildet das Hohenberg-Kohn-Theorem [33]. In der ursprünglichen, nicht-relativistischen Form gilt es für den nicht-entarteten Grundzustand von Vielteilchen-Systemen mit einer Zweiteilchen-Wechselwirkung und einem lokalen Potential. Das Theorem stellt einen eindeutigen funktionalen Zusammenhang zwischen der Vielteilchen-Wellenfunktion des Grundzustandes $\Psi_0(\mathbf{r})$, dem Potential $v(\mathbf{r})$ und der Elektronendichte $\rho(\mathbf{r})$ her:

$$\rho(\mathbf{r}) \longleftrightarrow v(\mathbf{r}) \longleftrightarrow \Psi_0(\mathbf{r})$$

Da $\Psi_0(\mathbf{r})$ ein eindeutiges Funktional der Dichte ist, können auch alle anderen Observablen als Funktional der Dichte geschrieben werden. Insbesondere betrifft dies die kinetische Energie $T[\rho]$, die Elektron-Elektron-Energie $W[\rho]$ und die totale Energie $E[\rho]$. Hohenberg und Kohn konnten außerdem zeigen, dass für ein gegebenes externes Potential $v(\mathbf{r})$ ein Funktional $E_v[\rho]$ existiert, für das die Grundzustands-Energie E_0 eine untere Schranke darstellt:

$$\begin{aligned} E_v[\rho] &= \langle \Psi[\rho] | \hat{T} + \hat{W} + \hat{V} | \Psi[\rho] \rangle \\ &= T[\rho] + W[\rho] + \int \rho(\mathbf{r})v(\mathbf{r})d^3r \geq E_0. \end{aligned} \quad (2.7)$$

Üblicherweise fasst man \hat{T} und \hat{W} zu einem Funktional $F[\rho]$ zusammen:

$$F[\rho] = T[\rho] + W[\rho] \quad (2.8)$$

Die Elektron-Elektron-Wechselwirkung $W[\rho]$ kann aufgeteilt werden in einen klassischen Coulomb-Anteil $J[\rho]$

$$J[\rho] = \frac{e^2}{2} \int \int \frac{\rho(\mathbf{r}_i)\rho(\mathbf{r}_j)}{|\mathbf{r}_i - \mathbf{r}_j|} d^3r_i d^3r_j, \quad (2.9)$$

und einen Restterm, der Austausch-Korrelations-Funktional $E^{\text{XC}}[\rho]$ genannt wird:

$$W[\rho] = J[\rho] + E^{\text{XC}}[\rho]. \quad (2.10)$$

Falls die Funktionale $T[\rho]$ und $W[\rho]$ bekannt sind, kann die totale Energie des Systems nun auf einfache Weise bestimmt werden, da das Vielteilchen-Problem reduziert wird auf das Auffinden derjenigen Funktion $\rho(\mathbf{r})$, welche $E_v[\rho]$ minimiert. Das Hohenberg-Kohn-Theorem wurde später für entartete Grundzustände verallgemeinert [34] und relativistische Systeme [35] verallgemeinert.

2.4. Die Kohn-Sham-Gleichungen

Die Schwierigkeit, geeignete Funktionale für $T[\rho]$ und $W[\rho]$ zu finden, führte zur Idee von Kohn und Sham (KS) [36], das Vielteilchen-Problem durch ein effektives Einteilchen-Problem zu ersetzen.

Dazu wird das Funktional $F[\rho]$ aus Gleichung (2.8) wie folgt aufgeteilt:

$$F[\rho] = J[\rho] + T^{\text{KS}}[\rho] + E^{\text{XC}}[\rho] \quad (2.11)$$

Der erste Term beschreibt die Coulomb-Energie der Ladungsverteilung, $T^{\text{KS}}[\rho]$ stellt die kinetische Energie des Einteilchen-Systems dar und der Rest-Term $E^{\text{XC}}[\rho]$ enthält alle darüber hinausgehenden Wechselwirkungen und wird Austausch-Korrelations-Funktional genannt.

Fordert man nun, dass die KS-Dichte $\rho^{\text{KS}}(\mathbf{r})$ exakt gleich der Dichte $\rho(\mathbf{r})$ des wechselwirkenden Vielteilchen-Problems ist, dann ist durch das HK-Theorem gewährleistet, dass auch die Grundzustands-Wellenfunktionen Ψ_0 und Ψ^{KS} und damit auch die Grundzustands-Energien übereinstimmen.

Um den Grundzustand aufzufinden, muss das Energiefunktional $E_v[\rho]$ aus Gleichung (2.7) unter der Nebenbedingung konstanter Elektronenzahl

$$\int \delta\rho(\mathbf{r})d^3r = 0 \quad (2.12)$$

minimiert werden. Eine notwendige Bedingung dafür ist, dass die erste Variation von $E_v[\rho]$ verschwindet. Drückt man das Energiefunktional durch die KS-Terme aus Gleichung (2.11) aus, so erhält man die Variationsgleichung

$$\int \delta\rho(\mathbf{r}) \left\{ v(\mathbf{r}) + \frac{\delta J[\rho]}{\delta\rho(\mathbf{r})} + \frac{\delta E^{\text{XC}}[\rho]}{\delta\rho(\mathbf{r})} + \frac{\delta T^{\text{KS}}[\rho]}{\delta\rho(\mathbf{r})} \right\} d^3r = 0. \quad (2.13)$$

Die ersten drei Terme können als effektives Einteilchen-Potential $v_{\text{eff}}(\mathbf{r})$ interpretiert werden, in dem sich die nicht-wechselwirkenden „Elektronen“ des Kohn-Sham-Systems bewegen. Definiert man das Austausch-Korrelations-Potential $v^{\text{XC}}(\mathbf{r})$ als Funktionalableitung von $E^{\text{XC}}[\rho]$ nach der Dichte, kann man schreiben

$$v_{\text{eff}}(\mathbf{r}) = v(\mathbf{r}) + e^2 \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + v^{\text{XC}}(\mathbf{r}). \quad (2.14)$$

Die Grundzustands-Dichte kann nun gefunden werden, indem zuerst die Einteilchen-Schrödinger-Gleichung¹ für das KS-System

$$\left(-\frac{1}{2m_e} \nabla^2 + v_{\text{eff}}(\mathbf{r}) \right) \psi_i^{\text{KS}} = \epsilon_i \psi_i^{\text{KS}} \quad (2.15)$$

gelöst wird. Die Gleichungen (2.15) werden Kohn-Sham-Gleichungen genannt und müssen iterativ gelöst werden, da das effektive Potential nichtlinear von den Einteilchen-Funktionen ψ_i^{KS}

¹In [36] wurde ausschließlich der nichtrelativistische Operator T^{KS} der kinetischen Energie behandelt.

abhängt. Wenn N Elektronen zur Verfügung stehen, werden damit die untersten Zustände besetzt. Da wir die Spin-Zustände des Elektrons nicht berücksichtigen, enthält jedes besetzte KS-Orbital ψ^{KS} zwei Elektronen und der Ausdruck für die Grundzustands-Dichte lautet

$$\rho = 2 \sum_i^{N/2} \left| \psi_i^{\text{KS}} \right|^2, \quad (2.16)$$

wobei die Orbitale aufsteigend nach den zugehörigen Einteilchen-Energien ϵ_i geordnet sind.

Die totale Energie $E[\rho]$ des Grundzustandes lässt sich ohne Zuhilfenahme des Funktionals der kinetischen Energie $T^{\text{KS}}[\rho]$ berechnen, indem man die Lösung der KS-Gleichungen verwendet. Dazu vergleicht man die Energiefunktionale des KS- und des wechselwirkenden Systems. Das Energiefunktional des KS-Systems besteht nur aus der kinetischen Energie des Einteilchen-Systems und dem externen Kernpotential:

$$E_v^{\text{KS}}[\rho] = T^{\text{KS}}[\rho] + \int \rho(\mathbf{r}) v(\mathbf{r}) d^3r \quad (2.17)$$

Der Unterschied besteht also in den Termen der Elektron-Elektron-Wechselwirkung:

$$E_v[\rho] = E_v^{\text{KS}}[\rho] + J[\rho] + E^{\text{XC}}[\rho] \quad (2.18)$$

Aus Gleichung (2.15) erhält man eine Beziehung, in der $T^{\text{KS}}[\rho]$ durch die Einteilchen-Energien ausgedrückt werden kann:

$$2 \sum_i^{N/2} \epsilon_i = T^{\text{KS}}[\rho] + \int \rho(\mathbf{r}) v_{\text{eff}}(\mathbf{r}) d^3r \quad (2.19)$$

Damit ergibt sich die totale Energie durch Einsetzen der Gleichungen (2.9), (2.14), (2.19) und (2.17) in Gleichung (2.18):

$$E_v[\rho] = 2 \sum_i^{N/2} \epsilon_i - e^2 \int \int \frac{\rho(\mathbf{r}) \rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' + E^{\text{XC}}[\rho] + \int \rho(\mathbf{r}) v^{\text{XC}}(\mathbf{r}) d^3r \quad (2.20)$$

Die Grundzustands-Energie des Vielteilchen-Systems kann somit bestimmt werden, falls ein Austausch-Korrelations-Funktional $E^{\text{XC}}[\rho]$ und dessen Funktionalableitung nach der Dichte $v^{\text{XC}}(\mathbf{r})$ zur Verfügung steht. Dieses Funktional enthält Beiträge der Vielteilchen-Wechselwirkung. Trotzdem hat es sich gezeigt, dass es schon mit der einfachsten Approximation, der Lokaldichte-Näherung, möglich ist, gute Ergebnisse zu erhalten.

2.5. Die Lokaldichte-Näherung

In der Lokaldichte-Näherung (LDA) wird für das Austausch-Korrelations-Funktional angenommen, dass es durch ein lokales Potential $\epsilon_{\text{xc}}(\rho)$ dargestellt werden kann [36]:

$$E^{\text{XC}}[\rho] = \int \rho(\mathbf{r}) \epsilon_{\text{xc}}(\rho(\mathbf{r})) d^3r. \quad (2.21)$$

Das Potential $\epsilon_{xc}(\rho(\mathbf{r}))$ hängt dann nur noch implizit vom Ort \mathbf{r} ab. Mit diesem Ansatz lässt sich die Funktionalableitung für das effektive Einteilchen-Potential (2.14) einfach berechnen, sie hängt nur von $\rho(\mathbf{r})$ ab:

$$v^{\text{XC}}(\mathbf{r}) = \frac{\delta E^{\text{XC}}[\rho]}{\delta \rho(\mathbf{r})} = \frac{d(\rho(\mathbf{r})\epsilon_{xc}(\rho(\mathbf{r})))}{d\rho(\mathbf{r})}. \quad (2.22)$$

Das einfachste Potential für $v^{\text{XC}}(\mathbf{r})$ stammt von Slater [37] und enthält nur den Beitrag für die Austausch-Wechselwirkung. Es ist unter der Annahme eines homogenen Elektronengases der Dichte ρ abgeleitet und lautet:

$$v^{\text{XC}}(\mathbf{r}) = -3(3\rho(\mathbf{r})/8\pi)^{1/3} \quad (2.23)$$

Spätere Herleitungen von Gaspar [38] und Kohn und Sham [36] enthielten einen zusätzlichen Faktor $2/3$. Von Slater selbst wurde ein Vorfaktor α als empirischer Parameter eingeführt, der an die experimentellen Daten angepasst wurde [39]. Es hat sich gezeigt, dass im $X\alpha$ -Modell geeignete Werte für α abhängig vom zu berechnenden System sind², gute Ergebnisse erhält man im Allgemeinen mit $\alpha = 0.7$. Dieser Wert wird in unseren Rechnungen durchgehend verwendet, wenn das Slater-Funktional eingesetzt wird.

2.6. Nichtlokale Dichtefunktionale

Bereits von Hohenberg und Kohn [33] wurde die Inhomogenität der Ladungsdichte als Störung zum Modell des homogenen Elektronengases behandelt. Die Abweichung wird als Gradientenentwicklung der Dichte im Energiefunktional ausgedrückt. Der Einsatz solcher Funktionale erfordert die Berechnung höherer Ableitungen der Dichte. In unserem Programm wird die Auswertung des Gradienten der molekularen Dichte auf die Differentiation der atomaren Basisfunktionen zurückgeführt, was die numerische Differentiation der Radialfunktionen und die Berechnung der zugehörigen Vektorkugelfunktionen erfordert. Der zusätzliche Rechenaufwand für die gradientenkorrigierten Funktionale ist während der SCF-Iteration nicht notwendig, da die Korrekturen normalerweise sehr klein sind, und wird daher erst am Ende der SCF-Schleife durchgeführt.

2.7. Relativistische Dichtefunktionale

Da die Herleitung des Hohenberg-Kohn-Theorems weitgehend unabhängig von der speziellen Form des Hamiltonians ist, gilt es auch für den relativistischen Dirac-Hamilton-Operator. Dabei ist allerdings nicht mehr die Dichte ρ die Grundgröße, sondern die relativistisch kovariante Viererstromdichte j^μ . Die Grundlagen der relativistischen Dichtefunktionaltheorie wurden von Rajagopal und Callaway [35] erarbeitet. Auch hier wird die Lokaldichte-Näherung (RLDA) verwendet, um das Austausch-Korrelations-Funktional zu approximieren.

²Eine Diskussion über die Wahl des α -Parameters findet sich in [40]

Das einfachste relativistische Energiefunktional ist wiederum das Slater'sche $X\alpha$ -Funktional, welches aus der Lagrange-Dichte für das relativistische, homogene Elektronengas hergeleitet werden kann [41].

Die relativistische Erweiterung besteht lediglich darin, die Einteilchen-Orbitale als vierkomponentigen Spinoren zu schreiben und die Operatoren der kinetischen Energie in Gleichung (2.3) durch die entsprechenden vierdimensionalen Matrizen zu ersetzen, das eigentliche Energiefunktional (2.21) ändert sich dabei nicht.

Eine Herleitung des Austauschpotentials aus der QED in der ersten Ordnung der Störungsentwicklung in α führt auf komplexe Ausdrücke für das Austausch-Korrelationspotential, die transzendente Funktionen enthalten und hier nicht angegeben werden sollen. Mögliche Erweiterungen der RLDA, wie z. B. die relativistische Gradienten-Entwicklung (RGGA) werden in [42] diskutiert. Eine Untersuchung des Einflusses der Gradientenkorrekturen in relativistischen, molekularen Systemen findet sich in [32].

2.8. Basisentwicklung der KS-Orbitale

Ein vollständiger Satz atomarer Basisfunktionen wäre bereits hinreichend, um die molekularen Wellenfunktionen zu beschreiben. Dies lässt sich aber in numerischen Rechnungen mit endlichen atomaren Basissätzen bezüglich eines festen Raumpunktes nicht realisieren, da sehr hohe Drehimpulse nötig wären, um die an den Orten aller Kerne stark variierenden molekularen Wellenfunktionen gut zu beschreiben. Daher verwendet man eine nicht-orthogonale Basis, die aus einer endlichen Anzahl atomarer Basisfunktionen zusammengesetzt ist. Die atomaren Funktionen sind an den entsprechenden molekularen Kernpositionen zentriert und werden Single-Site Orbitals (SSOs) genannt [14, 43–46].

Stellt man die unbekanntenen, exakten Lösungen Ψ_i der Einteilchen-Gleichungen (2.15) mit den Energieeigenwertern E_i als Linearkombinationen

$$\psi_i^{\text{KS}}(\mathbf{r}) = \sum_j c_{ij} \chi_j(\mathbf{r}) \quad (2.24)$$

einer endlichen Menge von Basisfunktionen $\{\chi_i(\mathbf{r})\}$ dar und minimiert den Erwartungswert des Operators $(\hat{H} - \epsilon)$, so erhält man Wellenfunktionen, deren Energieerwartungswerte ϵ_i eine obere Schranke für E_i darstellen. Die Minimierung der Energie durch Variation der Koeffizienten der Basisfunktionen unter der Nebenbedingung, dass deren Norm erhalten bleibt, nennt man Rayleigh-Ritz-Verfahren. Es führt auf die Säkulargleichung

$$\sum_j \underbrace{\langle \chi_k | \hat{H}^{\text{KS}} | \chi_j \rangle}_{H_{kj}} c_{ij} = \epsilon_i \sum_j \underbrace{\langle \chi_k | \chi_j \rangle}_{S_{jk}} c_{ij}, \quad (2.25)$$

in der die Überlappmatrix für die nicht-orthogonalen Basisfunktionen $\{\chi_i(\mathbf{r})\}$ vorkommt.

Die Variation der Einteilchen-Orbitale wird durch die Basisentwicklung (2.24) auf die Variation der Koeffizienten c_{ij} zurückgeführt. Definiert man die diagonale Matrix ϵ der Energieeigenwerte als $\epsilon_{ij} = \epsilon_i \delta_{ij}$, so lässt sich die Gleichung in übersichtlicher Form in Matrixschreibweise

darstellen:

$$\mathbf{H}\mathbf{c} = \epsilon\mathbf{S}\mathbf{c} \quad (2.26)$$

Zur Lösung dieses verallgemeinerten Eigenwert-Problems wird zunächst eine Cholesky-Zerlegung der symmetrischen, reellen Überlappmatrix $\mathbf{S} = \mathbf{L}^T\mathbf{L}$ durchgeführt, mit der Gleichung (2.26) auf eine gewöhnliche Eigenwert-Gleichung transformiert wird

$$\begin{aligned} \tilde{\mathbf{H}}\tilde{\mathbf{c}} &= \epsilon\tilde{\mathbf{c}} \\ \tilde{\mathbf{c}} &= \mathbf{L}\mathbf{c} \\ \tilde{\mathbf{H}} &= (\mathbf{L}^T)^{-1}\mathbf{H}\mathbf{L}^{-1}, \end{aligned} \quad (2.27)$$

die mit Standardverfahren gelöst werden kann [47, 48]. Die mit \mathbf{L} transformierten Basisvektoren $\tilde{\chi}_i = \sum_j L_{ij}\chi_j$ werden in der Literatur auch als „Natural Orbitals“ bezeichnet [49–51], da \mathbf{S} in dieser Basis diagonal ist. Die Transformation \mathbf{L} ist nicht eindeutig, sondern hängt vom Verfahren der Orthogonalisierung ab.

Die Minimierung der totalen Energie durch die Variation der Einteilchen-Orbitale lässt sich bei Molekülen nur mit sehr starken Vereinfachungen des Potentials und der Basisfunktionen analytisch ausführen. Näherungsweise Lösungen kann man zum Beispiel für Kristallgitter erhalten, bei denen die periodischen Randbedingungen die Verwendungen ebener Wellen als Basis für die Kohn-Sham-Orbitale zulassen, wenn man einfache Potentiale wie z. B. das Muffin-Tin-Potential benutzt. Dieses Verfahren wurde als Multiple-Scattering-Methode (MS) bekannt [52].

Ein numerisches Lösungsverfahren für die Kohn-Sham-Gleichung wurde erstmals von Ellis und Painter mit der Diskreten Variations-Methode (DVM) vorgeschlagen [53, 54]. Dabei werden analytische, lokalisierte Basisfunktionen verwendet und die zur Berechnung der Matrixelemente H_{jk} und S_{jk} notwendigen Integrationen numerisch ausgeführt.

Um die Basis möglichst klein zu halten, muss man dafür Sorge tragen, dass die Basisfunktionen dem Problem gut angepasst sind. In unserem Programm wird dazu eine numerische, atomare Basis verwendet, die von einem externen relativistischen Atomstrukturprogramm erzeugt wird³. Für die inneren Schalen stimmen die AOs sehr gut mit den KS-Orbitalen überein, für die äußeren Schalen kann eine vergrößerte Basis mit Zusatzfunktionen verwendet werden.

2.9. Symmetrieorbitale

Die atomare Basis, in der die Matrixelemente der \mathbf{S} - und \mathbf{H} -Matrix berechnet werden, besteht aus einem Satz von vierkomponentigen Dirac-Spinoren $|n\kappa m\rangle$, welche Lösungen der atomaren Dirac-Fock-Gleichung in der Lokaldichte-Näherung sind. Da das atomare Problem in einem kugelsymmetrischen externen Potential gelöst wird, faktorisiert die Lösung in ein Produkt aus Radial- und Winkelanteil:

$$\phi_{an\kappa m}(\mathbf{r})\langle r | an\kappa m \rangle = \frac{1}{r} \begin{pmatrix} P_{n\kappa}(r_a) & \mathcal{Y}_{\kappa m}(\theta_a, \phi_a) \\ iQ_{n\kappa}(r_a) & \mathcal{Y}_{-\kappa m}(\theta_a, \phi_a) \end{pmatrix} \quad (2.28)$$

³Das Atomstrukturprogramm wurde von Ellis und Ros [44, 55] entwickelt und von unserer Arbeitsgruppe um relativistische Dichtefunktionale erweitert.

Die Funktionen f und g bilden den Radialanteil der großen und kleinen Komponenten. Der Index a deutet an, dass sich die Radial- und Winkel-Koordinaten im molekularen System auf den Ort R_a des a -ten Kerns beziehen:

$$\mathbf{r}_a = \mathbf{r} - \mathbf{R}_a \quad (2.29)$$

Der Winkelanteil kann durch die Spinor-Kugelfunktionen \mathcal{Y}_κ^m ausgedrückt werden [56], wobei die Spin-Quantenzahl $s = 1/2$ hier unterdrückt wurde. Die κ -Quantenzahl beschreibt den Gesamt-Drehimpuls und kann die Werte

$$\kappa = \begin{cases} -(j + 1/2) = -(l + 1) & \text{für } j = l + 1/2 \\ j + 1/2 = l & \text{für } j = l - 1/2 \end{cases} \quad (2.30)$$

annehmen. Die Quantenzahl m nimmt wie gewöhnlich die Werte $m = -j, \dots, +j$ an. Die Spinor-Kugelfunktionen können durch Drehimpulskopplung

$$\mathcal{Y}_\kappa^m(\theta, \phi) = \sum_{m_s} \begin{pmatrix} l & s & j \\ m - m_s & m_s & m \end{pmatrix} Y_{l, m - m_s}(\theta, \phi) |s m_s\rangle \quad (2.31)$$

als Summe von gewöhnlichen Kugelfunktionen dargestellt werden. Da der Elektronenspin den Wert $s = 1/2$ hat, ist dies ein zweikomponentiger Vektor mit der Basis $|1/2, \pm 1/2\rangle$.

Die Anzahl der zu berechnenden Matrixelemente der Fock- und Überlappmatrix in Gleichung (2.25) ist im Allgemeinen jeweils N^2 , wobei N die Anzahl der Basisvektoren χ_i ist. Aufgrund gruppentheoretischer Überlegungen kann jedoch durch eine spezielle Wahl der Basis, den sog. Symmetrieorbitalen (SO), das Verschwinden bestimmter Matrixelemente vorhergesagt werden, falls die Anordnung der Kerne eine Symmetrie besitzt. Die Symmetrieorbitale werden dabei als Linearkombination der atomaren Orbitale (LCAO) geschrieben, die Koeffizienten werden durch die Projektionsoperator-Technik [57–60] bestimmt.

Zur Bestimmung der Symmetrieorbitale können wir zunächst ohne Beschränkung der Allgemeinheit annehmen, dass die Positionen $\{R_a\}$ aller Kerne durch die Symmetrieoperationen g der molekularen Symmetrie-Gruppe G ineinander überführt werden:

$$\mathbf{R}_b = g\mathbf{R}_a \quad (2.32)$$

Auch sollen die atomaren Basisfunktionen $\phi_{an\kappa m}(\mathbf{r})$ für alle Atome äquivalent sein. Falls das Molekül inäquivalente Kerne besitzt, müssen dieses in Äquivalenzklassen eingeteilt und folgenden Überlegungen für jede Äquivalenzklasse getrennt durchgeführt werden.

Die Wirkung der Symmetrieoperation g im Raum der kartesischen Koordinaten \mathbf{r} des Moleküls wird im Raum der Funktionen $f(\mathbf{r})$ durch einen Operator $T(g)$ dargestellt:

$$T(g)f(\mathbf{r}) = f(g^{-1}\mathbf{r}) \quad (2.33)$$

Speziell für die Komponenten $\varphi_{anlm}(\mathbf{r}) := R_{anl}(|\mathbf{r}_a|)Y_{lm}(\mathbf{r}_a)$ der atomaren Basisfunktionen gilt dann:

$$\begin{aligned} T(g)R_{anl}(|\mathbf{r}_a|)Y_{lm}(\mathbf{r}_a) &= R_{anl}(|g^{-1}\mathbf{r} - \mathbf{R}_a|)Y_{lm}(g^{-1}\mathbf{r} - \mathbf{R}_a) \\ &= R_{anl}(|\mathbf{r} - g\mathbf{R}_a|)Y_{lm}(g^{-1}(\mathbf{r} - g\mathbf{R}_a)) \\ &= R_{bnl}(|\mathbf{r}_b|)Y_{lm}(g^{-1}\mathbf{r}_b) \end{aligned} \quad (2.34)$$

Die Kugelfunktionen $\{Y_{lm}\}$ zu einem festen l bilden eine Basis für die Darstellung der Gruppe G , deren Darstellungsmatrizen hier mit $D_{m'm}^{(l)}(g)$ bezeichnet werden sollen:

$$T(g)Y_{lm}(\mathbf{r}) = \sum_{m'} D_{m'm}^{(l)}(g)Y_{lm'}(\mathbf{r}) \quad (2.35)$$

Eine explizite Form der Matrizen als Funktion der Euler-Winkel α, β, γ der Rotation g ist z. B. in [59] angegeben. Setzt man dies in Gleichung (2.34) ein, so tritt die Transformation g nur noch in der Darstellungsmatrix auf:

$$T(g)\varphi_{anlm} = \sum_{m'} D_{m'm}^{(l)}(g)\varphi_{bnlm'} \quad (2.36)$$

Dass die atomaren Funktionen eine Basis der Darstellung $T(g)$ bilden, erkennt man, wenn man eine weitere Summation über alle Zentren einfügt und dadurch die neue Darstellungsmatrix $\mathbf{d}^{(l)}$ definiert:

$$T(g)\varphi_{anlm} = \sum_{m'a'} \mathbf{d}_{m'm,a'a}^{(l)}(g)\varphi_{a'nlm'} \quad (2.37)$$

Falls die Darstellung $\mathbf{d}^{(l)}$ irreduzibel ist, lässt sich keine Vereinfachung durch gruppentheoretische Überlegungen durchführen. Falls die Darstellung aber reduzibel ist, kann sie symbolisch als direkte Summe von irreduziblen Darstellungen geschrieben werden

$$\mathbf{d}^{(l)}(g) = \bigoplus_j n_l^{(j)} \Gamma^{(j)}(g), \quad (2.38)$$

wobei der Vorfaktor $n_l^{(j)}$ die Vielfachheit des Auftretens der Darstellung $\Gamma^{(j)}$ in $\mathbf{d}^{(l)}$ beschreibt. Nun muss eine Basis für die irreduziblen Darstellungen $\Gamma^{(j)}$ gefunden werden. Dies wird mit der Projektionsoperator-Methode erreicht. Unter Zuhilfenahme des zweiten Schur'schen Lemmas [57] kann man zeigen, dass der Operator

$$P_{ii}^{(j)} = \frac{s^j}{n_G} \sum_g \Gamma_{ii}^{(j)}(g)T(g) \quad (2.39)$$

Funktionen aus dem Darstellungsraum von $\mathbf{d}^{(l)}$ in den Raum der Darstellung $\Gamma^{(j)}$ projiziert. Darin ist s^j die Dimension der Darstellung $\Gamma^{(j)}$ und n_G die Dimension der Symmetrie-Gruppe.

Setzt man nun die Matrixdarstellung aus Gleichung (2.37) in den Projektionsoperator ein, so erhält man die Basisvektoren des j -ten Unterraums, indem man den Projektor (2.39) auf alle Basisfunktionen φ_{1nlm} des ersten Atoms anwendet und die linear unabhängigen Funktionen auswählt. Die so erhaltene Symmetrie-Basis $\{\chi\}$ ist dann eine Linearkombination der atomaren Orbitale:

$$\chi_{iln}^{(j)} = \sum_{am} S_{ailnm}^{(j)} \varphi_{anlm} \quad n = 1, \dots, n_l^{(j)} \quad (2.40)$$

Die Symmetriekoeffizienten $S_{ailnm}^{(j)}$ können direkt aus der Matrixdarstellung von $\mathbf{d}^{(l)}$ und $\mathbf{\Gamma}^{(j)}$ berechnet werden. Die Symmetriebasis ändert ihre Eigenschaften nicht, wenn die Koeffizientenmatrix zeilenweise mit einem konstanten Vorfaktor multipliziert wird, und kann daher normiert werden:

$$\sum_{am} S_{ailnm}^{(j)} S_{ailn'm}^{(j)} = \delta_{nn'} \quad (2.41)$$

Die Berechnung der Symmetriekoeffizienten wird durch das externe Programm TSYM ausgeführt [59, 60], welches für eine vorgegebene Kernanordnung und Quantenzahlen l, m der atomaren Orbitale die Symmetrie-Gruppe bestimmt und mit der Projektionsoperator-Methode die Matrix $S_{ailnm}^{(j)}$ errechnet.

2.10. Der Dichtefit

Zur Bestimmung des elektronischen Potentials $V(\mathbf{r})$ aus der Elektronendichte $\rho(\mathbf{r})$ muss die Poisson-Gleichung

$$\Delta V(\mathbf{r}) = -4\pi\rho(\mathbf{r}) \quad (2.42)$$

gelöst werden. Falls $\rho(\mathbf{r}')/|\mathbf{r} - \mathbf{r}'|$ eine integrable Funktion für alle \mathbf{r} ist und hinreichend schnell gegen null geht für $|\mathbf{r}'| \rightarrow \infty$, so lautet eine Lösung der Poisson-Gleichung

$$V(\mathbf{r}) = \iiint \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r'. \quad (2.43)$$

Dies bedeutet, dass für jeden Punkt, an dem das Potential bestimmt werden soll, eine Integration über den gesamten \mathbb{R}^3 ausgeführt werden muss und führt bei der Berechnung von Energien der Form

$$E = \iiint V(\mathbf{r})\rho(\mathbf{r})d^3r \quad (2.44)$$

auf Sechsfach-Integrale, deren Bestimmung einen großen Rechenaufwand erfordern würde. Einen Ausweg bietet die Einschränkung des Lösungsraums auf Potentiale, die nicht aus beliebigen Dichten, sondern aus einer Linearkombination von atomaren Basisfunktionen erzeugt werden [7, 45, 61–64]. Eine solche Linearkombination wird als Modelldichte $\tilde{\rho}$ bezeichnet.

Da die Poisson-Gleichung linear in ρ ist, lässt sich die Lösung $V(\mathbf{r})$ der Poisson-Gleichung auch als lineares Funktional der Dichte $V(\mathbf{r}) = V[\rho; \mathbf{r}]$ auffassen. Man kann deshalb eine Basis $\{\tilde{\rho}_i\}$ wählen, und $V(\mathbf{r})$ als Linearkombination der Funktionen $V[\tilde{\rho}_i; \mathbf{r}]$ ausdrücken. Dies führt zu einer erheblichen Vereinfachung, falls $V[\tilde{\rho}_i; \mathbf{r}]$ leicht berechnet werden kann.

In unserem Programm werden die kugelsymmetrischen atomaren Elektronendichten $\tilde{\rho}_i(R) = f_i(R)^2 + g_i(R)^2$ benutzt, das zugehörige Potential $\tilde{v}_i(R)$ kann direkt angegeben werden [65]:

$$\tilde{v}_i(R) = -4\pi \int_0^\infty \frac{\tilde{\rho}_i(R')}{|R - R'|} dR' \quad (2.45)$$

Da die Funktionen $\tilde{\rho}_i(R)$ an den radialen Stützstellen tabelliert sind, wird $\tilde{v}_i(R)$ an den selben Stellen tabelliert, Zwischenwerte können dann durch Interpolation berechnet werden. Damit lautet die Modelldichte und das daraus resultierende Modellpotential:

$$\tilde{\rho}(\mathbf{r}) = \sum_i c_i \tilde{\rho}_i(|\mathbf{r} - \mathbf{r}_i|) \quad (2.46)$$

$$\tilde{V}(R) = \sum_i c_i v_i(|\mathbf{r} - \mathbf{r}_i|). \quad (2.47)$$

Nun stellt sich die Frage, welche Modelldichte am besten geeignet ist, um $\rho(\mathbf{r})$ anzunähern. Die nächstliegende Möglichkeit ist, die mittlere Abweichung der Modelldichte von der SCF-Dichte zu minimieren. Dabei wird man im Allgemeinen fordern, dass das Integral über die Modelldichte mit der Elektronenzahl übereinstimmt, was durch den Lagrange-Parameter μ berücksichtigt wird:

$$F[\tilde{\rho}] = \int (\rho - \tilde{\rho})^2 d^3r + \mu(N - \int \tilde{\rho}(\mathbf{r}) d^3r) \quad (2.48)$$

Um $F[\tilde{\rho}]$ zu minimieren, muss man nach den Koeffizienten c_i der Modelldichte variieren und erhält:

$$0 = \int (\rho(\mathbf{r}) - \tilde{\rho}(\mathbf{r})) \tilde{\rho}_j(|\mathbf{r} - \mathbf{r}_j|) d^3r - \mu \int \tilde{\rho}_j(|\mathbf{r} - \mathbf{r}_j|) d^3r, \quad (2.49)$$

Dieses Gleichungssystem wurde zuerst in [55] verwendet und dort als Diskrete Variational Method (DVM) bezeichnet.

Eine andere Vorgehensweise besteht darin, den Fehler in der Energie zu minimieren, der durch die Verwendung des Modellpotentials anstelle von $V(\mathbf{r})$ entsteht. Der Fehler durch die Verwendung der Modelldichte anstelle der SCF-Dichte wirkt sich ausschließlich auf die Coulomb-Energie (2.6) aus, da die Austausch-Korrelationsenergie (2.10) als Funktional der SCF-Dichte berechnet werden kann.

Zur Bestimmung der Modelldichte teilt man das Coulomb-Potential V^C und die Dichte ρ in einen Modell- und einen Restterm auf:

$$\rho = \tilde{\rho} + \Delta\rho \quad (2.50)$$

$$\begin{aligned} V^C[\rho; \mathbf{r}] &= V^C[\tilde{\rho} + \Delta\rho; \mathbf{r}] \\ &= V^C[\tilde{\rho}; \mathbf{r}] + V^C[\Delta\rho; \mathbf{r}] \end{aligned} \quad (2.51)$$

Der Ausdruck für den Fehler ΔE^C in der Coulomb-Energie

$$\begin{aligned} \Delta E^C = E^C[\rho] - \tilde{E}^C[\tilde{\rho}] &= \int V^C[\rho; \mathbf{r}] \rho(\mathbf{r}) d^3r - \int V^C[\tilde{\rho}; \mathbf{r}] \tilde{\rho}(\mathbf{r}) d^3r \\ &= \frac{1}{2} \iint \frac{2\rho(\mathbf{r})\Delta\rho(\mathbf{r}') - \Delta\rho(\mathbf{r})\Delta\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' \end{aligned} \quad (2.52)$$

enthält als letzten Summand im Zähler den Fehler zweiter Ordnung im Fehler der Dichte und kann nicht direkt berechnet werden, da er die exakte Coulomb-Energie enthält. Er wird daher

im Folgenden vernachlässigt und die der Fehler in Coulomb-Energie mit Hilfe von (2.51) als Funktional von $\tilde{\rho}$ ausgedrückt:

$$\Delta E_1^C[\rho, \tilde{\rho}] = \frac{1}{2} \iint \frac{(\rho(\mathbf{r}) - \tilde{\rho}(\mathbf{r}))(\rho(\mathbf{r}') - \tilde{\rho}(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' \quad (2.53)$$

Fordert man, dass dieser Fehler verschwindet, so können die Variationsgleichungen für die Molldichte hergeleitet werden. Analog zu Gleichung (2.49) lauten sie:

$$0 = 2 \int v_k(\mathbf{r})(\rho(\mathbf{r}) - \tilde{\rho}(\mathbf{r}))d^3r + \mu \int \tilde{\rho}_k(|\mathbf{r} - \mathbf{r}_k|)d^3r. \quad (2.54)$$

Eine Erweiterung des Verfahrens besteht darin, dass neben dem kugelsymmetrischen Anteil in (2.46) auch höhere Multipole mitgenommen werden. Wie bei den Symmetrieorbitalen können dann auch die Entwicklungsfunktionen für den Dichtefit in einer symmetrischen Basis dargestellt werden, um die Integration zu vereinfachen.

2.11. Die Frozen-Core-Näherung

Bei molekularen Systemen mit Geometrien in der Nähe der Gleichgewichts-Geometrie findet die interatomare Wechselwirkung hauptsächlich im Bereich der äußeren Elektronenhülle statt und die mittlere Bindungsenergien pro Atom liegt im Bereich von einigen Elektronenvolt. Die inneren Elektronen der einzelnen Atome sind sehr viel näher am Kern und stärker gebunden. Es ist daher zu erwarten, dass die Orbitale dieser Elektronen sich durch die chemische Bindung im Molekül nur geringfügig ändern.

Diese Tatsache lässt sich ausnutzen, indem die Freiheitsgrade der kernnahen (Core-)Orbitale „eingefroren“ werden. Sie beeinflussen durch ihr Potential dann zwar noch die äußeren (Valenz-)Elektronen, haben aber in allen SCF-Iterationen die gleiche Gestalt. Die Matrixelemente, an denen ausschließlich Core-Orbitale beteiligt sind, müssen also nur einmal berechnet werden, wodurch Berechnungen an großen Clustern mit schweren Kernen erst in überschaubarer Zeit möglich werden. Die Einsparung ist um so größer, je mehr Core-Orbitale eingefroren werden, wobei allerdings darauf zu achten ist, dass der energetische Beitrag der Core-Orbitale zur chemischen Bindung gering bleibt.

Die atomare Basis (2.28) wird in der Frozen-Core- (FC-) Näherung zunächst in Core- und Valenzorbitale aufgeteilt, wobei im Folgenden ein gemeinsamer Index für die Nummer a des Kerns und die Quantenzahlen n , κ und m benutzt wird:

$$\{\phi_i\} = \{\phi_i^c\} \cup \{\phi_i^v\} \quad (2.55)$$

In dieser Basis zerfallen sowohl die Überlappmatrix \mathbf{S} , als auch die Fock-Matrix \mathbf{H} in der Gleichung (2.25) in Core-, Valenz- und gemischte Anteile:

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}^{cc} & \mathbf{H}^{cv} \\ \mathbf{H}^{vc} & \mathbf{H}^{vv} \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} \mathbf{S}^{cc} & \mathbf{S}^{cv} \\ \mathbf{S}^{vc} & \mathbf{S}^{vv} \end{pmatrix} \quad (2.56)$$

Damit die Säkulargleichung (2.25) in Core- und Valenzanteil separiert, müssen die Nebendiagonalblöcke verschwinden. Die Überlappmatrix ändert sich während der SCF-Iterationen nicht und

kann daher vor der SCF-Schleife durch eine Transformation auf eine Block-Diagonal-Form gebracht werden. Dies geschieht in drei Schritten, zuerst werden die Core-Orbitale diagonalisiert, anschließend der Valenz-Orbitale auf dem Core und schließlich die Valenz-Orbitale unter sich selbst [66]. In der ursprünglichen, molekularen Basis χ_i diagonalisiert die zusammengesetzte Transformation U dann die Überlappmatrix:

$$S' = USU^\dagger = \mathbf{1} \quad (2.57)$$

$$U = \begin{pmatrix} 1 & 0 \\ 0 & U^{vv} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ (U^{vv})^{-1}U^{vc}(U^{cc})^{-1} & 1 \end{pmatrix} \begin{pmatrix} U^{cc} & 0 \\ 0 & 1 \end{pmatrix} \quad (2.58)$$

Auch die Fock-Matrix in der Säkulargleichung muss transformiert werden:

$$H' = U \begin{pmatrix} H^{cc} & H^{cv} \\ H^{vc} & H^{vv} \end{pmatrix} U^\dagger = \begin{pmatrix} H'^{cc} & H'^{cv} \\ H'^{vc} & H'^{vv} \end{pmatrix} \quad (2.59)$$

Hier verschwinden die Nebendiagonalblöcke allerdings nicht notwendigerweise. Die Frozen-Core-Näherung besteht nun darin, dass die Blöcke H'^{cv} und H'^{vc} vernachlässigt werden. Die Säkular-Gleichung kann dann getrennt für Core- und Valenzanteil gelöst werden. Der Core-Anteil bleibt in jeder Iteration konstant, da sich das Valenz-Potential durch die Entkopplung nicht mehr auf die Core-Elektronen auswirkt und der Core selbst immer als vollständig besetzt angenommen wird.

Die Frozen-Core-Methode kann auch in einer Symmetrie-Basis (2.40) verwendet werden, wobei die Block-Matrizen des Core- und Valenzanteils nochmals in die irreduziblen Darstellungen (2.38) der Symmetrie-Gruppe aufgespalten werden.

3. Das Mehrzentren-Integrationsverfahren

Die Lösung der Säkulargleichung (2.25) bedingt die Berechnung der Matrixelemente des Fock-Operators $H_{kj} = \langle \chi_k | \hat{H}^{\text{KS}} | \chi_j \rangle$ und der Überlappmatrix $S_{jk} = \langle \chi_k | \chi_j \rangle$ in der Basis der Symmetrie-Orbitale $|\chi\rangle$, welche sich wiederum aus einer Linearkombination der atomaren Orbitale (2.40) zusammensetzen. Das Molekülstrukturprogramm berechnet die Matrixelemente in der Ortsdarstellung, die man erhält, wenn man die Einheit $1 = \int |\mathbf{r}\rangle \langle \mathbf{r}| d^3r$ einschreibt und ausnutzt, dass der Fock-Operator diagonal im Ortsraum ist:

$$\begin{aligned} \sum_j c_{ij} \langle \chi_k | \hat{H}^{\text{KS}} | \chi_j \rangle &= \epsilon_i \sum_j c_{ij} \langle \chi_k | \chi_j \rangle \\ \sum_j c_{ij} \iint \langle \chi_k | \mathbf{r} \rangle \langle \mathbf{r} | \hat{H}^{\text{KS}} | \mathbf{r}' \rangle \langle \mathbf{r}' | \chi_j \rangle d^3r d^3r' &= \epsilon_i \sum_j c_{ij} \int \langle \chi_k | \mathbf{r}'' \rangle \langle \mathbf{r}'' | \chi_j \rangle d^3r'' \\ \sum_j c_{ij} \int \chi_k(\mathbf{r}) \hat{H}^{\text{KS}}(\mathbf{r}) \chi_j(\mathbf{r}) d^3r &= \epsilon_i \sum_j c_{ij} \int \chi_k(\mathbf{r}'') \chi_j(\mathbf{r}'') d^3r'' \quad (3.1) \end{aligned}$$

Sämtliche zu berechnenden Integrale über den \mathbb{R}^3 haben also die Form

$$I = \int \chi_k(\mathbf{r}) O(\mathbf{r}) \chi_j(\mathbf{r}) d^3r =: \int F(\mathbf{r}) d^3r, \quad (3.2)$$

wobei die Funktion $O(\mathbf{r})$ entweder ein Potentialterm des effektiven Einteilchen-Potentials (2.14), die kinetische Energie, oder im Fall der Überlappmatrix die konstante Funktion $O(\mathbf{r}) = 1$ ist.

Die analytische Struktur der Funktionen $F(\mathbf{r})$ kann sehr komplex werden. Die Symmetrieorbitale $\chi_j(\mathbf{r})$ sind aus atomaren Orbitalen aufgebaut, und sind daher an einer oder mehreren Kernkoordinaten unstetig oder singulär. Die Funktion $O(\mathbf{r})$ kann ebenfalls Singularitäten an einem oder mehreren Kernorten aufweisen, z. B. beim Coulomb-Potential. Je nachdem, an welcher Anzahl von Kernkoordinaten Singularitäten auftreten, werden die Integrale (3.2) als Ein- oder Mehrzentren-Integrale bezeichnet.

Die Aufgabe eines numerischen Integrationsverfahrens ist nun, die Integrale (3.2) zu approximieren, indem eine Anzahl von Stützstellen und Gewichten $\{\mathbf{r}_i, w_i\}$ berechnet wird, so dass gilt:

$$I = \iiint F(\mathbf{r}) d^3r = \left(\sum_i F(\mathbf{r}_i) w_i \right) + \epsilon \quad (3.3)$$

zu finden, durch die der Wert des Integrals approximiert wird. Gleichzeitig soll die numerische Genauigkeit ϵ der Integrationsregel abgeschätzt werden.

Für die Berechnung von atomaren Einzentren-Integralen ist die Einführung von Kugelkoordinaten naheliegend. Die Schrödinger- bzw. Dirac-Gleichung mit einem zentralen Kernpotential faktorisiert in diesen Koordinaten in einen Radial- und einen Winkelanteil. Der Winkelanteil wird meist analytisch behandelt, der Radialanteil je nach Methode in einer Basis von analytischen oder numerischen Basisfunktionen berechnet.

Bei zweiatomigen Systemen lässt mit Hilfe der elliptisch-hyperbolischen Koordinaten [67] ebenfalls eine Transformation durchführen, die die numerische Berechnung der Matrixelemente im Ortsraum vereinfacht. Ist der Hamilton-Operator rotationssymmetrisch bezüglich der Verbindungsachse der beiden Kerne, so lässt sich die Integration um diese Achse analytisch ausführen, über die beiden verbleibenden Koordinaten muss numerisch integriert werden.

Sind mehr als zwei Atome in dem zu berechnenden System enthalten, so lässt sich im Allgemeinen keine geeignete analytische Koordinatentransformation angeben, die eine numerische Integration erlaubt. Die Singularitäten der Integranden an den Orten der Kerne erlauben es aber auch nicht, eine einfache Produktregel für das Integrationsgebiet zu verwenden.

Daher zerlegen alle Mehrzentren-Verfahren den \mathbb{R}^3 in Teilräume mit endlicher oder unendlicher Ausdehnung, über die getrennt integriert wird. Dabei wird versucht, die Singularitäten des Integranden an den Kernorten durch die Raumaufteilung zu separieren. Die verschiedenen Möglichkeiten einer solchen Aufteilung können allgemein durch sog. Zellenfunktionen dargestellt werden und werden im folgenden Abschnitt behandelt.

Die nächsten Abschnitte beschreiben dann die Integrationsverfahren für die einzelnen Zellen, aufgeteilt in die Koordinatentransformation, die numerischen Integrationsregeln, die Adaption der Punktzahl an das Integrationsgebiet und die Ausnutzung der Molekülsymmetrie zur Vereinfachung der Integrationsregel.

Alle in diesem Kapitel beschriebenen Methoden sind im Programmpaket MULIN (MULTI-center INtegration) enthalten, dass im Rahmen dieser Arbeit entstanden ist. Auf das Programm selbst wird im letzten Abschnitt des Kapitels eingegangen.

3.1. Die Zellenfunktionen

Alle bekannten numerischen Verfahren behandeln Mehrzentrenintegrale der Form (3.2), indem sie eine Zerlegung der Integranden auf einzelne Zellen durchführen, die durch Zellenfunktionen $z_A(\mathbf{r})$ beschrieben werden können. Das Gesamtintegral wird dabei als Summe der Integrale aller Zellen dargestellt:

$$\begin{aligned} I &= \iiint F(\mathbf{r}) \sum_A z_A(\mathbf{r}) d^3r \\ &= \sum_A \iiint \underbrace{F(\mathbf{r}) z_A(\mathbf{r})}_{F_A(\mathbf{r})} d^3r = \sum_a I_A. \end{aligned} \quad (3.4)$$

Vergleicht man (3.2) und (3.4), so sieht man, dass die Zellenfunktionen für alle \mathbf{r} normiert sein müssen:

$$1 = \sum_A z_A(\mathbf{r}). \quad (3.5)$$

Für die Zellenfunktionen, die in der Literatur auch als Projektionsfunktionen [15], oder nukleare Gewichtsfunktionen [16] bezeichnet werden, existieren die verschiedensten Ansätze. Allen gemeinsam ist, dass sie das Integral über den gesamten \mathbb{R}^3 in räumliche Teilgebiete zerlegen, so dass die Mehrzentren-Integrale in Summen von Einzentren-Integrale zerfallen, und sich dann numerisch einfach integrieren lassen.

Die verschiedenen Verfahren der Raumaufteilung lassen sich grob in zwei Klassen einteilen, die einen, die disjunkte Raumgebiete verwenden, und die anderen, bei denen sich die Zellenfunktionen überlappen („Fuzzy-Cells“).

Die früheste Anwendung der Raumaufteilung mittels Zellenfunktionen findet sich bei Boys und Rajagopal [68]. Das Verfahren wurde später von Ellis in Verbindung mit der Diophantine-Integration [11, 12, 68, 69] verwendet für Kristalle [53] und nicht-periodische Systeme [70]. Die bekannteste darauf basierende Arbeit ist jedoch von Becke [13, 71], die in vielfältiger Form [16, 18, 20] abgewandelt wurde, und dessen Raumaufteilung in einem folgenden Abschnitt vorgestellt wird.

Ebenfalls besprochen wird das Verfahren von Baerends et al. [9], der eine Aufteilung in nicht-überlappende Raumgebiete benutzt. Das Baerends'sche Verfahren wird bisher von uns für die Integration der Matrixelemente verwendet und wird in dieser Arbeit zum Vergleich herangezogen.

3.1.1. Das Verfahren von Baerends

Die Methoden, die mit nicht-überlappenden Raumgebieten arbeiten, haben im Vergleich zu anderen Verfahren den Vorteil, dass man eine klare Vorstellung von der Geometrie der Teilräume hat. Das Verfahren von Baerends et al. [9, 72] arbeitet mit Kugeln um die atomaren Zentren, daran anschließenden Pyramidenstümpfen, aus deren Spitzen an den Kernen zentrierte Kugelabschnitte herausgeschnitten sind, sowie Polyedern in den Zwischenräumen.

Die Zellenfunktionen $z_A(\mathbf{r})$ bei diesem Verfahren ordnen einem Raumpunkt \mathbf{r} den Wert 1 zu, wenn er sich innerhalb des Volumens der A -ten Zelle befindet, anderenfalls 0. Das Volumen aller Zellen ist beschränkt, da die gesamte Struktur noch in ein einhüllendes äußeres Polynom eingeschlossen ist.

Der Nachteil dieses Verfahrens ist, dass die Dreifachintegrale über die verschiedenen geometrischen Objekte komplizierte Abhängigkeiten der Integrationsgrenzen aufweisen und daher numerisch erst nach einer Variablentransformation mit einer Produktregel behandelt werden können. Die Jacobi-Determinanten dieser Transformationen sind teilweise singulär und erhöhen die Schwierigkeit der numerischen Integration.

Ein weiteres Problem ist die Vielzahl der separierten Raumbereiche. Um eine hohe Genauigkeit zu erreichen, müssen für eine große Zahl von Raumbereichen getrennte Integrationsregeln aufgestellt werden. Dadurch kann die Eigenschaft des Integranden, an fast allen Punkten des \mathbb{R}^3 differenzierbar zu sein, weniger gut ausgenutzt werden.

3.1.2. Die Zellenfunktion von Becke

Die Zellenfunktion von Becke [13, 71] zerlegt in Integrationsraum mit analytischen Zellenfunktionen in überlappende Integrationszellen. Die atomare Gewichtsfunktion $z_A(\mathbf{r})$ in Glei-

chung (3.4) wird so gewählt, dass sie in der Nähe des Kernes von Atom A gegen 1 geht, in Richtung aller anderen Kerne hinreichend schnell gegen 0 abfällt. Dadurch wird erreicht, dass im Integral I_A singuläre Anteile des Integranden oder seiner Ableitungen nur am Punkt \mathbf{r}_A^0 vorhanden sind, während die Singularitäten am Ort aller anderen Kerne unterdrückt werden.

Die grundlegende Idee dabei ist, für ein System aus zwei Zentren eine Funktion zu konstruieren, die in der Nähe des einen Zentrums einen konstanten Wert hat und in der Richtung des anderen Zentrums gegen null geht. Eine verallgemeinerte Funktion für den n -Zentren-Fall erhält man dann durch die Multiplikation von $(n - 1)$ Zwei-Zentren-Funktionen.

Führt man für zwei Zentren \mathbf{r}_A^0 und \mathbf{r}_B^0 die Transformation von kartesischen Koordinaten \mathbf{r} auf elliptisch-hyperbolische Koordinaten (λ, μ, ϕ) ein, die durch

$$\lambda_{AB}(\mathbf{r}) = \frac{|\mathbf{r} - \mathbf{r}_A^0| + |\mathbf{r} - \mathbf{r}_B^0|}{|\mathbf{r}_A^0 - \mathbf{r}_B^0|} \quad (3.6)$$

$$\mu_{AB}(\mathbf{r}) = \frac{|\mathbf{r} - \mathbf{r}_A^0| - |\mathbf{r} - \mathbf{r}_B^0|}{|\mathbf{r}_A^0 - \mathbf{r}_B^0|} \quad (3.7)$$

definiert sind [67], so ist die Variable $\mu_{AB}(\mathbf{r}) \in [-1, 1]$ geeignet, um die Nähe zum einen oder anderen Zentrum zu parametrisieren. Für $\mathbf{r} \rightarrow \mathbf{r}_A^0$ geht $\mu_{AB}(\mathbf{r})$ gegen -1 , für $\mathbf{r} \rightarrow \mathbf{r}_B^0$ aber gegen $+1$.

Mit Hilfe von μ_{AB} lässt sich nun eine polynomiale Abschneidefunktion $h(\mu)$ mit geeigneten analytischen Eigenschaften definieren:

$$\begin{aligned} h(1) &= 1 & h(-1) &= -1 \\ h'(1) &= 0 & h'(-1) &= 0 \end{aligned} \quad (3.8)$$

Das einfachste Polynom mit diesen Eigenschaften ist ein Polynom dritten Grades, z. B.

$$h(\mu) = \frac{3}{2}\mu - \frac{1}{2}\mu^3. \quad (3.9)$$

Der Definitionsbereich der Abschneidefunktion ist mit ihrem Wertebereich identisch, so dass weitere Abschneidefunktionen durch Iteration der Funktion erzeugt werden können, wodurch der Verlauf der Funktion mit jeder Iteration steiler wird. In [13] werden drei Iterationen gewählt:

$$h_3(\mu) := h(h(h(\mu))), \quad (3.10)$$

wodurch die ersten sieben Ableitungen verschwinden.

Die Zwei-Zentren-Zellenfunktion $z_A^{(2)}$ für die Zelle A erhält man durch die Skalierung auf das Intervall $[0, 1]$

$$z_A^{(2)}(\mathbf{r}) = (1 - h_3(\mu_{AB}(\mathbf{r}))) / 2, \quad B \neq A \quad (3.11)$$

wobei die verschiedenen Indices A und B die beiden Kerne bezeichnen.

Die Zellenfunktion $z_A(\mathbf{r})$ der Zelle A im allgemeinen N -Zentren-Fall erhält man durch Bildung eines Produkts von Zwei-Zentren-Zellenfunktionen

$$p_A(\mathbf{r}) = \prod_{B \neq A}^N z_B^{(2)}(\mathbf{r}), \quad (3.12)$$

welches anschließend noch normiert werden muss, um die Bedingung (3.5) zu erfüllen:

$$z_A(\mathbf{r}) = \frac{p_A(\mathbf{r})}{\sum_B p_B(\mathbf{r})}. \quad (3.13)$$

3.1.3. Modifikationen der Becke-Funktion

In [13] wurde ein weiterer Parameter eingeführt, der es erlaubt, die Ausdehnung der einzelnen Zellen noch von der Art der zugehörigen Atome abhängig zu machen. Die Oberflächen $\mu_{AB}(\mathbf{r}) = 0$ teilen die Verbindung der Kerne A und B senkrecht in zwei Hälften. Diese Aufteilung ist für homonukleare Systeme sicherlich gerechtfertigt. Für heteronukleare Systeme wird dagegen vorgeschlagen, die Zellengrenze in Richtung des „kleineren“ Atoms zu verschieben. Dazu wurde eine Variablentransformation

$$\nu_{AB} = \mu_{AB} + a_{AB}(1 - \mu_{AB}^2) \quad (3.14)$$

eingeführt und die Funktion $h(\nu_{AB})$ anstelle von $h(\mu_{AB})$ verwendet. Es wurde argumentiert, dass ein guter Wert von a_{AB} durch den Zusammenhang

$$a_{AB} = u_{AB}/(u_{AB}^2 - 1) \quad (3.15)$$

$$u_{AB} = (\chi_{AB} - 1)/(\chi_{AB} + 1) \quad (3.16)$$

$$\chi_{AB} = R_A/R_B \quad (3.17)$$

gegeben ist, wobei R_A den Bragg-Slater-Radius des Atoms A bezeichnet und in [73] tabelliert ist.

Treutler et al. [18] erhalten bessere Ergebnisse, wenn die Definition (3.17) durch

$$\chi_{AB} = \sqrt{\frac{R_A}{R_B}} \quad (3.18)$$

ersetzt oder ganz auf die Veränderung der Zellgröße verzichtet wird, was sich mit den Ergebnissen aus [74] deckt.

3.1.4. Voronoi-Zellen

Eine spezielle Wahl der Abschneidefunktion $h(\mu)$ in Gleichung (3.8) führt zu einer Zerlegung des Raums in disjunkte Bereiche, den sog. Voronoi-Zellen¹

$$h(\mu) = 2\theta(\mu) - 1, \quad (3.19)$$

wobei θ die Stufenfunktion ist ($\theta(\mu) = 0$ für $\mu < 0$; $\theta(\mu) = 1$ für $\mu > 0$). Die einzelnen Zellen werden dann von den Flächen begrenzt, auf denen $\mu = 0$ ist. Da die Zellenfunktion das Produkt aller Abschneidefunktionen ist, sind die Grenzen der Zellen jeweils Polygone, die sich

¹In der Festkörperphysik werden die Voronoi-Zellen für unendlich ausgedehnte Kristalle auch Wigner-Seitz-Zellen genannt

zwischen dem Zentrum der Zelle und allen nächsten Nachbarzentren befinden. Die Polygone schneiden die Verbindungsgeraden zwischen jeweils zwei benachbarten Zentren genau in der Mitte, falls a_{AB} in Gleichung (3.14) gleich null ist.

Die Zerlegung in Voronoi-Zellen ist für die Integration allerdings nicht praktikabel, da eine solche Zellenfunktion an den Zellgrenzen nicht stetig ist und die numerische Integration dadurch erschwert wird.

3.1.5. Die Zellenfunktionen von Delley

Eine andere Art der Raumaufteilung wurde von Delley [14] vorgeschlagen. Ausgehend von Funktionen $g_i(\mathbf{r})$, die am Ort \mathbf{r}_A^0 des Kerns A singular sind, wird die Zellenfunktion analog zu Gleichung (3.13) direkt durch Normierung erzeugt:

$$z_A(\mathbf{r}) = \frac{g_A(\mathbf{r} - \mathbf{r}_A^0)}{\sum_{B \neq A} g_B(\mathbf{r} - \mathbf{r}_A^0)}. \quad (3.20)$$

Als geeignete Funktionen werden dabei

$$g_A(r) = \rho_A^2(r) \quad (3.21)$$

$$g_A(r) = \rho_A^2(r)/r^2 \quad (3.22)$$

$$g_A(r) = \rho_A(r)(e^{r_0/r} - 1 - r_0/r) \quad (3.23)$$

vorgeschlagen. Darin stellt die Funktion $\rho_A(r)$ die sphärische atomare Elektronendichte dar, die z. B. aus den atomaren Basissätzen berechnet werden kann. Diese Funktionen hängen implizit von der Ausdehnung der atomaren Basis ab, eine Größenkorrektur wie bei den Beckenfunktionen entfällt somit.

In Gleichung (3.23) verhält sich der Term in Klammern für $r \rightarrow 0$ wie $e^{r_0/r}$, für $r \rightarrow \infty$ dagegen wie $\frac{1}{2}(r_0/r)^2$. Dies bewirkt, dass sich $g_A(r)$ in Kernnähe wie die Dichte ρ_A verhält, für große Abstände jedoch schneller abfällt, nämlich proportional zu ρ_A/r^2 . Gleichung (3.23) interpoliert also zwischen den Funktionen (3.21) und (3.22), wobei der Parameter r_0 eine Längenskala für die Umschaltung vorgibt.

Für den Parameter wird in [14] kein Wert angegeben, in [21] wird $r_0 = 0.5 a_0$ benutzt.

3.1.6. Die Zellenfunktionen von Stratmann et al.

Die Zellenfunktion von Becke ist so beschaffen, dass in der Nähe eines Kerns die zugehörige Zellenfunktion nicht exakt gleich eins ist, so dass die Singularität des Kerns noch in anderen Zellenfunktionen Beiträge liefert. Dort kann die Singularität nicht durch die Radialtransformation behoben werden, da sie sich nicht im Ursprung des Koordinatensystems befindet.

Eine Lösung für dieses Problem wird von Stratmann et al. in [20] vorgeschlagen. Dazu wird die Abschneidefunktion $g(\mu, a)$ stückweise wie folgt definiert:

$$h(\mu, a) = \begin{cases} -1 & \text{für } \mu \leq -a \\ z(\mu, a) & \text{für } -a < \mu < a \\ +1 & \text{für } \mu \geq a \end{cases} \quad (3.24)$$

Die Funktion $z(\mu, a)$ muss analog zu Gleichung (3.8) die folgenden Anschlussbedingungen erfüllen:

$$\begin{aligned} z(\mu, a) &= 1 & z(\mu = -a, a) &= -1 \\ z'(\mu = a, a) &= 0 & z'(\mu = -a, a) &= 0 \end{aligned} \quad (3.25)$$

Auf das verkürzte Intervall $[-a, a]$ transformiert, lautet Gleichung (3.9):

$$z(\mu, a) = \frac{3}{2}(\mu/a) - \frac{1}{2}(\mu/a)^3 \quad (3.26)$$

Auch hier lässt sich durch Iteration von $z(\mu, a)$ der Übergang der Zellen beeinflussen. Wird die Funktion wie in Gleichung (3.10) iteriert, so erhält man die Abschneidefunktion

$$z_3(\mu, a) = \frac{1}{16} (35(\mu/a) - 35(\mu/a)^3 + 21(\mu/a)^5 - 5(\mu/a)^7) \quad (3.27)$$

deren erste drei Ableitungen bei $\mu = \pm a$ verschwinden. Die Zellenfunktion $z_A(\mathbf{r})$ erhält man wie bei Becke durch Einsetzen von Gleichung (3.27) in Gleichung (3.11), (3.12) und (3.13).

Diese Funktion verschiebt den Punkt auf der internuklearen Achse, bei dem die Funktion exakt den Wert ± 1 annimmt, von den Kernen weg, so dass die Singularitäten ausschließlich in der zum Kern gehörenden Zelle auftreten. Der dimensionslose Parameter a definiert die Entfernung, bei der alle nicht zum eigenen Kern gehörigen Zellenfunktionen verschwinden. Als geeigneter Wert wird in [20] $a = 0.64$ gewählt.

Ein Vorteil dieser Funktion ist, dass durch einen einfachen Test ermittelt werden kann, ob das Polynom in Gleichung (3.26) überhaupt berechnet werden muss, oder der Wert von $z_B^{(2)}(\mathbf{r})$ in Gleichung (3.12) gleich eins ist. Dies ist der Fall, wenn für \mathbf{r} in der Zelle A gilt:

$$|\mathbf{r} - \mathbf{r}_A^0| < \frac{1}{2}(1 - a)|\mathbf{r}_A^0 - \mathbf{r}_B^0|, \quad (3.28)$$

wobei B den Index der nächsten Nachbarzelle bezeichnet. Ist diese Bedingung erfüllt, so sind die Zellenfunktionen aller anderen Zellen $B \neq A$ gleich 0, so dass bei einer numerischen Integrationsregel die zugehörigen Integrationspunkte keinen Beitrag zum Gesamtintegral liefern und gestrichen werden können.

3.2. Die Koordinatentransformationen

Abhängig von der Wahl der Zellenfunktion können verschiedene Koordinatentransformationen des \mathbb{R}^3 auf intrinsische Koordinaten

$$\mathbf{r} \rightarrow \mathbf{u}(\mathbf{r}) \in \mathbb{R}^3 \quad (3.29)$$

geeignet sein, die Integration der Funktionen $F_A(\mathbf{r})$ in den einzelnen Zellen (3.4) zu vereinfachen.

Da die Integranden für die einzelnen Zellen nach der Raumaufteilung die Struktur von Einzentren-Integralen haben, ist zunächst eine Transformation auf Kugelkoordinaten angemessen. In diesen Koordinaten haben die Radial- und Winkelvariablen einen unterschiedlichen Charakter. In der Radial-Variablen verhält sich der Integrand ähnlich dem Radialanteil der atomaren

Wellenfunktion und muss deswegen durch eine zusätzliche Radial-Transformation für die numerische Quadratur vorbereitet werden. In den Winkel-Koordinaten werden die Integranden in Kernnähe von den Wellenfunktionen des zur Zelle gehören Atoms und je nach Integrand zusätzlich vom Potential am Kernort dominiert. Im interatomaren Bereich tragen auch die Anteile der Nachbaratome bei. Für große Abstände geht die Zellenfunktion in den Richtungen, in denen andere Zellen liegen, gegen Null und anderenfalls gegen Eins.

Die Koordinatentransformationen können getrennt für einzelne Koordinaten oder für alle drei Koordinaten gemeinsam definiert sein. Im Allgemeinen wird man auch mehrere Koordinatentransformationen verknüpfen, um geeignete intrinsische Koordinaten zu erhalten.

3.2.1. Kugelkoordinaten

Enthält eine Zelle einen Kern bei r_A^0 , so können Integranden am Kernort singularär sein, wenn sie Wellenfunktionen des A -ten Kerns enthalten. Geeignete Koordinaten sind in diesem Fall die Kugelkoordinaten mit r_A^0 als Ursprung, die die Singularität in einen Radial- und einen Winkelanteil separieren. Der Fall, dass Zellen keine Kernsingularität enthalten, tritt lediglich beim Baerends'schen Integrationsverfahren auf und soll in dieser Arbeit nicht behandelt werden.

Bei der Wahl von Kugelkoordinaten lässt sich Lage der Koordinatenachsen noch wählen. Mit Hilfe einer Drehung kann das lokale Koordinatensystem so ausgerichtet werden, dass die Symmetrie-Gruppe der numerischen Integrationsregel und des Moleküls eine gemeinsame Untergruppe besitzen und die Zahl der Integrationspunkte verringert werden kann. Die Drehung kann durch eine orthogonale Drehmatrix \mathbf{R} dargestellt werden. Die gesamte Transformation lautet damit:

$$\mathbf{x} = \mathbf{x}_0 + \mathbf{R} \begin{pmatrix} r \sin(\theta) \cos(\phi) \\ r \sin(\theta) \sin(\phi) \\ r \cos(\theta) \end{pmatrix} \quad (3.30)$$

3.2.2. Die Radialtransformation

Die Abhängigkeit der Radialanteile von vielen der vorkommenden Integranden wird durch das singularäre Verhalten der Wellenfunktionen bei $r \rightarrow 0$ und den exponentiellen Abfall für $r \rightarrow \infty$ bestimmt. Die Aufgabe der Radialtransformation ist nun, die Variable r in eine geeignete intrinsische Variable u zu überführen, über die sich leicht numerisch integrieren lässt.

Bei den Gauß'schen Integrationsformeln werden die Funktionen $f(u)$ implizit durch eine endliche Entwicklung in orthogonale Polynome über dem Integrationsbereich entwickelt. Die Approximation von $f(u)$ ist exakt für Polynome bis zum Grad $\leq d$, wenn eine Gauß'sche Formel vom Grad d verwendet wird und nicht exakt für mindestens ein Polynom mit dem Grad $d + 1$.

Die numerische Genauigkeit der Integration hängt daher davon ab, wie schnell die Entwicklung der Funktion f in den Polynomen der Integrationsformel konvergiert. Eine für alle Integranden gleichermaßen geeignete Radialtransformation zu finden, gestaltet sich äußerst schwierig und verschiedene Möglichkeiten wurden vorgeschlagen [13, 18, 19, 24].

3.2.3. Radialtransformationen für nicht-relativistische Wellenfunktionen

Ausgangspunkt der Entwicklung der Radialtransformationen ist die Publikation von Becke [13]. Dort wird die Transformation

$$r(u) = r_m \frac{1+u}{1-u} \quad (3.31)$$

vorgeschlagen, die auch in späteren Arbeiten [71, 75] verwendet wird. Die Konstante r_m wird dabei auf den halben Bragg-Slater-Radius gesetzt. Für die Elemente von $Z = 1$ bis $Z = 83$ sind die Radien in [73] tabelliert.

Die Funktionaldeterminante $r^2 dr$ der Radialtransformation lautet dann:

$$dr = 2r_m \frac{1}{(1-u)^2} du \quad (3.32)$$

Basierend auf der Euler-MacLaurin Formel haben Murray et al. [76] die folgende Transformation für das Intervall $u \in [0, 1]$ entwickelt:

$$r(u) = \alpha \frac{x^m}{(1-x)^m} \quad (3.33)$$

$$dr = \alpha m \frac{\left(\frac{1-x}{x}\right)^{-m}}{x(1-x)} du \quad (3.34)$$

Der Exponent m ist eine ganze Zahl und liefert bei dem Wert $m = 2$ gute Ergebnisse. Der Vorfaktor α ist wiederum der Bragg-Slater-Radius [73, 77].

In der Veröffentlichung von Treutler und Ahlrichs [18] werden vier verschiedene Radialtransformationen verglichen:

$$(M1): \quad r(u) = \xi \frac{a+u}{\sqrt{1-u^2}} \quad (3.35)$$

$$dr = \xi \frac{1+au}{(1-u^2)^{3/2}} du$$

$$(M2): \quad r(u) = \xi \frac{a+u}{1-u} \quad (3.36)$$

$$dr = \xi \frac{1+a}{(1-u)^2} du$$

$$(M3): \quad r(u) = \frac{\xi}{\ln 2} \ln \frac{a+1}{1-u} \quad (3.37)$$

$$dr = \frac{\xi}{\ln 2(1-u)} du$$

$$(M4): \quad r(u) = \frac{\xi}{\ln 2} (a+u)^\alpha \ln \frac{a+1}{1-u} \quad (3.38)$$

$$dr = \xi \frac{(a+u)^\alpha}{\ln 2} \left(\frac{\alpha \log \frac{1+a}{1-u}}{a+u} + \frac{1}{1-u} \right) du$$

Die Konstante a wird auf 1 gesetzt, falls die Integrationsregel das Intervall $u \in [-1, 1]$ integriert, für das Intervall $u \in [0, 1]$ wird sie als $a = 0$ gewählt.

Die Transformation M2 (3.36) entspricht dabei der Becke'schen Transformation (3.31). Die Werte für α und ξ wurden für mehrere Elemente von H bis Kr optimiert und mit verschiedenen Quadraturen kombiniert.

Als die am besten geeignete Transformation wird von den Autoren die Transformation (3.38) angegeben, zusammen mit dem Parameter $\alpha = 0.6$ und einer Tschebyscheff-Quadratur zweiter Ordnung. Die zugehörigen Werte von ξ sind für verschiedene Elemente tabelliert und liegen im Bereich von 0.8 bis 1.8.

Mura und Knowles schlagen in [19] die folgende Transformation vor:

$$r(u) = -\alpha \ln(1 - x^m) \quad (3.39)$$

Die Idee bei dieser Transformation ist, Funktionen vom Typ $\exp(-r/\alpha)$ in den intrinsischen Koordinaten als Polynom m -ten Grades darzustellen. Die Autoren finden für m einen Wert von 4 als am besten geeignet und tabellieren den Wert von α bezogen auf einen basisabhängigen Skalenfaktor α_c .

Yamamoto et al. [23, 24] teilen den radialen Integrationsbereich $[0, \infty)$ in n Abschnitte auf, deren Grenzen mit den Nullstellen der s -Orbitale der nichtrelativistischen atomaren Wellenfunktionen übereinstimmen. Die endlichen Intervalle werden mit einer Gauß-Legendre-Regel integriert, das äußerste, unendliche Intervall mit einer Gauß-Laguerre-Regel.

3.2.4. Radialtransformation für relativistische Wellenfunktionen

Die bisher besprochenen Radialtransformationen sind für nicht-relativistische analytische Wellenfunktionen optimiert. Da unser Programm aber durchgehend mit numerischen, relativistischen Basissätzen arbeitet, die das Verhalten $r^\gamma e^{-\alpha r}$ am Ort der Kerne besitzen, muss diese zusätzliche Singularität durch eine entsprechende Transformation behoben werden. Wir verwenden dazu die in [66] vorgeschlagene Radialtransformation

$$x(u) = x_i \frac{1-u}{2} + x_a \frac{1+u}{2} \quad (3.40)$$

$$m(x) = m_1 e^{-\beta_1 x} + m_2 (1 - e^{-\beta_2 x}) \quad (3.41)$$

$$r(x) = t_0 \cdot x^{m(x)}. \quad (3.42)$$

Durch die verschiedenen Parameter ist eine genaue Kontrolle des asymptotischen Verhaltens der transformierten Funktion möglich. Um dies genauer zu analysieren, entwickelt man die Funktion für $r \rightarrow 0$ und $1/r \rightarrow 0$. Die Parameter werden üblicherweise so gewählt, dass $m(x)$ annähernd konstant ist für $x \rightarrow x_i$ und $x \rightarrow x_a$. Dies entspricht den Bedingungen $\beta_1 x_i \ll 1$ und $\beta_2 x_a \gg 1$. In diesem Fall kann man in Gleichung (3.42) die Variation des Exponenten mit x vernachlässigen und erhält

$$\lim_{x \rightarrow x_i} r(x) = t \cdot x^{m_1(x_i)} \quad (3.43)$$

$$\lim_{x \rightarrow x_a} r(x) = t \cdot x^{m_2(x_i)}. \quad (3.44)$$

Man erkennt, dass die Transformation zwischen zwei Potenzfunktionen mit den Exponenten m_1 bei kleinen x und m_2 bei großen x interpoliert. Singularitäten an den Integrationsgrenzen der Form r^γ lassen sich somit durch genügend große Werte von m in ein polynomiales Verhalten transformieren. Die transformierte Funktion lässt sich anschließend leicht numerisch integrieren.

Die Parameter der Transformation (3.42) können auf verschiedene Weise bestimmt werden. Zum einen können $m_1, \beta_1, m_2, \beta_2, x_i, x_a$ und t direkt bestimmt werden. In den meisten Fällen will man dagegen Vorgaben machen, die eine physikalische oder geometrische Bedeutung haben, z. B. den innersten und äußersten Integrationspunkt für eine vorgegebene numerische Regel (u_i, w_i) . Folgendes Vorgehen hat sich als praktikabel erwiesen:

- Vorgabe von t_0, m_1, β_1 und β_2 .
- An einen bestimmten Radius r_m soll der Exponent x_m vorgegeben werden, d. h. es soll gelten: $r_m := r(t_m) = t_0 t_m^{x_m}$. Dazu muss das zugehörige t_m bestimmt werden:

$$t_m = \left(\frac{r_m}{t_0} \right)^{1/x_m} \quad (3.45)$$

Nun kann der Parameter m_2 nach Gleichung (3.41) berechnet werden. Es gilt:

$$m_2 = \frac{x_m - m_1 e^{-\beta_1 t_m}}{1 - e^{-\beta_2 t_m}} \quad (3.46)$$

- Vorgabe der innersten und äußersten Punkte r_i und r_a . Daraus können durch Lösen der Gleichung (3.42) die Grenzen x_i und x_a berechnet werden. Da die Gleichung nicht algebraisch nach x auflösbar ist, wird ein Newton-Verfahren eingesetzt. Der innerste Punkt r_i kann gleich Null gesetzt werden, falls die numerische Integrationsregel keine Stützstelle an der zugehörigen Intervallgrenze $u = -1$ enthält.

3.3. Aufteilung des dreidimensionalen Integrals

Durch die Zerlegung der Integrale (3.2) in eine Summe von Integralen I_A in Zellenfunktionen (3.4) und eine anschließende Transformation (3.29) auf angepasste intrinsische Koordinaten $\mathbf{u}_A(\mathbf{r})$ gelangt man zur Darstellung

$$I_A = \iiint F(\mathbf{r}) z_A(\mathbf{r}) d^3 r = \iiint \underbrace{F(\mathbf{r}(\mathbf{u})) z_A(\mathbf{r}(\mathbf{u})) \left| \frac{\partial \mathbf{r}_A(\mathbf{u})}{\partial \mathbf{u}} \right|}_{f_A(\mathbf{u})} d^3 u. \quad (3.47)$$

Der Zellenindex A wird im Folgenden unterdrückt. Die numerische Integration des dreidimensionalen Integrals über die intrinsischen Koordinaten kann nun auf Integrale in niedrigerer Dimension zurückgeführt werden. Da einige Sonderfälle berücksichtigt werden müssen, fassen wir allgemein einzelne intrinsischen Koordinaten mit Hilfe eines Indexvektors X_j^k zu Vektoren \mathbf{K}^k von intrinsischen Koordinaten zusammen.

$$\begin{aligned} \mathbf{K}^k &= (u^{X_1^k}, \dots, u^{X_{N_k}^k}), & X_j^k &\in \{1, 2, 3\}, & j &\in \{1 \dots J_{\max}^k\} \\ & & k &\in \{1 \dots K_{\max}\}, & \sum_k & N_k = 3, \end{aligned} \quad (3.48)$$

für die eine Umkehrabbildung U

$$u^i = U^i(\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}}), \quad i \in \{1, 2, 3\} \quad (3.49)$$

existieren muss. Wir definieren einen neuen Integranden in diesen Koordinaten

$$g(\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}}) := f(U_1(\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}}), \dots, U_3(\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}})) \quad (3.50)$$

und integrieren iterativ über \mathbf{K}^k , wobei die Integrationsgrenzen mit $\mathbf{K}^1_{1,2}$ bezeichnet sind und von den Integrationsvariablen der äußeren Integrale abhängen können:

$$I = \int_{\mathbf{K}^1_1}^{\mathbf{K}^1_2} d\mathbf{K}^1 \dots \int_{\mathbf{K}^{K_{\max}}_1}^{\mathbf{K}^{K_{\max}}_2} g(\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}}) d\mathbf{K}^{K_{\max}} \quad (3.51)$$

Mit Hilfe des Indexvektors X_j^k lässt sich nun festlegen, in welcher Reihenfolge über die intrinsischen Koordinaten integriert wird und ob sie einzeln, paarweise oder zu Tripeln gruppiert werden.

3.4. Numerische Integrationsregeln

Die Quadratur erfolgt nun ebenfalls iterativ in den Variablen $\mathbf{K}^1, \dots, \mathbf{K}^{K_{\max}}$, die Stützstellen sind mit $\mathbf{K}^k_{i_k}$ bezeichnet, die entsprechenden Gewichte mit $w_{i_k}^k$:

$$I = \epsilon^1 + \sum_{i^1}^{N^1} w_{i^1}^1 \dots \left(\epsilon^{K_{\max}} + \sum_{i^{K_{\max}}}^{N^{K_{\max}}} w_{i^{K_{\max}}}^{K_{\max}} g(\mathbf{K}^1_{i^1}, \dots, \mathbf{K}^{K_{\max}}_{i^{K_{\max}}}) \right) \quad (3.52)$$

Zieht man alle Integrationsgewichte in die innerste Summe und spaltet die Fehlerterme ab, so bekommt man die dreidimensionale Integrationsregel in der Form von Gleichung (3.3):

$$I = \sum_{i^1}^{N^1} \dots \sum_{i^{K_{\max}}}^{N^{K_{\max}}} w_{i^1}^1 \dots w_{i^{K_{\max}}}^{K_{\max}} g(\mathbf{K}^1_{i^1}, \dots, \mathbf{K}^{K_{\max}}_{i^{K_{\max}}}) + \epsilon^1 + \dots + \sum_{i^1}^{N^1} \dots \sum_{i^{K_{\max}-1}}^{N^{K_{\max}-1}} w_{i^1}^1 \dots w_{i^{K_{\max}-1}}^{K_{\max}-1} \epsilon^{K_{\max}}. \quad (3.53)$$

Man erkennt, dass der Integrationsfehler nur von der numerischen Integrationsregel, nicht aber vom Integranden abhängt. Der Gesamtfehler der Integrationsregel setzt sich summarisch zusammen aus den Fehlern der einzelnen Integrale. Der Anteil des Fehlers der inneren Integrale ist über die Integrationsgewichte wesentlich stärker abhängig von der numerischen Integrationsregel als der Fehler der äußeren Integrale.

Ein Spezialfall der mehrdimensionalen Integrationsregeln (3.51) ist die Produktregel. Bei ihr sind die Integrationsregeln der inneren Integrale nicht vom Integrationsindex des äußeren Integrals abhängig. Dies betrifft sowohl die Transformation, als auch die numerische Regel $\{\mathbf{r}, w\}$ und die Zahl der Stützstellen N^i . In diesem Fall können die Summationen vertauscht werden.

Um die endgültigen Punkte \mathbf{r}_i und Gewichte w_i für Gleichung (3.3) zu erhalten, müssen die Rücktransformationen auf kartesische Koordinaten ausgeführt und die Regeln aller Zellen zusammengefasst werden. Die Stützstellen werden durch einen vollständigen Satz der Quadraturindizes der einzelnen Koordinaten $(i^1, \dots, i^{K_{\max}})$ beschrieben:

$$\begin{aligned}
\mathbf{U}_A^{(i^1, \dots, i^{K_{\max}})} &:= \mathbf{U}_A \left(\mathbf{K}_{A, i^1}^1, \dots, \mathbf{K}_{A, i^{K_{\max}}}^{K_{\max}} \right) \\
\{\mathbf{r}\}_A &= \mathbf{r}_A \left(\mathbf{U}_A^{(i^1, \dots, i^{K_{\max}})} \right) \\
\{w\}_A &= w_{A, i^1}^1 \cdot \dots \cdot w_{A, i^{K_{\max}-1}}^{K_{\max}-1} \\
&\quad \cdot z_A \left(\mathbf{U}_A^{(i^1, \dots, i^{K_{\max}})} \right) \cdot \left| \frac{\partial \mathbf{r}_A(\mathbf{u})}{\partial \mathbf{u}} \right|_{\mathbf{U}_A^{(i^1, \dots, i^{K_{\max}})}} \\
\{\mathbf{r}, w\} &= \bigcup_A \{\mathbf{r}, w\}_A.
\end{aligned} \tag{3.54}$$

Das folgende Beispiel soll die Vorgehensweise verdeutlichen: nach einer Transformation auf Kugelkoordinaten (r, θ, ϕ) und vor dort auf intrinsische Koordinaten

$$(u_1, u_2, u_3) = \left(U_r(r), \sin \theta, \frac{\phi}{2\pi} \right) \tag{3.55}$$

soll eine numerische Integrationsregel in der intrinsischen „Radialkoordinate“ $\int du_1$ und der Kugeloberfläche $\iint d\Omega$ aufgestellt werden. Die innere Integration soll über den Raumwinkel erfolgen, die äußere über die Radialkoordinate:

$$\begin{aligned}
X_1^1 &= 1, \quad X_1^2 = 2, \quad X_2^2 = 3 \\
\mathbf{K}^1 &= (u_1), \quad \mathbf{K}^2 = (u_2, u_3) \\
U_1 &= K_{11}, \quad U_2 = K_{21}, \quad U_3 = K_{22} \\
I &= \int d\mathbf{K}^1 \int d\mathbf{K}^2 g(\mathbf{K}^1, \mathbf{K}^2) \approx \sum_{i_1} w_{i_1} \sum_{i_2} w_{i_2} g(\mathbf{K}_{i_1}^1, \mathbf{K}_{i_2}^2).
\end{aligned} \tag{3.56}$$

3.4.1. Eindimensionale numerische Integrationsregeln

Die eindimensionale Quadratur ist der einfachste Fall einer numerischen Integrationsregel. Das bestimmte Integral einer Funktion $f(x)$ wird mit einer Genauigkeit ϵ approximiert durch eine gewichtete Summation von Funktionswerten an Stützstellen im Integrationsintervall:

$$\int_a^b f(x) dx = \epsilon + \sum_{i=1}^N f(x_i) w_i \tag{3.57}$$

Der Fehler ϵ wird Integrationsfehler genannt und resultiert aus der Unkenntnis des Funktionsverlaufs zwischen den Stützstellen.

Die Eigenschaft einer guten Integrationsregel ist, dass der Integrationsfehler für eine möglichst große Klasse von Funktionen $f(x)$ minimiert wird. Die Quadraturregeln lassen sich grob in zwei

Klassen einteilen — die Newton-Cotes-Regeln mit äquidistanten und den Gauß-Regeln, bei denen die Stützstellen durch die Regel vorgegeben werden.

Die Herleitungen der Integrationsregeln sind in Standardwerken der numerischen Mathematik [47, 78, 79] behandelt und sollen hier nicht wiederholt werden. Stattdessen sollen die für die Mehrzentren-Integration notwendigen Regeln kurz vorgestellt und lediglich auf Besonderheiten näher eingegangen werden.

Bei allen Integrationsregeln sind die Stützstellen x_i und Gewichte w_i für das Standardintervall $a = -1$, $b = 1$, die Schrittweite h und die Ordnung des Integrationsfehlers angeben. Der Index i läuft von 1 bis N . Da die einfachen Newton-Cotes-Regeln eine feste Zahl von Stützstellen vorgeben, werden hier immer die erweiterten Regeln mit N Punkten beschrieben. Je nachdem, ob einer oder beide Randpunkte des Integrationsintervalls Stützstellen sind, spricht man von einer offenen, halboffenen oder geschlossenen Regel.

3.4.1.1. Die Trapezregel

Die einfachste Integrationsregel ist die Trapezregel, bei der die zu integrierende Funktion durch Trapez-Abschnitte interpoliert wird. Die Fehlerordnung der Trapezregel ist $\epsilon = O\left(\frac{f''}{N^2}\right)$.

Trotz der schlechten Konvergenzordnung der Trapezregeln für gewöhnliche Funktionen, besitzt die halboffene Trapezregel (3.59) die optimale Konvergenz für periodische Funktionen $f(x) = f(x + (b - a))$. Der Beweis dafür findet sich im Anhang B.1.

- offene Trapezregel

$$\begin{aligned} h &= \frac{2}{N+1} \\ x_i &= -1 + h \cdot i \\ w_i &= \begin{cases} 3/2h & \text{für } i = 1 \text{ und } i = N \\ 1h & \text{sonst} \end{cases} \end{aligned} \quad (3.58)$$

- halboffene Trapezregel

$$\begin{aligned} h &= \frac{2}{N} \\ x_i &= -1 + h \cdot (i - 1) \\ w_i &= h \end{aligned} \quad (3.59)$$

- geschlossene Trapezregel [78, Abschnitt 25.4.2]

$$\begin{aligned} h &= \frac{2}{N-1} \\ x_i &= -1 + h \cdot (i - 1) \\ w_i &= \begin{cases} 1/2h & \text{für } i = 1 \text{ und } i = N \\ 1h & \text{sonst} \end{cases} \end{aligned} \quad (3.60)$$

3.4.1.2. Die Simpson-Regel

Die Simpson-Regel [78, Abschnitt 25.4.6] interpoliert den Integranden abschnittsweise durch Parabelstücke durch drei benachbarte Stützstellen. Diese Integrationsregel wird bei der Erstellung der atomaren Basissätze benutzt und kann verwendet werden, um den Molekülcode mit einer Ein-Zentren-Rechnung zu testen. Die Simpson-Regel ist nur für eine ungerade Anzahl von Stützstellen definiert.

$$\begin{aligned}h &= \frac{2}{N-1} \\x_i &= -1 + h \cdot (i-1) \\w_i &= \begin{cases} 1/3h & \text{für } i = 1 \text{ und } i = N \\ 4/3h & \text{für gerade } i < N \\ 2/3h & \text{für ungerade } i > 1 \end{cases} \\ \epsilon &= O\left(\frac{f^{(4)}}{N^5}\right).\end{aligned}\tag{3.61}$$

3.4.1.3. Kubische Quadratur

Analog zur Simpson-Regel können jeweils vier benachbarte Stützstellen durch ein Polynom dritter Ordnung interpoliert werden. Die Abszissen und Gewichte dafür lauten:

$$\begin{aligned}h &= \frac{2}{N-1} \\x_i &= -1 + h \cdot (i-1) \\w_i &= \begin{cases} 3/8h & \text{für } i = 1 \text{ und } i = N \\ 7/6h & \text{für } i = 2 \text{ und } i = N-1 \\ 23/24h & \text{für } i = 3 \text{ und } i = N-2 \\ h & \text{sonst} \end{cases}\end{aligned}\tag{3.62}$$

Eine Regel für die kubische Quadratur, bei der die Randpunkte des Intervalls keine Stützstellen sind, existiert ebenfalls. Ihre Abszissen und Gewichte lauten:

$$\begin{aligned}h &= \frac{2}{N+1} \\x_i &= -1 + h \cdot (i) \\w_i &= \begin{cases} 23/12h & \text{für } i = 1 \text{ und } i = N \\ 7/12h & \text{für } i = 2 \text{ und } i = N-1 \\ h & \text{sonst} \end{cases}\end{aligned}\tag{3.63}$$

3.4.1.4. Gauß Quadratur

Bei den bisherigen Integrationsregeln wurden die zu integrierenden Funktionen auf einem äquidistanten Gitter ausgewertet und mit den Integrationsgewichten multipliziert, die so gewählt waren, dass der Integrationsfehler für Polynome eines vorgegebenen Grades minimiert wurde.

Die Gauß-Quadraturformeln erweitern dieses Verfahren. Zum einen werden nun auch die Gitterpunkte frei gewählt, wodurch zusätzliche Freiheitsgrade geschaffen werden, die die Konvergenz verbessern. Zum anderen werden nicht mehr nur die Integrale von Polynomen approximiert, sondern von Polynomen multipliziert mit einer Gewichtsfunktion $W(x)$. Diese Gewichtsfunktion kann so gewählt werden, dass integrable Singularitäten des Integranden entfernt werden. Die Quadraturformel (3.57) lautet bei der Gauß-Integration nun allgemeiner

$$\int_a^b f(x)W(x)dx = \epsilon + \sum_{i=1}^N f(x_i)w_i. \quad (3.64)$$

In unseren Integralen kommt die Gewichtsfunktion nicht explizit vor. Es gibt aber zwei verschiedene Möglichkeiten, $W(x)$ in den numerischen Gewichten w_i zu „verstecken“:

Zum einen kann die Integrationsformel durch die Ersetzungen

$$g(x) = f(x)W(x), \quad v_i = w_i/W(x_i) \quad (3.65)$$

undefiniert werden, so dass über die Funktion $g(x)$ integriert wird:

$$\int_a^b g(x)dx = \epsilon + \sum_{i=1}^N g(x_i)v_i. \quad (3.66)$$

dadurch wird die Funktion $g(x_i)$ weiterhin an den Stützstellen x_i ausgewertet, die Gewichte ändern sich aber.

Die andere Möglichkeit ist, eine Variablentransformation $du = W(x)dx$ und eine Ersetzung $h(x) = f(x(u))$ auszuführen, durch die sich die Stützstellen w_i transformieren und die Gewichte erhalten bleiben:

$$\begin{aligned} u(x) - u_a &= \int_a^x W(x')dx' \\ \epsilon + \sum_i f(x_i)w_i &= \int_a^b f(x)W(x)dx \\ \epsilon + \sum_i h(u_i)w_i &= \int_{u(a)}^{u(b)} h(u)du \quad \text{mit } u_i = u(x_i). \end{aligned} \quad (3.67)$$

Die Transformation $u(x)$ ist durch die Integrationsgrenzen a, b , die Wahl von $u(a)$ und die Gewichtsfunktion eindeutig definiert. Andere Grenzen als $u(a), u(b)$ für die Integration von $h(u)$ erhält man durch eine zusätzliche Transformation.

Die Gauß-Quadraturformeln für einige spezielle Gewichtsfunktionen $W(x)$ werden häufig angewandt und haben eigene Namen. Diese gehen auf die Namen der Differentialgleichungen zurück, deren Lösungen die mit den Gewichtsfunktionen verbundenen Polynome sind. Die Abszissen entsprechen den n verschiedenen, reellen Nullstellen der Polynome n -ten Grades.

In den folgenden Abschnitten sind die im Programm implementierten Gauß-Quadraturen zusammen mit den Gewichtsfunktionen, den Stützstellen und Gewichten aufgelistet.

3.4.1.5. Gauß-Legendre Quadratur

Die Gauß-Legendre Quadratur wird durch die Gewichtsfunktion $W(x) = 1$ und das Intervall $-1 < x < 1$ definiert. Die Stützstellen sind die Nullstellen der Legendre-Polynome [78, Abschnitt 25.4.29, 25.4.40], für die jedoch kein geschlossener Ausdruck angegeben werden kann. Die Nullstellen werden daher durch ein Eigenwert-Verfahren nach [80] numerisch bestimmt.

3.4.1.6. Gauß-Tschebyscheff Quadratur erster Art

Gegenüber der Gauß-Legendre Quadratur haben die Gauß-Tschebyscheff Quadraturen [78, Abschnitt 25.4.38, 25.4.40] den Vorteil, dass Punkte und Gewichte in geschlossener Form bekannt sind:

$$\begin{aligned} W(x) &= \frac{1}{\sqrt{1-x^2}}, \quad -1 < x < 1 \\ x_i &= \cos \frac{(2i-1)\pi}{2N}, \quad w_i = \frac{\pi}{N}. \end{aligned} \quad (3.68)$$

Die Gewichtsfunktion $W(x)$ wird durch eine Variablentransformation (3.67) eliminiert:

$$\begin{aligned} u(x) &= \pi - \arccos(x) \\ u_i &= u(x_i) = \frac{2\pi(N-i) - \pi}{2N}. \end{aligned} \quad (3.69)$$

Anschließend wird das Integrationsintervall $0 < u < \pi$ durch eine lineare Transformation auf das Standardintervall $-1 < \tilde{u} < 1$ abgebildet. Dabei werden die Stützstellen und Gewichte der halboffenen Trapezregel (3.59) reproduziert:

$$\tilde{u}_i = \frac{N-2i+1}{N}, \quad \tilde{w}_i = \frac{2}{N}. \quad (3.70)$$

3.4.1.7. Gauß-Tschebyscheff Quadratur zweiter Art

Die Gauß-Tschebyscheff Quadratur ist definiert durch die Gewichtsfunktion $W(x) = \sqrt{1-x^2}$. Die Stützstellen sind die Nullstellen der Tschebyscheff-Polynome zweiter Art:

$$x_i = \cos \frac{\pi \cdot i}{n+1}, \quad w_i = \frac{1}{N+1} \sin \frac{\pi \cdot i}{N+1} = \frac{\pi(1-x_i^2)}{N+1}. \quad (3.71)$$

Auch hier kann die Gewichtsfunktion $W(x)$ wieder durch die Transformation (3.67) eliminiert, und u_i auf das Intervall $-1 < \tilde{u} < 1$ transformiert werden:

$$\begin{aligned} u(x) &= \frac{x}{2} \sqrt{1-x^2} - \frac{1}{2} \arccos(x), \quad 0 < u < \frac{\pi}{2} \\ \tilde{u}_i &= \frac{1}{4} \sin \frac{2\pi i}{n+1} - \frac{\pi i}{2(N+1)} \\ \tilde{w}_i &= \frac{2}{\pi(N+1)} \sin \frac{\pi \cdot i}{N+1}. \end{aligned} \quad (3.72)$$

3.4.1.8. Gauß-Lobatto Quadratur

Für einige Anwendungen kann es nützlich sein, wenn bestimmte Stützstellen fest vorgegeben werden. Bei der Gauß-Lobatto Quadratur [78, Abschnitt 25.4.32] sind die Intervallgrenzen festgelegte Abszissen. Eine Anwendung ist die numerische Integration über die Kugeloberfläche mittels einer Produktregel in ϕ und $\cos \theta$ (siehe Abschnitt 3.4.2.2). Die Gewichtsfunktion ist $W(x) = 1$ im Intervall $-1 < x < 1$ und die Integrationsregel lautet somit:

$$\int_{-1}^1 f(x) dx = f(1)w_1 + \sum_i = 2^{N-1} f(x_i)w_i + f(-1)w_N. \quad (3.73)$$

Die Abszissen x_i entsprechen den Nullstellen der Ableitung des Legendre-Polynoms $P_{N-1}(x)$ und Gewichte haben die Werte

$$w_i = \begin{cases} \frac{2}{N(N-1)} & \text{für } i = 1, N \\ \frac{2}{N(N-1)(P_{N-1}'(x_i))^2} & \text{sonst.} \end{cases}. \quad (3.74)$$

3.4.2. Zweidimensionale numerische Integrationsregeln für die Kugeloberfläche

3.4.2.1. Genauigkeit der Integrationsregel

Will man eine numerische Integrationsregel für die Kugeloberfläche anwenden, so muss man zunächst die Anforderungen an die Regel definieren. Für einen allgemeinen Integranden $f(\vec{r})$ mit den entsprechenden analytischen Eigenschaften kann man für einen festen Radius R eine Entwicklung in Kugelflächenfunktionen $Y_l^m(\theta, \phi)$ angeben, so dass gilt:

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=0}^l a_{lm} Y_l^m(\theta, \phi) \quad (3.75)$$

Wird die Entwicklung bei einem bestimmten Drehimpuls l_{\max} abgebrochen, so erhält man eine Approximation der Form

$$f^{l_{\max}}(\theta, \phi) = \sum_{l=0}^{l_{\max}} \sum_{m=0}^l a_{lm} Y_l^m(\theta, \phi) \quad (3.76)$$

wobei die Differenz $|f(\theta, \phi) - f^{l_{\max}}(\theta, \phi)|$ die Genauigkeit der Entwicklung angibt.

Ein Maß für die Genauigkeit einer numerischen Regel für die Integration ist also der höchste Drehimpuls l_{\max} bis zum dem die Werte von F_{num} und F_L :

$$F_{\text{num}} \approx \int f(\theta, \phi) d\Omega \quad (3.77)$$

$$F_L = \int f^{l_{\max}}(\theta, \phi) d\Omega \quad (3.78)$$

exakt übereinstimmen.

3.4.2.2. Zweidimensionale Produktregeln

Wählt man eine ausgezeichnete Z -Achse der Kugeloberfläche, so kann man die Kugelkoordinaten θ und ϕ zur Beschreibung eines Punktes auf der Oberfläche der Kugel einführen. Um eine Kugelflächenfunktion Y_l^m mit einer numerischen Regel in diesen Koordinaten zu integrieren, muss man die Abhängigkeit dieser Funktion von θ und ϕ betrachten.

Die Kugelflächenfunktion Y_l^m ist ein Legendre-Polynom P_l der Ordnung l in $\cos \theta$. Polynome der Ordnung $l \leq L$ werden durch eine Gauß-Regel $(u_i; w_i^u)$ mit $\frac{L+1}{2}$ Punkten exakt integriert.

In der Variablen ϕ ist die Kugelflächenfunktion Y_l^m periodisch mit der Periode 2π , sie ist also proportional zu $e^{im\phi}$. Periodische Funktionen einer Variable lassen sich in eine Fourierreihe $\sum_k a_i \exp(ik\phi)$ entwickeln. Die Funktion Y_l^m besitzt höchstens $m + 1$ nichtverschwindende Fourier-Komponenten a_i mit $0 \leq i < m$.

Für periodische Funktionen ist eine Trapezregel $(v_i; w_i^v)$ am besten geeignet, da sie mit m Stützstellen die ersten m Fourier-Komponenten von f exakt integriert. Der Beweis dazu findet sich im Anhang (B.1).

Schreibt man die numerische Regel nun als Produkt einer Trapez- und einer Gaußregel, so erhält man die Produktregel

$$(\cos \theta_j, \phi_k; w_i^{u,v}) = (u_j, v_k; w_j^u \cdot w_k^v) \quad (3.79)$$

mit $\text{NP}(l) = \frac{l(l+1)}{2}$ Stützstellen.

Die Anzahl der benötigten Stützstellen der Produktregel lässt sich verringern, indem man für die Integration über $\cos \theta$ eine Gauß-Lobatto-Regel verwendet. Bei dieser speziellen Gauß-Regel sind zwei Stützstellen an der Ober- und Untergrenze des Integrationsintervalls vorgewählt. Um Polynome der Ordnung $l \leq L$ mit ungeradem L exakt zu integrieren, benötigt man mit dieser Regel $n_\theta = (L + 3)/2$ Punkte.

Diese Stützstellen der Randpunkte liegen an den „Polen“ $\theta = 0$ und $\theta = \pi$. Dort wird für die Integration über ϕ statt l Stützstellen nur eine einzige benötigt, da alle ϕ_k auf die gleiche kartesische Koordinate abgebildet werden, also auch den gleichen Funktionswert $f(\vec{x}(0, \phi_k))$ bzw. $f(\vec{x}(\pi, \phi_k))$ liefern. Die Gesamtzahl der Stützstellen reduziert sich somit auf $\text{NP}(l) = \frac{l(l-1)}{2} + a$.

Treutler et al. [18] modifizieren die so erhaltene Produktregel, um eine höhere numerische Stabilität zu erhalten. Sie verschieben die Stützstellen jeder zweiten Trapezregel in der ϕ -Koordinate um die Phase π/n_ϕ , so dass die Stützstellen auf benachbarten „Breitengraden“ jeweils um einen halben „Längengrad“ gegeneinander verschoben sind. Die so verschobenen Stützstellen ϕ_k^i haben also die Werte

$$\phi_k^i = \frac{2\pi}{n_\phi}k + \frac{\pi}{n_\phi} \text{mod}(j, 2).$$

3.4.2.3. Spezielle zweidimensionale Regeln

Für die Integration über die Kugeloberfläche kann man zeigen, dass numerische Regeln existieren, die Kugelflächenfunktionen mit $l \leq l_{\max}$ exakt integrieren, die eine minimale Anzahl

$$\text{NP}(L) \approx \frac{(L+1)^2}{3}$$

von Gitterpunkten benötigen.

Ein Ansatz für Integrationsregeln mit näherungsweise optimaler Punktzahl wurde von Sobolev [81,82] gemacht und von Lebedev [83–89] weiterentwickelt. Bei dem Verfahren wird eine maximale Ordnung L und eine endliche Punktgruppe vorgegeben, unter der sich die Knoten der Regel transformieren. In einer Basis von reellen Kugelflächenfunktionen, die zur total symmetrischen Darstellung der Punktgruppe gehören und für deren Ordnung l gilt $l \leq L$, muss die numerische Integrationsregel nun nur noch über ein irreduzibles Teilstück der Kugeloberfläche integrieren. Die Knoten und Gewichte für die numerische Integration über dieses Teilstück werden analog zu den eindimensionalen Gauß-Regeln nicht vorgegeben. Sie werden als Lösungen eines nichtlinearen Gleichungssystems bestimmt, welches aus der Forderung resultiert, dass die Kugelflächenfunktionen durch die Regel exakt integriert werden müssen.

Diese Gleichungen können i. A. nur durch ein numerisches Verfahren gelöst werden. Lebedev hat in seinen Veröffentlichungen Lösungen der Gleichung für die Symmetrie-Gruppe T_d und den Ordnungen von $L = 3$ bis $L = 59$ angegeben. Die Punkte und Gewichte sind in seinen Tabellen auf zehn Stellen für $L \leq 53$ und auf zwölf Stellen für $L = 59$ angegeben.

3.4.3. Automatische Integration

Bei der Aufstellung einer numerischen Integrationsregel möchte man normalerweise nicht die Anzahl der Stützstellen vorgeben, sondern eine Toleranz ϵ , mit der ein Integral approximiert wird. Möchte man, wie in unserem Fall, die Integrationsregel auf mehrere Integrale angewandt werden, so soll diese Toleranz für alle Integrale eingehalten werden. Eine allgemeingültige Abschätzung für die Toleranz kann nur für spezielle Integrale gefunden werden. In unserem Fall wird das Problem zusätzlich dadurch erschwert, dass die Integrale zum Zeitpunkt der Aufstellung der Integrationsregel noch gar nicht bekannt sind.

Um dieses Problem lösen zu können, wurde die Methode der automatischen Integration in Verbindung mit einer Menge von Testfunktionen gewählt. Dabei wird die Genauigkeit einer Integrationsregel abgeschätzt, indem Testfunktionen mit einer unterschiedlichen Anzahl von Stützstellen integriert werden. Die Differenz der Ergebnisse gibt einen Anhaltspunkt für die Genauigkeit der Regeln. Dieses Verfahren hat sich als zuverlässig erwiesen, falls die Testfunktionen den wirklichen Integranden ähnlich sind. Das Verfahren wird nun im Einzelnen beschrieben.

Ein automatischer numerischer Integrator ist ein Programm oder Algorithmus, der eine numerische Approximation I_{n_k} mit n_k Stützstellen eines Integrals $I = \int f(x)dx$ liefert, deren relativer numerischer Fehler

$$\epsilon_k = \frac{|I_{n_k} - I|}{I} \quad (3.80)$$

vermutlich kleiner als eine vorgegebene relative Toleranz ϵ ist [90]. Die Methode, mit der der automatische numerische Integrator arbeitet, ist sehr einfach. Das Programm erzeugt eine Sequenz I_{n_1}, I_{n_2}, \dots von immer genaueren Approximationen des exakten Wertes. Nach jeder neuen Approximation I_{n_k} wird der Fehler ϵ_k abgeschätzt und die Sequenz beendet, falls die Toleranz unterschritten worden ist, d. h. $\epsilon_k < \epsilon$ ist.

3.4.3.1. Fehlerabschätzung

Die Verfahren zur Bestimmung des geschätzten Fehlers ϵ_k unterscheiden sich darin, welche Integrale I_{n_k} evaluiert werden, und wie sie in die Berechnung von ϵ_k eingehen. Werden mehrere Funktionen $f^{(j)}(x)$ zur Abschätzung herangezogen, so muss aus den Einzelabschätzungen $\epsilon_k^{(j)}$ eine Gesamtaberschätzung bestimmt werden. In dem Integrationsprogramm ist die Abschätzung

$$\begin{aligned}\epsilon_k^{(j)} &= \frac{|I_{n_k}^{(j)} - I_{n_{k+1}}^{(j)}|}{I_{n_{k+1}}^{(j)}} \\ \epsilon_k &= \max_j \epsilon_k^{(j)}\end{aligned}\tag{3.81}$$

implementiert. Der Gesamtfehler der Integrationsregel I_{n_k} wird also als Maximum der relativen Einzelabweichungen zur nächst genaueren Integrationsregel $I_{n_{k+1}}$ bestimmt.

3.4.3.2. Auswahl der Sequenz

Bei der Auswahl der Sequenz n_1, n_2, \dots müssen mehrere Bedingungen beachtet werden. Zum einen muss die numerische Integrationsregel I_{n_k} für alle n_k der Sequenz definiert sein. Dies ist insbesondere bei den mehrdimensionalen Integrationsregeln für die Kugeloberfläche (Abschnitt 3.4.2) wichtig, da diese durch die Ordnung L und nicht durch die Punktzahl beschrieben sind. Zum anderen muss die „Schrittweite“ zwischen zwei Punktzahlen richtig gewählt werden. Erhöht man die Punktzahl zu schnell, so liefert der automatische numerische Integrator mit größerer Wahrscheinlichkeit eine Regel mit einer viel höheren als der verlangten Genauigkeit und entsprechend unnötig vielen Stützstellen. Wird die Punktzahl in der Sequenz dagegen zu langsam erhöht, so wird sehr viel Rechenzeit für die Testintegrationen verbraucht, bis man bei einer hinreichenden Genauigkeit angelangt ist.

Bei den eindimensionalen Integrationsregeln gibt man eine Startpunktzahl n_1 vor, alle weiteren Punktzahlen werden nach dem folgenden Schema bestimmt:

- Multiplikation der Punktzahl n_k mit einem Faktor C und Runden zur nächsten natürlichen Zahl²:

$$n_{k+1} = \lfloor C \cdot n_k + \frac{1}{2} \rfloor.\tag{3.82}$$

- falls n_{k+1} gleich dem Wert von n_k ist, wird n_{k+1} um eins erhöht. Dies verhindert eine Endlosschleife bei zu kleinem C oder zu geringer Anfangspunktzahl n_1 .

²Hier wird die praktische Notation von Iverson [91] benutzt: Der Term $\lfloor x \rfloor$ hat den Wert der größten ganzen Zahl, die kleiner oder gleich x ist.

- falls n_{k+1} gerade ist und die Simpson-Regel (3.61) verwendet werden soll, wird n_{k+1} um eins erhöht, da diese Integrationsregel nur für eine ungerade Zahl von Stützstellen definiert ist.

Bei den zweidimensionalen Integrationsregeln für die Kugeloberfläche wird versucht, die Ordnung L in jedem Schritt um eins zu erhöhen.

Bei den Integrationsregeln von Lebedev (Abschnitt 3.4.2.3) wird jeweils die nächsthöhere tabellierte Regel benutzt, falls für das entsprechende L keine Regel verfügbar ist. Ist keine höhere Ordnung mehr tabelliert und die geforderte Genauigkeit noch nicht erreicht, so wird die letzte tabellierte Regel verwendet und eine Warnung ausgegeben.

Bei den zweidimensionalen Produktregeln (Abschnitt 3.4.2.2) können beliebige Ordnungen generiert werden, diese benötigen allerdings eine höhere Anzahl von Stützstellen als die Integrationsregeln von Lebedev.

Eine Kombination beider Verfahren benutzt immer dann eine Produktregel, wenn keine Lebedev-Regel für die benötigte Ordnung tabelliert ist und erreicht so eine geringe Punktzahl, ist aber nicht in der Genauigkeit beschränkt.

3.4.3.3. Testfunktionen

Das Verfahren der automatischen Integration arbeitet nur wirksam, wenn die Testfunktionen den eigentlichen Integranden gut angepasst sind. Das Integrationssystem stellt daher verschiedene Sätze von Testfunktionen zur Verfügung, die miteinander kombiniert werden können:

- Atomare Dichten

Die atomaren Basisfunktionen und Besetzungszahlen können benutzt werden, um eine Testfunktion für die atomare Elektronendichte zu bilden:

$$f(\mathbf{r}) = \sum_{nljm_j} \left| \psi_{A,nljm_j}^{\text{AO}}(\mathbf{r} - \mathbf{r}_A^0) \right|^2 \quad (3.83)$$

- Summe der kugelsymmetrischen, atomaren Dichten

Durch Summation der atomaren Dichtefunktionen aller Atome wiederum kann eine „molekulare“ Testfunktion gebildet werden:

$$f(\mathbf{r}) = \sum_A \psi_A^{\dagger \text{AO}}(\mathbf{r} - \mathbf{r}_A^0) \psi_A^{\text{AO}}(\mathbf{r} - \mathbf{r}_A^0) \quad (3.84)$$

Diese Funktionen sind bereits gut geeignet, um realistische Ergebnisse zu erhalten. Allerdings fehlen hier die Winkelanteile der atomaren Wellenfunktionen, so dass noch zusätzliche Testfunktionen hinzugenommen werden sollten.

- Überlapp-Funktionen

Die Überlapp-Testfunktion zweier atomarer Basisfunktionen $\psi_{A,(nljm_j)_A}^{\text{AO}}$ und $\psi_{B,(nljm_j)_B}^{\text{AO}}$ ist definiert als:

$$f(\mathbf{r}) = \left| \psi_{A,(nljm_j)_A}^{\dagger \text{AO}}(\mathbf{r} - \mathbf{r}_A^0) \psi_{B,(nljm_j)_B}^{\text{AO}}(\mathbf{r} - \mathbf{r}_A^0) \right| \quad (3.85)$$

- Slater-Funktionen

Slater-Funktionen haben im Gegensatz zu den Gauß-Funktionen einen Cusp am Kernort und fallen exponentiell für große r ab. Sie sind daher geeignet, das Verhalten der atomaren Basisfunktionen zu approximieren und sind wie folgt definiert:

$$f(\mathbf{r}) = Y_{lm}(\theta, \phi) r^{n-1} e^{-\alpha r} \quad (3.86)$$

Die Kugelkoordinaten (r, θ, ϕ) beziehen sich dabei auf den Ort \mathbf{r}_A^0 des A -ten Kerns.

Mit Hilfe von Slater-Funktionen kann eine Klasse von Testfunktionen generiert werden, die die Aufstellung einer gut angepassten, adaptiven Integrationsregel für das gesamte Molekül ermöglicht. Dazu wird für jedes Atom A zunächst für jede Hauptquantenzahl n der höchste vorkommende Drehimpuls $L_{A,n}^{\max}$ der atomaren Basis $\psi_{A,nl_i}^{\text{AO}}$ bestimmt. Die Testfunktionen sind nun die Slater-Funktionen

$$\begin{aligned} & \left\{ Y_{lm}(\theta, \phi) r^{n-1} e^{-\alpha_t^{(n)} r} \right\} & (3.87) \\ l & = 2L_{A,n}^{\max} \\ m & = -L_A^{\max} \dots L_A^{\max} \\ t & = 1 \dots N_t \end{aligned}$$

Die Exponenten $\alpha_t^{(n)}$ werden so gewählt, dass die Slater-Funktionen zur Hauptquantenzahl n gleichmäßig temperiert sind. Dies bedeutet, dass die Exponenten aufsteigend geordnet sind und der „Überlapp“ der Radialanteile zweier Slater-Funktionen mit benachbartem Index des Slater-Exponenten einen konstanten Wert annimmt:

$$C = \int r^{n-1} e^{-\alpha_t^{(n)} r} \cdot r^{n-1} e^{-\alpha_{t+1}^{(n)} r} dr. \quad (3.88)$$

Die Konstante C wird nach dem Schema von Baerends [10] so gewählt, dass der erste und der letzte Exponent die Werte

$$\begin{aligned} \alpha_1^{(n)} & = \frac{3}{2} \max\left(\frac{1}{2}, n-1\right) \\ \alpha_{N_t}^{(n)} & = \max\left(\max(\alpha_1^{(n)}, 1), \frac{2Z_A}{\max(\frac{n}{2}, 2)}\right) \end{aligned} \quad (3.89)$$

annehmen. Die Slater-Funktion mit dem Exponenten $\alpha_1^{(n)}$ hat ihr Maximum im Valenzbereich und ist die diffuseste Funktion, während die Funktion mit dem Exponenten $\alpha_{N_t}^{(n)}$ ein Maximum bei kleinen r hat, welches durch die Kernladung bestimmt ist. Abschirmeffekte werden ab $n > 4$ durch den Nenner des Exponenten berücksichtigt. Als Vorgabe werden pro Hauptquantenzahl $N_t = 4$ Slater-Funktionen getestet.

Hinzu kommen einige Testfunktionen, deren Integrale bekannt sind, aber nur zum Test der Algorithmen dienen, da sie nicht an das physikalische Problem angepasst sind:

- Gauß-Funktionen

Funktionen von Gauß-Typ kommen in den Molekülintegralen nicht vor, sind aber zu Testzwecken implementiert, da der exakte Wert des Integrals bekannt ist. Sie sind in der Implementation immer am Ort \mathbf{r}_A^0 eines Kerns zentriert und haben eine Reichweite r_m :

$$f(\mathbf{r}) = e^{-\frac{(\mathbf{r}-\mathbf{r}_A^0)^2}{r_m^2}} \quad (3.90)$$

- Konstante Funktion

Mit Hilfe einer konstanten Funktion

$$f(\mathbf{r}) = c \quad (3.91)$$

kann das „Volumen“ einzelner Zellen durch Integration berechnet werden. Da in einem endlichen Molekül immer Randzellen mit unendlichem Volumen existieren, ist das Integral über den gesamten \mathbb{R}^3 nicht definiert.

- Exponentialfunktion

Die Exponentialfunktion ist eine weitere Funktion zum Programmtest:

$$f(\mathbf{r}) = e^{-|\mathbf{r}-\mathbf{r}_A^0|} \quad (3.92)$$

3.4.3.4. Mehrdimensionale adaptive Integration

Sollen mehrere adaptive numerische Integratoren nach dem Schema von Gleichung (3.52) geschachtelt werden, so ergibt sich eine zusätzliche Schwierigkeit. Will man für das Gesamtintegral eine Toleranz ϵ vorgeben, so müssen daraus die Toleranzen ϵ^K der Einzelintegrale gebildet werden. Es ist klar, dass die Gesamt toleranz nicht besser als ϵ^1 sein kann. Um für die äußere Integration aber diese Toleranz zu erreichen, müssen die inneren Integrale alle höchstens diese Toleranz besitzen, im Allgemeinen jedoch genauer integriert werden, da die Toleranz nur einen Schätzwert für den Fehler darstellt.

Die Wahl der Toleranzen wird in der Literatur als Interface-Problem bezeichnet [16, 22, 92]: Wählt man ϵ^K zu klein gegenüber ϵ^{K+1} , so wird aufgrund der numerischen Ungenauigkeit der Approximation des inneren Integrals die Genauigkeit ϵ^K des äußeren Integrals unter Umständen nie erreicht. Es wird daher in [92] empfohlen, $\epsilon^K \gg \epsilon^{K+1}$ zu wählen. Für unsere Zwecke hat sich die Wahl

$$\epsilon^K = (2 \dots 5) \cdot \epsilon^{K+1} \quad (3.93)$$

als stabil erwiesen. Wählt man den Wert für die Toleranz der inneren Integration zu gering im Vergleich zur Toleranz der äußeren Integration, so wird für die inneren Integrale ein zu großer numerischer Aufwand betrieben, der durch die ungenaue äußere Integration wieder zunichte gemacht wird.

3.5. Ausnutzung der Molekülsymmetrie

In diesem Abschnitt soll gezeigt werden, wie durch die Ausnutzung der Molekülsymmetrie die numerische Integrationsregel für die molekularen Integrale vereinfacht werden kann. Diese Vereinfachung ist unabhängig von der Wahl einer Basis von Symmetrie-Orbitalen (Abschnitt 2.9), in der durch gruppentheoretische Überlegungen gezeigt werden kann, dass alle Integrale, die nicht zur total symmetrischen (A_1) Darstellung der Punktgruppe des Moleküls gehören, identisch verschwinden und daher nicht berechnet werden müssen. Stattdessen wird die numerische Integrationsregel für die verbleibenden Integrale dadurch reduziert, dass nicht mehr über den vollen Raum integriert werden muss, sondern nur noch über einen irreduziblen Teilraum des \mathbb{R}^3 .

Die Überlegungen zur Reduktion der numerischen Regel beruhen auf der Annahme, dass alle zu berechnenden Integranden

$$F(\mathbf{r}) = \psi_i^\dagger(\mathbf{r})\hat{O}(\mathbf{r})\psi_j(\mathbf{r}), \quad (3.94)$$

symmetrisch unter der molekularen Punktgruppe G sind. Dies ist der Fall, wenn das direkte Produkt der Darstellungen der Wellenfunktionen und Operatoren in Gleichung (3.94) lediglich eine total symmetrische Darstellung von G enthält.

Diese Voraussetzung ist in den meisten Fällen gegeben, Ausnahmen sind aber bekannt, so dass sie im Einzelfall überprüft werden sollte. Notwendig für die Symmetrie des Integranden ist z. B., dass die Grundzustands-Dichte $\rho(\mathbf{r})$ und damit auch die von der Dichte abhängigen und in $\hat{O}(\mathbf{r})$ enthaltenen Funktionale die Molekülsymmetrie besitzen.

In diesem Fall gilt für alle Symmetrioperationen $g \in G$:

$$F(\mathbf{r}) = F(g\mathbf{r}). \quad (3.95)$$

Eine Möglichkeit, diese Invarianz auszunutzen wäre, nur noch über einen irreduziblen Teilraum $W \subseteq \mathbb{R}^3$ zu integrieren, der aus einer Menge von Punkten \mathbf{x} besteht, für die gilt:

$$\forall \mathbf{y} \in \mathbb{R}^3 \quad \exists g \in G, \mathbf{x} \in W : \quad g\mathbf{x} = \mathbf{y}. \quad (3.96)$$

Eine numerische Integrationsregel für einen Unterraum des \mathbb{R}^3 aufzustellen ist jedoch sehr schwierig. Wesentlich einfacher ist es, zunächst eine Integrationsregel für den gesamten \mathbb{R}^3 zu erstellen und erst dann die Symmetrie zu ausnutzen, um nur die invarianten Integrationspunkte der numerischen Regel beizubehalten, da die entsprechenden symmetrischen Punkte bei der Evaluation des Integranden die gleichen Funktionswerte liefern.

Allgemein lässt sich die Reduktion der symmetrie-äquivalenten Integrationspunkte wie folgt formulieren: ist eine Menge P von Stützstellen und Gewichten $(\mathbf{r}, w) \in P$, $\mathbf{r} \in \mathbb{R}^3$ symmetrisch unter einer Untergruppe $H \subseteq G$ der Symmetrie-Gruppe des Moleküls und gehört zu allen Stützstellen \mathbf{r} der Menge P das gleiche Gewicht w , so liefert jeder Integrationspunkt den gleichen Beitrag zur Teilsumme

$$S = \sum_i F(\mathbf{r}_i)w_i \quad \text{mit } (\mathbf{r}, w)_i \in P \quad (3.97)$$

der gesamten Integrationsregel. Die Summe S kann demnach ersetzt werden durch den Wert

$$S = NF(\mathbf{r}_1)w_1, \quad (3.98)$$

der nur noch die Funktionsauswertung eines (beliebigen) invarianten Teilpunktes benötigt. Der Vorfaktor N ist die Zahl der zu r_1 symmetrischen Stützstellen, d. h. die Ordnung der Untergruppe H .

Die Voraussetzung, dass jedem symmetrie-äquivalenten Integrationspunkt $r \in P$ das gleiche Gewicht w zugeordnet ist, ist erfüllt, wenn die Zellenfunktion $z_A(r(\mathbf{u}))$ und die Jacobi-Determinante der Transformation $\left| \frac{\partial r_A(\mathbf{u})}{\partial \mathbf{u}} \right|$ in Gleichung (3.47) für jeden dieser Punkte übereinstimmt. Eine hinreichende Bedingung dafür ist, dass die Zellenfunktion und die Jacobi-Determinante invariant unter der Gruppe G ist, also eine symmetrische Zellenfunktion und eine Transformation gewählt wird, die für alle symmetrie-äquivalenten Zellen gleich ist.

Die Reduktion der numerischen Integrationsregel kann für die praktische Anwendung noch weiter verbessert werden. Damit nicht zuerst eine Integrationsregel für alle Zellen aufgestellt und anschließend die symmetrischen Punkte wieder verworfen werden müssen, lässt sich der Prozess in zwei Schritte aufteilen. Bei einer symmetrischen Raumaufteilung $z_A(r)$ liefert jede äquivalente Zelle den gleichen Beitrag zur Summe (3.97). Die Zellen lassen sich daher in Symmetrieklassen unterteilen und die numerische Regel muss nur für irreduzible Zellen generiert werden. Innerhalb jeder inäquivalenten Zelle z_A kann es nun noch symmetrische Integrationspunkte $(r, w) \in P_A$ geben, die im zweiten Schritt wie in Gleichung (3.98) zusammengefasst werden:

$$S = N_A N F(r_{A,1}) w_{A,1}. \quad (3.99)$$

Der hier hinzugekommene Faktor N_A ist die Anzahl der symmetrie-äquivalenten Zellen zur Zelle A .

3.6. MULIN — Das Mehrzentren-Integrationsprogramm

Bei der Implementation der vorgestellten Verfahren im Programmpaket MULIN (MULTicenter INtegration) wurden verschiedene Anforderungen berücksichtigt, die während der Entwicklungsphase, im normalen Betrieb und für spätere Erweiterungen wichtig sind. Auf die Details soll in diesem Abschnitt eingegangen werden.

Für den Benutzer ist es wichtig, dass die umfangreichen Parameter komfortabel gesetzt werden können und dass sinnvolle Voreinstellungen existieren. Das Programm sollte möglichst ohne Kenntnis der unterliegenden Verfahren benutzbar sein, und alle Parameter, die schon in der Eingabe-Datei des Molekülprogramms vorkommen, sollten nicht noch einmal in der Eingabe-Datei des Integrationsprogramms vorkommen. Trotzdem sollte das Integrationsprogramm zum Testen auch ohne den Molekülcode ablauffähig sein.

Dazu wurde eine Eingabesprache entworfen, die ähnlich wie eine Programmiersprache linear abgearbeitet wird. Die einzelnen Elemente der Eingabesprache sind im Anhang C.1 detailliert beschrieben. Die Sprache kennt verschiedene Arten von Befehlen. Zum einen sind dies Befehle, die Parameter des Integrationssystems setzen, wie z. B. die Zellenfunktion, die Transformation auf intrinsische Koordinaten oder die numerischen Integrationsregeln. Bei den numerischen Parametern sind auch algebraische Ausdrücke zugelassen. Weiterhin gibt es Steuerbefehle, die das Programm veranlassen, Berechnungen durchzuführen oder Ergebnisse auszugeben.

Eingabebefehle lesen externe Dateien ein, wie zum Beispiel die atomaren Basissätze, oder auch weitere Befehle in der MULIN-Sprache, die vom Molekülcode zur Beschreibung des molekularen Systems erzeugt werden. Die Übergabe der Parameter zwischen RELMOS und MULIN geschieht ausschließlich über Textdateien, die aber nicht in die Integrationsdatei von MULIN eingebunden werden müssen. Dadurch ist die Möglichkeit gegeben, die Vorgaben von RELMOS nach den eigenen Vorstellungen zu verändern. Dieser Weg des Datenaustausches über Dateien anstatt über Parameter einer funktionalen Schnittstelle wurde bewusst gewählt, um den Integrationscode auch unabhängig von Molekülcode betreiben zu können.

Die letzte Gruppe von Befehlen sind Diagnosebefehle zur Fehlersuche. Mit Hilfe dieser Befehle lässt sich vorgeben, in welchem Umfang Informationen und Fehlermeldungen ausgegeben werden. Alle Meldungen des Programms sind in acht verschiedene Klassen eingeteilt, die von Hinweisen über Warnungen bis zu echten Fehlermeldungen gehen. Welche dieser Meldungen ausgegeben werden und bis zu welcher Schachtelungstiefe der Unterprogramme dies geschehen soll, lässt sich einstellen.

Das Programm stellt hohe Anforderungen an die Speicherverwaltung. Dies resultiert einerseits aus dem Verfahren der automatischen Integration (Abschnitt 3.4.3), welches immer wieder verschieden dimensionierte Integrationsgitter vergleicht, als auch aus der Variabilität der Integrationsreihenfolge, der Unkenntnis über die Größe der eingelesenen Basissätze und der endgültigen Zahl der Integrationspunkte bei der Verwendung adaptiver Integrationsregeln. Im Programm wird daher durchgehend eine dynamische Speicherverwaltung verwendet, wenn Array-Größen nicht zur Kompilierzeit festliegen. Dies hat den Vorteil, dass das Programm im Hauptspeicher auch nur den wirklich benötigten Speicherplatz belegt. Dynamische Speicherallokation hat allerdings den Nachteil, dass Programmierfehler oftmals zur Belegung von Speicher führen, der später nicht mehr freigegeben wird. Es wurde beim Entwurf versucht, diesen Mangel durch einen objektorientierten Ansatz zu beheben. Alle dynamisch allozierten Objekte besitzen einen Konstruktor zur Allokation und einen Destruktor zur Deallokation des Speichers im Sinne der objektorientierten Programmierung.

Weiterhin wurde bei der Programmierung auf gute Vektorisierbarkeit geachtet. Eine Programmoperation ist vektorisierbar, wenn sie auf mehrere Daten gleichzeitig ausgeführt werden muss. Vektorisierbare Operationen kommen in MULIN vor allem dann immer vor, wenn mehrere Stützstellen oder Integrationsregeln bearbeitet werden müssen. Im Programm werden daher durchgehend Vektoren von Stützstellen und Integrationsstrukturen an Unterprogramme übergeben. Keine der Unterfunktionen wird für einzelne Gitterpunkte aufgerufen. Eine gute Vektorisierbarkeit ist außerdem durch eine Standardisierung der hauptsächlich verwendeten Schleifenstruktur gewährleistet, die durch Änderung einiger weniger Makro-Befehlen an den jeweiligen Kompiler angepasst werden kann.

Das Programm wurde auch auf eine eventuelle zukünftige Parallelisierung entworfen. Dies wird beim Übergang zu großen Systemen wichtig, bei denen die Bearbeitung einzelner atomarer Zellen auf verschiedene Knoten eines Rechnerclusters verteilt werden soll. Die Skalierbarkeit muss dann natürlich auch für die Erstellung der Integrationsregel gewährleistet sein. Im Programm wurde daher auf globale Variablen vollständig verzichtet. Weiterhin wurde eine Schnittstelle geschaffen, mittels der Daten für die Integration einzelner Zellen oder bei Bedarf sogar einzelner Unterintegrale über eine einzige Datenstruktur abläuft. Diese Datenstruktur enthält alle zur Berechnung notwendigen Parameter beim Aufruf und auch die Ergebnisse der Berech-

nung bei der Rückgabe. Eine Parallelisierung mittels des Message Passing Interfaces (MPI) oder des Parallel-Virtual-Machine (PVM)-Systems ist dadurch direkt möglich.

Da das Programm sowohl als Teil des Molekülstrukturprogramms `RELMOS`, als auch als eigenständiges Programm benutzbar sein sollte, wurde es als Bibliothek konzipiert, die entweder zu `RELMOS` oder zu einem Frontend für die interaktive Ein/Ausgabe hinzugelinkt werden kann. Damit die Einbindung in das Molekülprogramm für verschiedene Plattformen und Compiler portabel bleibt, wurde das Makefile von `RELMOS` und `MULIN` so gestaltet, dass durch die Auswahl eines Satzes von Compilern und Linkern mittels eines einzelnen Parameters die Binärkompatibilität von Objekt- und Bibliotheksdateien sichergestellt ist.

Das Programmpaket `MULIN` ist in der Programmiersprache ANSI C geschrieben und hat einen Umfang von etwa 20000 Zeilen in 160 Quellcode-Dateien. Es benötigt die Standard-Parsergeneratoren `lex` und `yacc` oder alternativ die GNU-Programme `flex` und `bison`, falls die Syntax der Eingabesprache abgeändert werden soll.

4. Testrechnungen

Anhand verschiedener Testsysteme soll festgestellt werden, in wie weit das neue Mehrzentren-Integrationsverfahren gegenüber dem bisherigen Verfahren einsetzbar ist. Da unser Molekülstrukturprogramm besonders für hochrelativistische Systeme entwickelt wurde, wurden für die Testrechnungen ein schweres Atom sowie Moleküle mit schweren Kernen ausgewählt. Die Testsysteme unterscheiden sich in der Dimensionalität: das atomare System ist kugelsymmetrisch, als lineares System wurde ein Gold-Dimer gewählt und echt dreidimensionale Rechnungen wurden an RfCl_4 und einem Ad-Atom auf der Oberfläche eines Gold-Clusters durchgeführt.

Eine gute Genauigkeit bei der Lösung der Dirac-Gleichung ist hierbei notwendig, da die Bindungsenergie als Differenz zwischen den totalen Energien der separierten und gebundene Atome berechnet wird. Gerade bei Systemen mit schweren Konstituenten unterscheiden sich aber die Bindungsenergie und die totale Energie des Moleküls um viele Größenordnungen. Der relative numerische Fehler muss daher klein gegen das Verhältnis zwischen Bindungs- und totaler Energie sein.

Diese Argument wird beim Übergang zu Systemen mit vielen Atomen gewichtiger, da die Bindungsenergie pro Atom etwa gleich bleibt, während die totale Energie annähernd linear mit der Zahl der Atome wächst. Von praktischer Bedeutung wird dieses Problem, wenn man die Bindungsenergie von Ad-Atomen auf Cluster-Oberflächen berechnen will [3].

4.1. Atomare Rechnungen: Rf

Atomare Rechnungen mit dem Molekülprogramm dienen in erster Linie dazu, das Verfahren zu testen und eventuelle numerische Fehler im Programmcode aufzudecken. Im Gegensatz zum Integrationsschema von Baerends kann mit unserem Integrationscode die volle Kugelsymmetrie des atomaren Systems ausgenutzt werden, die Symmetrie muss nicht durch Angabe von Dummy-Zentren reduziert werden. Bei einer atomaren Rechnung wird daher praktisch nur die Radialintegration getestet, die Zellenfunktion (3.13) ist hier konstant gleich eins für den ganzen \mathbb{R}^3 und die Integration über den Raumwinkel lässt sich mit einer geringen Punktzahl exakt im Rahmen der Genauigkeit des Molekülstrukturprogramms ausführen.

Die atomaren Rechnungen können auch der Vorbereitung auf die Rechnungen an molekularen Systemen dienen, man bekommt aus den Ergebnissen einen Anhaltspunkt für die Anzahl der radialen Stützstellen, die zum Erreichen einer bestimmten Genauigkeit mindestens nötig sind.

Für die erste Testrechnung wurde das Element Rf ($Z = 104$) ausgewählt, die atomare Basis wurde an 1000 Radialpunkten eines logarithmischen Gitters mit den Grenzen $5.4 \cdot 10^{-9} a_0$ und $58 a_0$ tabelliert. Der Basissatz wurde als neutrale, minimale Basis gewählt mit den Orbitalen bis $7s_{1/2}$. Als Dichtefunktional wurde das relativistische Austausch-Korrelations-Funktional

von Vosko-Wilk-Nusair verwendet.

In den molekularen Rechnungen wurden die Besetzungszahlen der Orbitale festgehalten und die Anzahl der radialen Stützstellen variiert. Als Radialtransformation wurde die Transformation aus Gleichung (3.42) mit den Parametern $t_0 = 1$, $m_1 = 2$, $x_m = 4$, $\beta_1 = 2$, $\beta_2 = 0.8$, $r_m = 20 a_0$, $r_{\min} = 0 a_0$ und $r_{\max} = 58 a_0$ verwendet.

Für die Winkel-Integration wurde für alle Abstände eine zweidimensionale numerische Regel mit der Ordnung $L = 11$ mit 38 Gitterpunkten verwendet, die alle vorkommenden Winkelanteile exakt integriert.

Diskussion der Ergebnisse

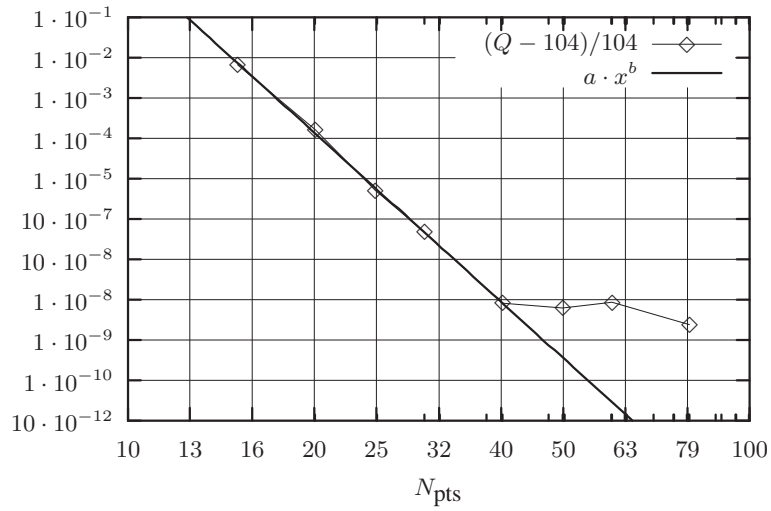
Tabelle 4.1 zeigt die Abhängigkeit der Größen „Gesamtladung“ und „totale Energie“ von der Zahl der radialen Stützstellen.

Trägt man den Logarithmus der Funktion $\epsilon = |Q - 104|/104$ über dem Logarithmus der Punktzahl n_{rad} auf, so erkennt man, dass der Zusammenhang einem Potenzgesetz $\epsilon = ax^b$ genügt. Ein Least-Square-Fit ergibt die Parameter $\log_e a = 32.8$ und $b = -13.9$. Dies bedeutet, dass man die Zahl der Radialpunkte um dem Faktor $\sqrt[13.9]{10} = 1.18$ erhöhen muss, um eine weitere Stelle zu gewinnen.

Eine analoge Rechnung für die totale Energie ergibt die Parameter $\log_e a = 26$ und $b = -11.6$. Um die Größenordnung der Genauigkeit der totalen Energie also um eine Stelle zu verbessern, muss man die Zahl der Radialpunkte um den Faktor 1.22 erhöhen. Abbildung 4.1 stellt das Konvergenzverhalten graphisch dar.

Man erkennt, dass die Gesamtladung nur bis zu einer relativen Genauigkeit von 10^{-8} und die totale Energie nur bis auf 10^{-9} bestimmt werden kann. Eine genaue Überprüfung hat gezeigt, dass dies mit den numerischen Basissätzen zusammenhängt, die durch den Atom-Code generiert worden sind. Die Basis wird mit diesem Programm an einer Anzahl von Stützstellen tabelliert, die vom Molekülcode interpoliert werden müssen. Im Atom-Code wird zur Integration die erweiterte Simpson-Regel für das Radial-Gitter verwendet, im Molekülcode werden die Funktionswerte durch das Lagrange-Verfahren interpoliert. Beide Verfahren machen unterschiedliche Annahmen über das Verhalten der Funktion zwischen den Stützstellen, so dass die Radialintegration unterschiedliche Ergebnisse liefert. Eine Lösung dieses Problems würde darin bestehen, von den numerischen Basissätzen wegzugehen und einen Satz von Radialfunktionen zu benutzen, deren Werte für jedes r ohne Interpolation direkt berechnet werden können. Hierfür würden sich beispielsweise die Slater-Orbitale (STO) anbieten.

Genauigkeit der Gesamtladung als Funktion der Punktzahl



Genauigkeit der totalen Energie als Funktion der Punktzahl

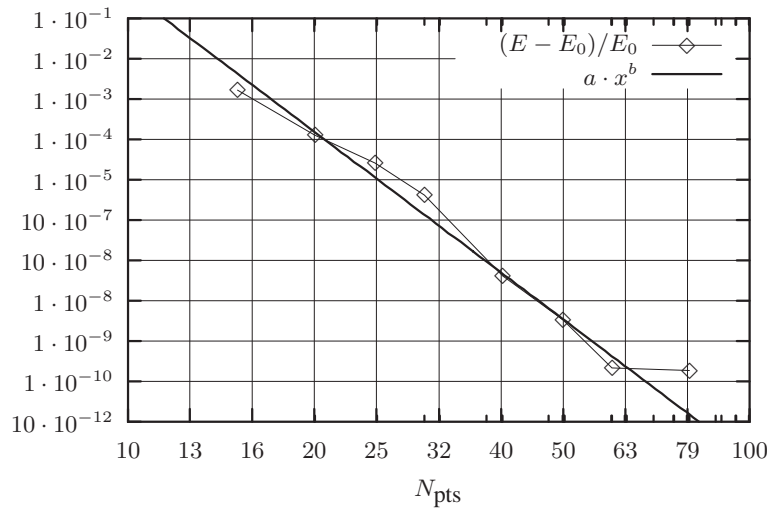


Abbildung 4.1.: Konvergenz von Gesamtladung und totaler Energie für Rf.

Molekülrechnung für das Rf-Atom		
n_{rad}	Q	$E_{\text{tot}}/E_{\text{h}}$
15	103.31	-38 580
20	104.0 17	-3865 5.7
25	104.000 53	-3865 1.7
30	103.9999 50	-38650.87
40	104.000000 85	-38650.71 53
50	103.999999 35	-38650.71 701
60	104.000000 90	-38650.7168 72
80	103.999999 74	-38650.71688 81
Atom	104.000000000	-38650.7168809

Tabelle 4.1.: Abhängigkeit von Gesamtladung und totaler Energie von der Zahl der radialen Stützstellen

4.2. Dimere: Au₂

Bei zweiatomigen Systemen sind viele physikalisch interessante Größen vom Abstand der beiden Atome abhängig. Experimentell leicht zu bestimmen ist der Gleichgewichts-Abstand im Grundzustand sowie die Bindungsenergie des Dimers. Variiert man den Abstand der Atome, so ergeben die berechneten Werte der totalen Energie in Abhängigkeit des Kernabstands die Potentialkurve, aus der Bindungsenergie und -länge, sowie einfache Aussagen über das Vibrationspektrum abgeleitet werden können. Von großer physikalischer Bedeutung ist dabei der Zustand der Hüllenelektronen, da diese für die chemische Bindung verantwortlich sind. Trägt man den Verlauf der Einteilchenenergien in Abhängigkeit des Bindungsabstandes auf, so erhält man das Korrelations-Diagramm, welches für Ion-Atom-Stöße Informationen über den Ladungstransfer oder das molekulare Röntgenspektrum [93].

Um unser Verfahren an einem zweiatomigen System zu überprüfen, bietet sich das System Au₂ an, da verlässliche experimentelle Ergebnisse und Vergleichsrechnungen zur Verfügung stehen. Die Rechnungen an diesem System sollen die Genauigkeit verschiedener physikalischer Größen in Abhängigkeit der Integrationsgenauigkeit und dem Integrationsverfahren vergleichen.

4.2.1. Die Referenzrechnung

Zunächst wurde eine Testrechnung mit dem Baerends-Integrationscode in einer hohen Genauigkeit durchgeführt und diese Werte als Referenz benutzt. Als Basis wurde dazu ein Satz von numerischen Wellenfunktionen aus der atomaren Rechnung für Au verwendet. Die Basis enthält die Wellenfunktionen $1s_{1/2}$ bis $6s_{1/2}$, den höchsten vorkommenden Drehimpuls besitzt die $4f_{3/2}$ -Wellenfunktion.

Die Berechnung der Potentialkurve wurde durch Variation des Kernabstandes in Schritten von $0.1 a_0$ bestimmt. Dabei wurde jeweils die totale Energie mit dem RLDA Austausch-Korrelationspotential nach dem Erreichen des Konvergenzlimits von $10^{-4} E_h$ bestimmt.

Die typische Anzahl von SCF-Iterationsschritten beträgt 50 bei kleinen Abständen und verringert sich auf 20 bei großen Abständen. Der Wert der totalen Energie zeigt vor dem Erreichen des Limits eine exponentielle Konvergenz, so dass der SCF-Grenzwert durch Anfitzen einer Exponentialfunktion bestimmt wurde (Abbildung 4.2).

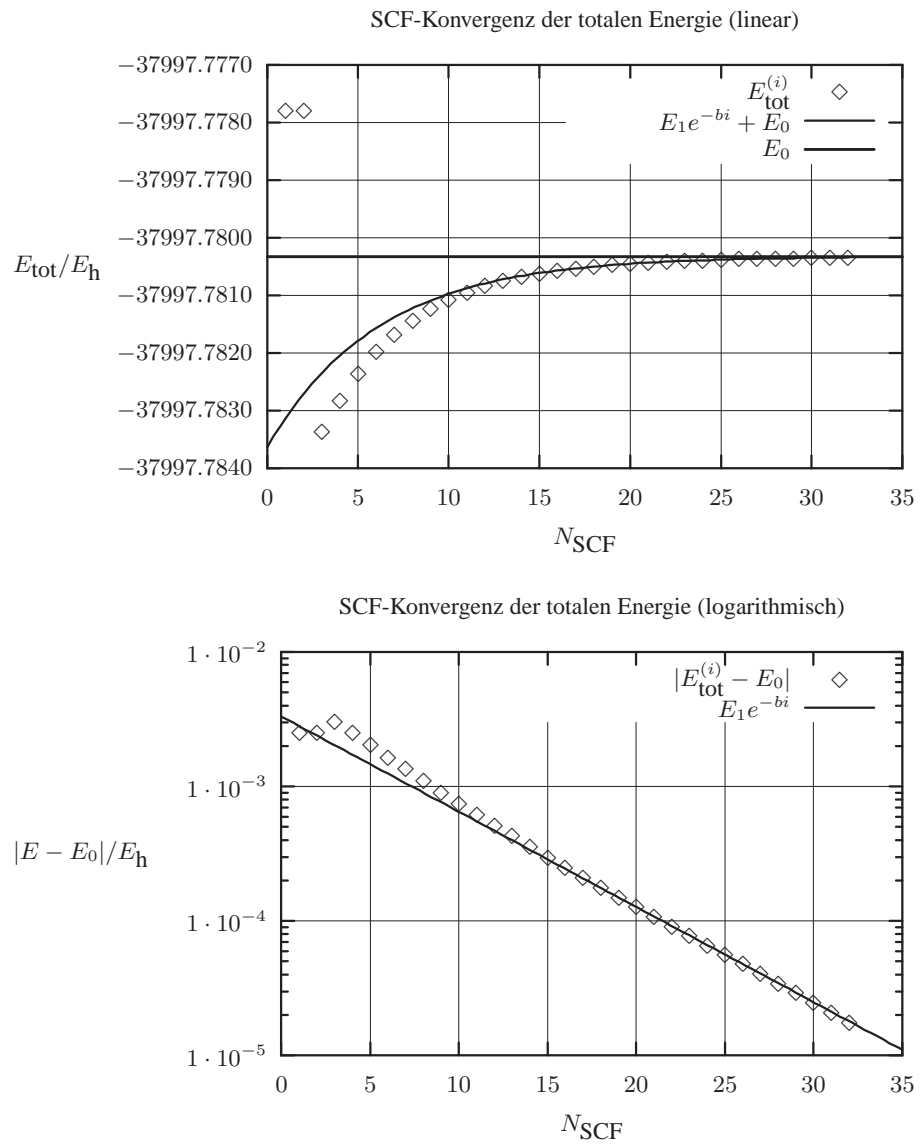


Abbildung 4.2.: Au₂: exponentielle Konvergenz des SCF-Prozesses am Beispiel der Rechnung für den interatomaren Abstand 4.6 a_0 . Im oberen Diagramm ist der Energiemaßstab linear, unter ist die absolute Abweichung vom extrapolierten Wert E_0 logarithmisch über der Nummer des SCF-Zyklus aufgetragen.

Die Grafik 4.3 zeigt das Ergebnis. Um die genauen Koordinaten des Minimums zu bestimmen, wurde zwischen $4.4 a_0$ und $5.0 a_0$ eine Parabel $f(x) = a(x - x_0)^2 + b$ angefitet. Der berechnete Gleichgewichts-Abstand beträgt $(4.78 \pm 0.01) a_0^1$, die totale Energie im Minimum hat den Wert $-37997.781 E_h$. Da der absolute Wert der totalen Energie in DFT-Rechnungen keine große Verlässlichkeit besitzt, wurde zusätzlich die Bindungsenergie bestimmt, indem an die Potentialkurve eine Morse-Funktion der Form

$$f(x) = b_0 + b_d \left[1 - \exp\left(\frac{x - x_0}{b_r}\right) \right]^2 \quad (4.1)$$

angepasst wurde. Tabelle 4.2 zeigt die Resultate der Fits. Die Bindungsenergie entspricht dem Parameter b_d und hat einen Wert von $(2.979 \pm 0.011) \text{ eV}$.

Parabel-Fit			
$a/(E_h a_0^{-2})$	b/E_h	x_0/a_0	
0.078 ± 0.008	-37997.7808 ± 0.0005	4.78 ± 0.01	
Morse-Fit			
b_0/E_h	b_d/E_h	x_0/a_0	b_r/a_0
-37997.7793 ± 0.0003	0.1095 ± 0.0004	4.7671 ± 0.0009	1.42 ± 0.01

Tabelle 4.2.: Au_2 : Parameter der Fits an die Potentialkurve. Alle Werte sind in atomaren Einheiten angegeben.

Aus der Krümmung der Parabel lässt sich unter der Annahme einer harmonischen Streck-schwingung die Schwingungsfrequenz bestimmen. Für die Kraftkonstante D des harmonischen Oszillators gilt:

$$V(x) = \frac{1}{2} D(x - x_0)^2. \quad (4.2)$$

Die Frequenz ω der Grundschwingung ergibt sich dann aus der Lösung der DGL für den klassischen harmonischen Oszillator mit der reduzierten Massen μ :

$$\mu \frac{d^2 x}{dt^2} + D(x - x_0) = 0 \rightarrow \omega^2 = \frac{D}{\mu} \quad (4.3)$$

In der Literatur wird die Schwingungsfrequenz üblicherweise nicht in Hz, sondern als Wellenzahl $\tilde{\omega} = \omega/c$ in der Einheit cm^{-1} angegeben. Die Umrechnungsformel für dimensionslose Einheiten $\tilde{\mu}$, \tilde{D} lautet dann:

$$\tilde{\mu} := \mu/u \quad (4.4)$$

$$\tilde{D} := D/(E_h a_0^{-2}) \quad (4.5)$$

$$\tilde{\omega} = 32298.634 \text{ cm}^{-1} \sqrt{\frac{\tilde{D}}{\tilde{\mu}}} \quad (4.6)$$

Setzt man die Zahlen aus Tabelle 4.2 ein, so erhält man einen Wert von $\tilde{\omega} = 909 \text{ cm}^{-1}$.

¹Die angegebene Toleranz ist ein Ergebnis der Fitprozedur, sie enthält keine Information über die Ungenauigkeit der eigentlichen Rechnung

4.2.2. Vergleichsrechnungen für konstanten Abstand

Die erste Gruppe von Vergleichsrechnungen wurde bei einem festen internuklearen Abstand von $4.8 a_0$ durchgeführt. In den beiden nächsten Abschnitten wird beschrieben, wie die einzelnen Parameter aufgesetzt wurden. Mit diesen Parametern wurde eine Reihe von Rechnungen mit verschiedenen Genauigkeiten ausgeführt und im dritten Abschnitt verglichen.

Bestimmung der Integrationsparameter: Baerends

Für das Baerends-Integrationsschema wurde der Genauigkeitsparameter ϵ_B für die Integration in den atomaren Kugeln, im Zwischenraum und im Außenraum jeweils auf den gleichen Wert gesetzt. Die Genauigkeit wurde, beginnend mit dem minimal erlaubten Wert von 10^{-3} , jeweils um den Faktor 10 gesteigert. Die höchste Genauigkeit wird für $\epsilon_B = 10^{-11}$ erreicht, danach bricht das Programm ab. Alle weiteren Parameter wurden so wie in Abschnitt 4.2.1 beschrieben gewählt.

Bestimmung der Integrationsparameter: MULIN

Um die Integration mit unserem Schema durchzuführen, sind einige Vorüberlegungen notwendig.

Bei Au_2 handelt es sich um ein homonukleares Dimer. Die zu integrierenden Funktionen besitzen eine Rotationssymmetrie um die internukleare Verbindungsachse. Die symmetrieangepassten Linearkombinationen der Basisfunktionen machen die Integration um den Azimutalwinkel ϕ der durch die Rotationsachse definierten Zylinderkoordinaten (z, r, ϕ) unnötig. Sie setzen sich aus Funktionen zusammen, die invariant unter der Gruppe D^∞ sind. Die Integration muss daher lediglich in der z - r -Halbebene mit $z > 0$ erfolgen. Die Symmetrie-Orbitale enthalten allerdings nicht die Information über die Spiegelsymmetrie des Moleküls, so dass über alle z im Intervall $(-\infty, \infty)$ integriert werden muss.

Um die Rotationssymmetrie auszunutzen, wurden die Z -Achsen der beiden Integrationszellen parallel zur Z -Achse des Zylinderkoordinatensystems gelegt. Da der Molekülcode Dimere immer so umorientiert, dass der geometrische Schwerpunkt im Ursprung und die Verbindungsachse parallel zur x -Achse des \mathbb{R}^3 liegt, wurde für beide Integrationszellen das Kugelkoordinatensystem durch die Rotationsmatrix

$$\mathbf{R} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

gedreht. Das Vorzeichen von \mathbf{R}_{22} stellt sicher, dass es sich bei der Transformation um eine eigentliche Rotation handelt, die die Händigkeit des Koordinatensystems nicht verändert.

In den Kugelkoordinaten (r, θ, ϕ) der beiden Zellen wurde für das Dreifachintegral die Reihenfolge

$$\int dr \int d\theta \int d\phi \tag{4.7}$$

gewählt. Das innere Integral über ϕ kann aufgrund der C^∞ -Symmetrie der Integranden in der Symmetrie-Basis mit einem einzelnen Integrationspunkt ausgeführt werden, was durch die Wahl der halboffenen Trapezregel mit $n = 1$ erreicht wird. Die Integration ist damit auf den zweidimensionalen Unterraum der r - θ -Ebene zurückgeführt.

In r läuft die Integration über das Intervall $[0, \infty)$, muss aber nur über ein endliches Teilintervall $[0, r'_{\max}]$ ausgeführt werden, da die tabellierten numerischen Radialanteile der atomaren Basisfunktionen, die in allen Integranden als Faktor vorkommen, nur eine endliche Reichweite r_{\max} besitzen. Die Radialfunktionen sind aber am Ort der beiden Kerne zentriert, so dass die äußere Integrationsgrenzen als

$$r'_{\max} = r_{\max} + d$$

gewählt werden, wobei d den internuklearen Abstand bezeichnet. Dieses Intervall wird durch die Transformation (3.42) auf die intrinsische Radialkoordinate abgebildet, über welches mit einer Gauß-Tschebyscheff-Regel integriert wird.

In θ schließlich muss über das Intervall $[0, \pi]$ integriert werden, diese Integration durch eine Gauß-Legendre-Regel in der Variablen $\cos \theta$ ausgeführt. Die Zerlegung in die beiden Integrationszellen erfolgt mit Hilfe der Zellenfunktion von Delley [14] mit der Dichte $\rho(\vec{x})$, die aus der Summe der radialsymmetrischen atomaren Dichten der Basisfunktion gewonnen wurde.

Die Bestimmung der für die Integration in $\cos \theta$ notwendigen Zahl der Stützstellen erfolgte durch automatische Integration mit Vorgabe einer Genauigkeit für einen Satz von Testfunktionen. Dazu gehörte die Summe der atomaren Dichten und Slater-Funktionen, deren Integral analytisch berechnet und verglichen werden kann.

Für die Bestimmung der Anzahl der Stützstellen in r sind zwei verschiedene Vorgehensweisen denkbar. Zum einen kann auch in dieser Variablen die Zahl der Stützstellen durch eine automatische Integration bestimmt werden. Da die r -Integration in unserer Rechnung das äußere Integral darstellt, ist es nicht sinnvoll eine höhere Genauigkeit als für die θ -Integration vorzugeben. Testrechnungen zeigen, dass eine Genauigkeit im Bereich $\epsilon_r = 0.5\epsilon_\theta \dots 1.0\epsilon_\theta$ gute Ergebnisse liefert. Problematisch hinsichtlich der benötigten Rechenzeit wird diese Methode, wenn eine hohe Genauigkeit gefordert wird, oder in einer echt dreidimensionalen Integration die Punktzahl für alle Variablen automatisch bestimmt werden soll.

Für eine systematische Untersuchung ist es sinnvoll, die Anzahl der Stützstellen in r zunächst festzuhalten und die Genauigkeit in θ zu variieren. Trägt man die relative Genauigkeit der Gesamtladung oder der totalen Energie über der Genauigkeit der θ -Integration auf, so erwartet man, dass für eine vorgegebene Zahl von radialen Stützstellen das Ergebnis nicht mehr verbessert wird, auch wenn man die Genauigkeit in θ erhöht. Abbildung 4.4 zeigt den Zusammenhang.

Aus diesen Kurven ist es allerdings nicht leicht, die optimalen Parameter für eine vorgegebene Genauigkeit abzulesen. Dies wird durch einen anderen Ansatz einfach. Im allgemeinen skaliert die numerische Genauigkeit der Ergebnisgrößen über eine Potenzfunktion mit der Anzahl der Stützstellen:

$$\epsilon \propto (n_{\text{pts}})^p \quad (4.8)$$

Der Exponent p stellt ein Maß für die Geschwindigkeit der Konvergenz mit der Erhöhung der Punktzahl n_{pts} dar und wird Konvergenzrate genannt. Trägt man p über der Gesamtpunktzahl

auf, so erkennt man, dass es einen optimalen Zusammenhang zwischen ϵ_θ und n_r gibt (Abbildung 4.5). Wählt man die Anzahl der Radialpunkte zu klein, so verringert sich die Genauigkeit, wählt man sie zu groß, so werden unnötig viele Punkte für die Radialintegration benötigt, die nicht zu einer Verbesserung der Genauigkeit beitragen.

Die Auswertung der Kurvenschar in Abbildung 4.5 ergibt den gesuchten Zusammenhang zwischen n_r und ϵ_θ . Dazu wird zunächst für jede $n_r = \text{const.}$ -Kurve durch einen Parabel-Fit das ϵ_θ bestimmt, bei dem p maximal ist, d. h. eine optimale Konvergenzrate erreicht wird. Abbildung 4.6 zeigt die so errechnete Kurve für die totale Energie.

Man erkennt einen linearen Zusammenhang zwischen n_r und $\log_{10}(\epsilon_\theta)$. Eine Ausgleichsgerade durch die Punkte besitzt die folgende Form:

$$\begin{aligned} -\log_{10}(\epsilon_\theta) &= 0.0525 \cdot n_r + 0.772 \\ \Leftrightarrow n_r &= -19.04 \cdot \log_{10}(\epsilon_\theta) - 14.71 \end{aligned} \quad (4.9)$$

Diskussion der Ergebnisse

Da sowohl beim Verfahren von Baerends, als auch in unserem Schema mit adaptiver Punktzahl, die Zahl der Stützstellen nicht direkt gewählt werden kann, werden die Ergebnisse sowohl einzeln in Tabellenform, als auch gemeinsam in einem Plot überlagert dargestellt.

Das einfachste Resultat ist die Berechnung der elektronischen Gesamtladung als Integral der Ladungsdichte. Das Ergebnis ist für die Überlagerung der atomaren Dichten bekannt und entspricht der Zahl der Elektronen. Da das Molekülprogramm in den SCF-Iterationen dieses Integral als variationale Nebenbedingung mit der Elektronenzahl gleichsetzt, ist die Genauigkeit dieser Größe wichtig. Die Tabellen 4.2.2 und 4.2.2 sowie die Abbildung 4.7 zeigen die Ergebnisse für die Gesamtladung und für die totale Energie.

Bei der Gesamtladung erkennt man, dass das beide Integrationsverfahren die Gesamtladung etwa bis auf 10^{-9} integrieren können. Eine weitere Erhöhung der Punktzahl ist beim Baerends-Schema nicht möglich und führt in unserem Schema nicht zu einer höheren Genauigkeit.

Bei der totalen Energie liegt die Unsicherheit der genauesten Rechnungen in der fünften Nachkommastelle. Mit dem Baerends-Verfahren wird dieses Ergebnis mit dem Genauigkeitsparameter 10.0 erreicht, die erzeugte Integrationsregel besitzt etwa 2300 Punkte. Unser Verfahren liefert bereits mit 620 Integrationspunkten dieses Ergebnis.

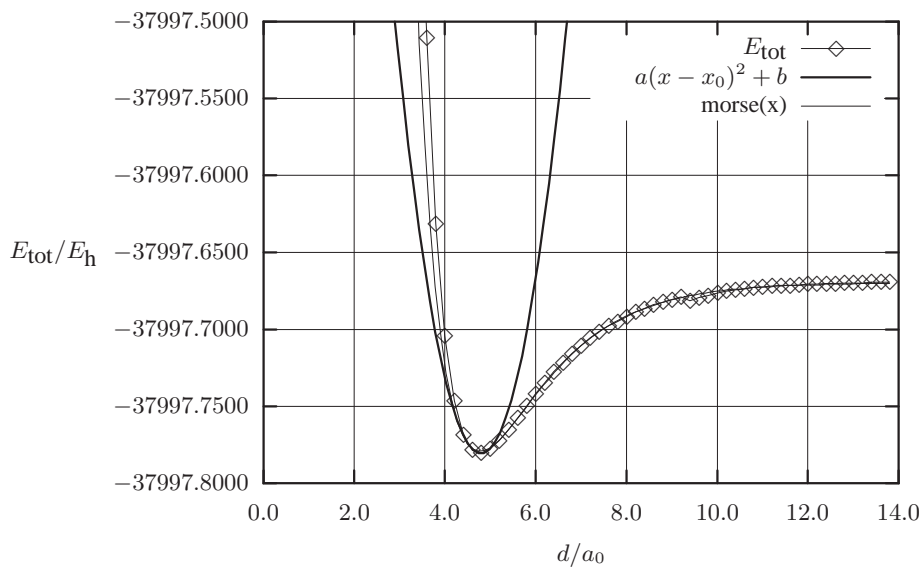


Abbildung 4.3.: Au₂: Potentialkurve (RLDA). Neben den berechneten totalen Energien wurde eine Parabel im Energieminimum und eine Morse-Funktion angefitet.

ϵ_θ	n_{rad}	n_{tot}	Q	E_{tot}/E_h
$1.0 \cdot 10^{-2}$	23	138	158.00 27	-37997. 85
$0.5 \cdot 10^{-2}$	29	190	158.000 62	-37997.7 76
$1.0 \cdot 10^{-3}$	42	330	157.999 79	-37997.7 73
$0.5 \cdot 10^{-3}$	48	402	158.0000 68	-37997.78 26
$1.0 \cdot 10^{-4}$	61	620	157.99999 34	-37997.7803 15
$1.0 \cdot 10^{-5}$	80	1152	157.999999 37	-37997.7803 16
$1.0 \cdot 10^{-6}$	100	2590	157.9999999 45	-37997.7803 09
$1.0 \cdot 10^{-7}$	119	6690	158.00000000 65	-37997.7803 77

Tabelle 4.3.: Au₂: Gesamtladung in totale Energie in Abhängigkeit der Zahl der Stützstellen (MULIN)

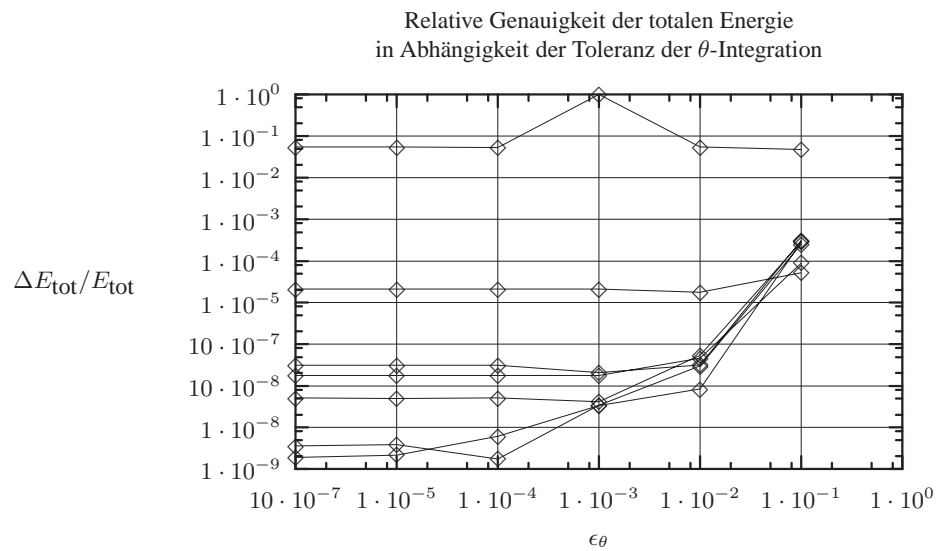
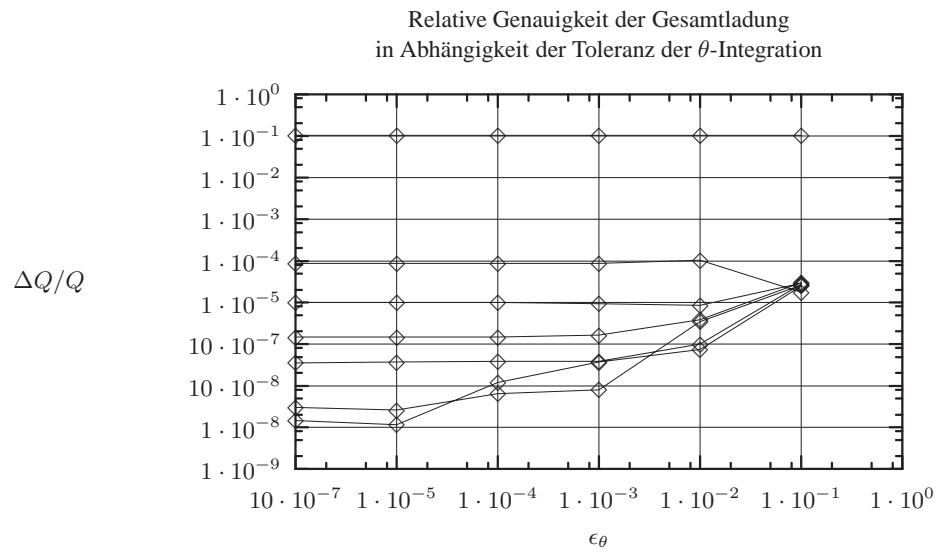


Abbildung 4.4.: Au₂: Genauigkeit der Gesamtladung und der totalen Energie in Abhängigkeit der Punktzahl in r und der Toleranz der θ -Integration. Die Linien verbinden jeweils Werte von Rechnungen mit gleicher Punktzahl in r , welche von 10 (obere Kurve) bis 70 (untere Kurve) variiert wurde.

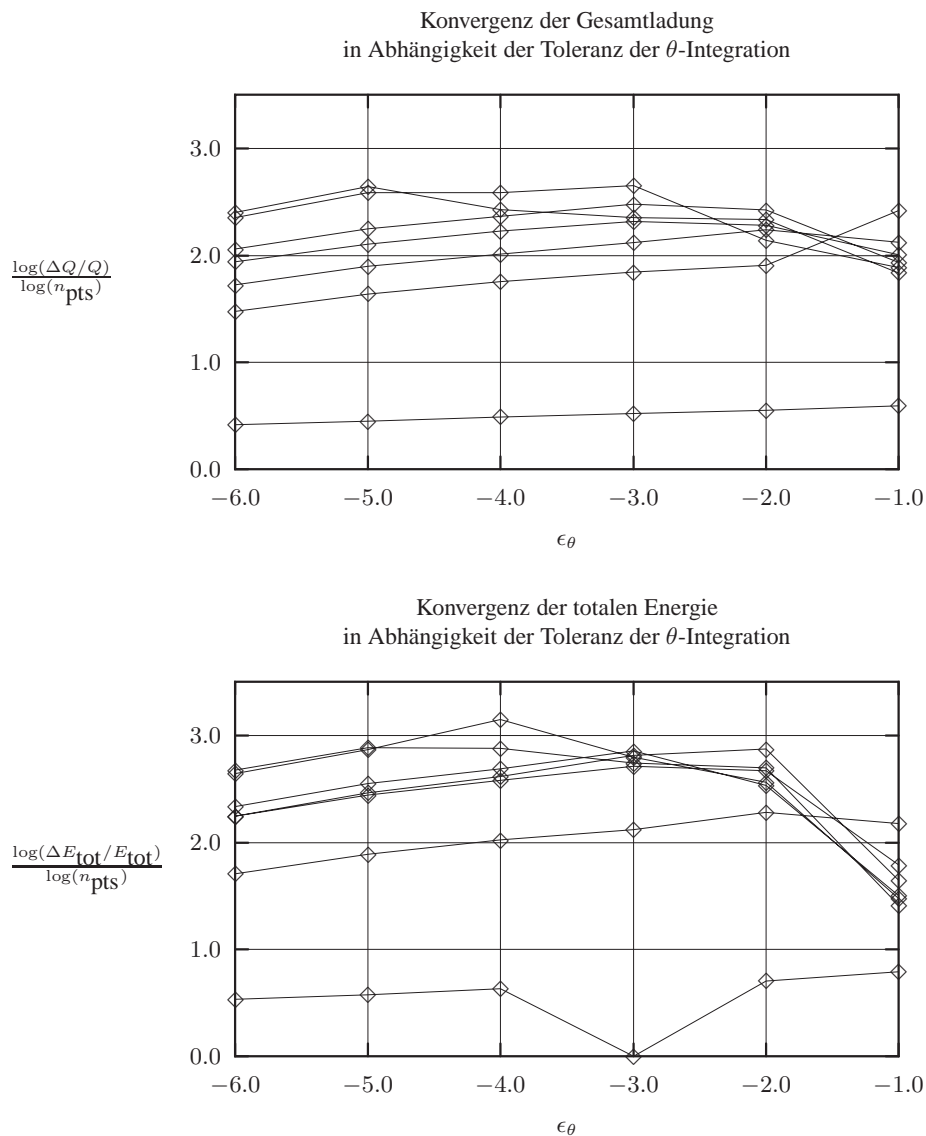


Abbildung 4.5.: Au₂: Konvergenzrate für die Gesamtladung und die totale Energie. Aufgetragen ist die Konvergenzrate p über dem Logarithmus der Genauigkeit der θ -Integration. Rechnungen mit gleicher Anzahl von Radialpunkten sind verbunden. Die untersten Kurven haben $n_r = 10$, die obersten $n_r = 70$ Stützstellen.

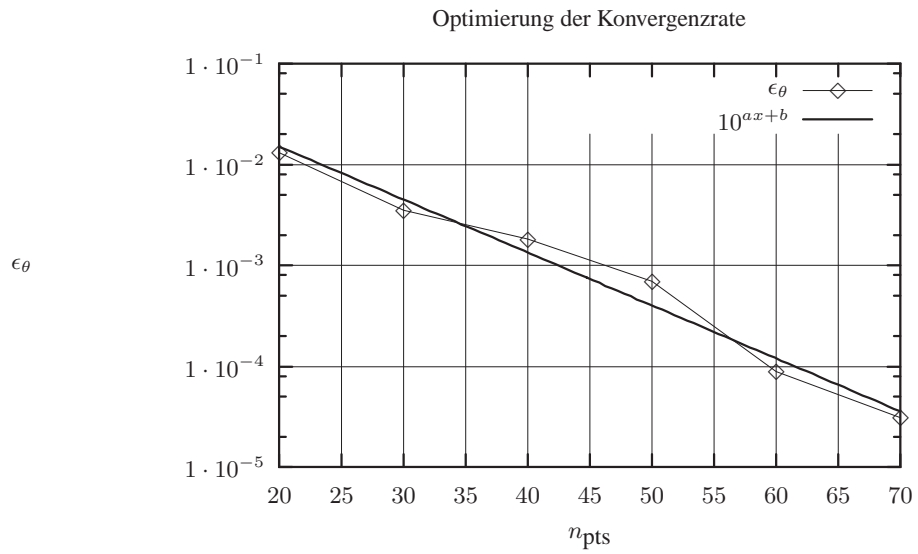


Abbildung 4.6.: Au₂: Optimierung der Konvergenzrate. Für die totale Energie aus Abbildung 4.5 ist ϵ_θ mit der höchsten Konvergenzrate über der Zahl der radialen Stützstellen n_r aufgetragen.

ϵ_B	n_{tot}	Q	E_{tot}/E_h
$1.0 \cdot 10^{-3}$	426	158.49	-38 330
$1.0 \cdot 10^{-4}$	539	158.091	-379 76
$1.0 \cdot 10^{-5}$	758	157.984	-3799 4.9
$1.0 \cdot 10^{-6}$	953	158.00 18	-37997. 26
$1.0 \cdot 10^{-7}$	1176	157.999 83	-3799 8.0
$1.0 \cdot 10^{-8}$	1417	158.0000 22	-37997.7 25
$1.0 \cdot 10^{-9}$	1796	158.000000 15	-37997.78 16
$1.0 \cdot 10^{-10}$	2324	158.000000 49	-37997.7803 22
$1.0 \cdot 10^{-11}$	4791	158.000000 33	-37997.7803 18

Tabelle 4.4.: Au₂: Gesamtladung in totale Energie in Abhängigkeit der Zahl der Stützstellen (Baerends)

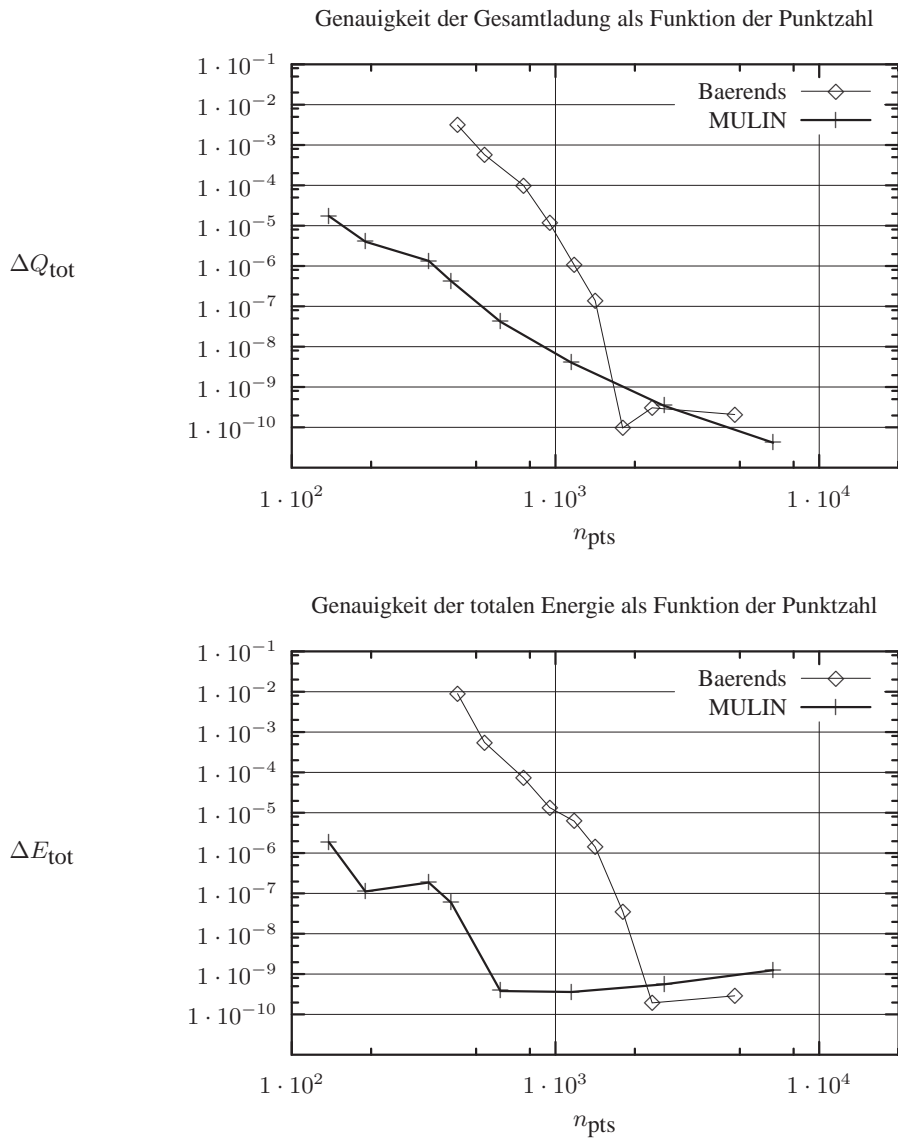


Abbildung 4.7.: Au₂: Konvergenz von Gesamtladung und totaler Energie.

4.3. Kleine Moleküle: RfCl_4

Die Metall-Tetrachloride stellen einen interessanten Test für die Integrationsroutinen dar. Nimmt man an, dass die Kernanordnung T_d -Symmetrie besitzt, so gibt es nur noch einen geometrischen Parameter, von der die Potentialenergiekurve abhängt. Trotzdem ist das Molekül dreidimensional und somit muss auch die Integration über den vollen \mathbb{R}^3 erfolgen. Wenn die

Integration die Symmetrie des Moleküls ausnutzt, kann die Zahl der Integrationspunkte dennoch reduziert werden.

4.3.1. Die Referenzrechnung

Für das zentrale Metallatom wurde das schwere Rf ($Z=104$) verwendet. Auch für dieses System wurde zunächst eine Referenz-Rechnung mit dem Baerends'schen Integrationscode durchgeführt. Die Rf-Basisfunktionen enthalten die Wellenfunktionen des einfach neutralen Atoms von $1s_{1/2}$ bis $7s_{1/2}$. Die Basisfunktionen des Chloratoms wurden zusammengesetzt aus einer -0.2 -fach ionisierten Basis, die die Funktionen $1s_{1/2}$ bis $2p_{3/2}$ enthält und einer Zusatzbasis vom $+2$ -fach ionisierten Chlor mit den Wellenfunktionen $3d_{3/2}$, $3d_{5/2}$ und $4s_{1/2}$.

Die Rechnung wurde mit dem Baerends-Schema in einer hohen Genauigkeit für verschiedene M-Cl-Abstände in Schritten von $0.1 a_0$ durchgeführt. Die Abbildungen 4.3.1 zeigt die entsprechenden Potentialenergiekurve für RfCl_4 .

4.3.2. Vergleichsrechnungen für konstanten Abstand

Für den Vergleich der beiden Integrationsschemata wurde der Abstand $d = 4.6 a_0$ für RfCl_4 ausgewählt, der nahe beim RLDA-Energieminimum liegt.

Bestimmung der Integrationsparameter: Baerends

Die Rechnungen für das Baerends-Schema wurden wieder wie in Abschnitt 4.2.2 beschrieben mit sukzessiv erhöhtem Parameter ϵ_B ausgeführt. Auch hier ist $\epsilon_B = 10^{-11}$ die höchste Genauigkeit, für die der Baerends-Code Integrationspunkte erzeugen kann.

Bestimmung der Integrationsparameter: unser Schema

In unseren Rechnungen setzen wir voraus, dass das RfCl_4 -Molekül eine T_d -Symmetrie besitzt. Diese hohe Symmetrie führt zu einer erheblichen Reduktion der Zahl der Integrationspunkte, da die Integrationszelle des zentralen Atoms durch alle 24 Operatoren der Symmetrie-Gruppe auf sich selbst abgebildet wird, die Integrationszellen der Cl-Atome alle symmetrie-äquivalent sind und jede einzelne diese Zellen noch symmetrisch unter der Gruppe C_{3v} bezüglich der Verbindungsachse Rf-Cl ist. Sowohl die Achsen der Operatoren der T_d -Symmetrie, als auch die der C_{3v} -Symmetrie stimmen mit denen der Lebedev-Integrationsgitter für die Kugeloberfläche überein, falls die Z-Achsen aller Integrationszellen parallel zu einer der Koordinatenachsen gewählt werden. In diesem Fall wird durch die Symmetrie des Moleküls die Anzahl der Integrationspunkte minimiert.

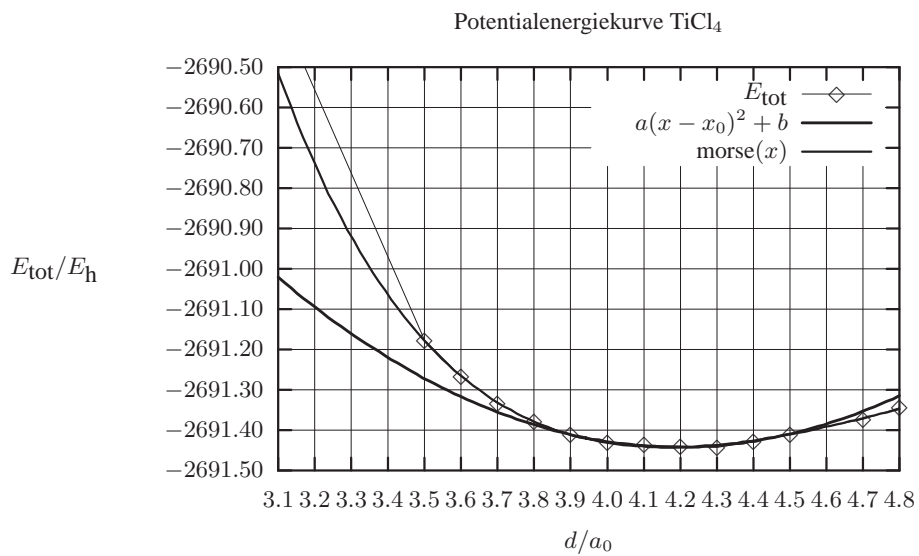
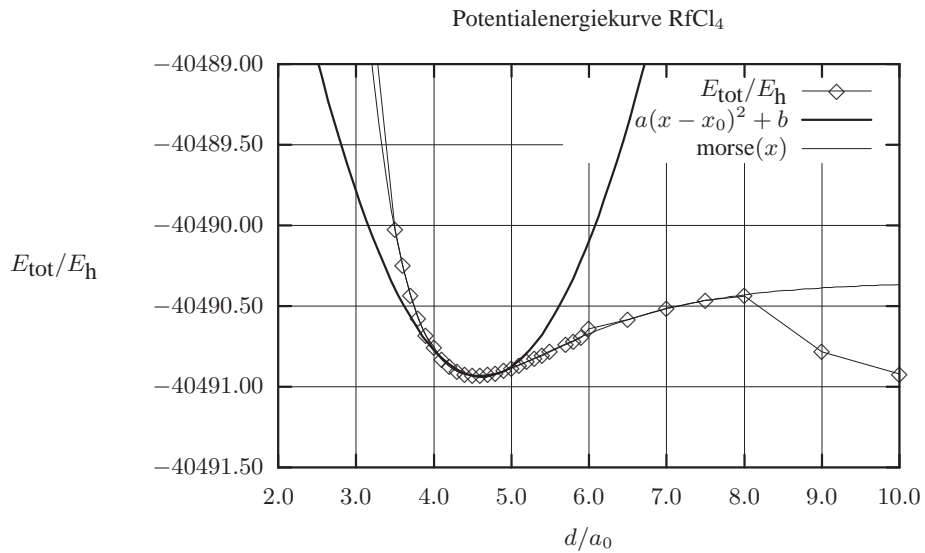


Abbildung 4.8.: Potentialkurven MCl₄ (RLDA). Neben den berechneten totalen Energien wurde jeweils eine Parabel im Energieminimum und eine Morse-Funktion angefitet.

Alle weiteren Parameter wurden genau wie in Abschnitt 4.2 beim Au₂-Molekül gesetzt. Für die Integration über $d\Omega$ wurde eine Genauigkeit ϵ_θ vorgegeben, die Zahl der Radialpunkte n_r wurde mit Gleichung (4.9) bestimmt, die auch für RfCl₄ gute Ergebnisse liefert.

Diskussion der Ergebnisse

Die Tabellen 4.5 und 4.6 sowie Abbildung 4.9 zeigen die Ergebnisse der Vergleichsrechnung für RfCl₄. Die Konvergenzrate des Baerends-Verfahrens scheint bei der Gesamtladung etwas besser zu sein, die Gesamtpunktzahl ist allerdings bei MULIN-Programm immer geringer bis zu einer Genauigkeit von 10^{-8} . Bei der totalen Energie ist die Konvergenz beider Verfahren etwa gleich, das Baerends-Schema benötigt allerdings immer einen Faktor 8 . . . 10 mehr Punkte, um die gleiche numerische Genauigkeit zu erreichen.

Auch in diesem System zeigt sich wieder, dass die Konvergenz der errechneten Größen beschränkt ist. Bei der Gesamtladung ist dies der Fall bei etwa 10^{-10} , bei der totalen Energie bei 10^{-9} . Dies wird im letzten Systems im Abschnitt 4.4 ausführlicher diskutiert.

MULIN				
ϵ_{Ω}	n_{rad}	n_{tot}	Q	$E_{\text{tot}}/E_{\text{h}}$
$1.0 \cdot 10^{-2}$	23	195	172.00 92	-40490. 30
$0.5 \cdot 10^{-2}$	29	277	172.0 10	-4049 1.5
$3.2 \cdot 10^{-3}$	33	383	171.99 69	-40490. 84
$1.0 \cdot 10^{-3}$	42	589	171.999 10	-40490. 88
$0.5 \cdot 10^{-3}$	48	771	172.000 90	-40490.9 36
$3.2 \cdot 10^{-4}$	52	919	171.9999 78	-40490.92 01
$1.0 \cdot 10^{-4}$	61	1285	171.999 89	-40490.9 18
$0.3 \cdot 10^{-4}$	71	1856	172.0000 16	-40490.92 59
$1.0 \cdot 10^{-5}$	80	2646	172.0000 16	-40490.92 94
$0.3 \cdot 10^{-5}$	90	3322	171.99999 84	-40490.930 04
$1.0 \cdot 10^{-6}$	100	4442	172.00000 19	-40490.930 20
$0.3 \cdot 10^{-6}$	109	8776	172.00000 11	-40490.9304 35
$1.0 \cdot 10^{-7}$	119	14125	172.000000 38	-40490.9304 33
$0.3 \cdot 10^{-7}$	128	22247	172.0000000 19	-40490.9304 14
$1.0 \cdot 10^{-8}$	138	33084	171.9999999 85	-40490.9304 24

Tabelle 4.5.: RfCl_4 : Gesamtladung in Abhängigkeit der Zahl der Stützstellen (MULIN)

Baerends			
ϵ_{B}	n_{tot}	Q	$E_{\text{tot}}/E_{\text{h}}$
$1.0 \cdot 10^{-3}$	1078	17 4.4	-40 880.
$1.0 \cdot 10^{-4}$	1933	172.0 39	-404 57.
$1.0 \cdot 10^{-5}$	2907	171.99 09	-404 88.
$1.0 \cdot 10^{-6}$	4442	172.000 76	-4049 1.4
$1.0 \cdot 10^{-7}$	6420	171.9999 37	-40490. 76
$1.0 \cdot 10^{-8}$	8753	172.00000 24	-40490.9 91
$1.0 \cdot 10^{-9}$	11724	171.999999 34	-40490.9 27
$1.0 \cdot 10^{-10}$	15716	172.000000 47	-40490.930 52
$1.0 \cdot 10^{-11}$	25934	172.000000 40	-40490.9304 97

Tabelle 4.6.: RfCl_4 : totale Energie in Abhängigkeit der Zahl der Stützstellen (Baerends)

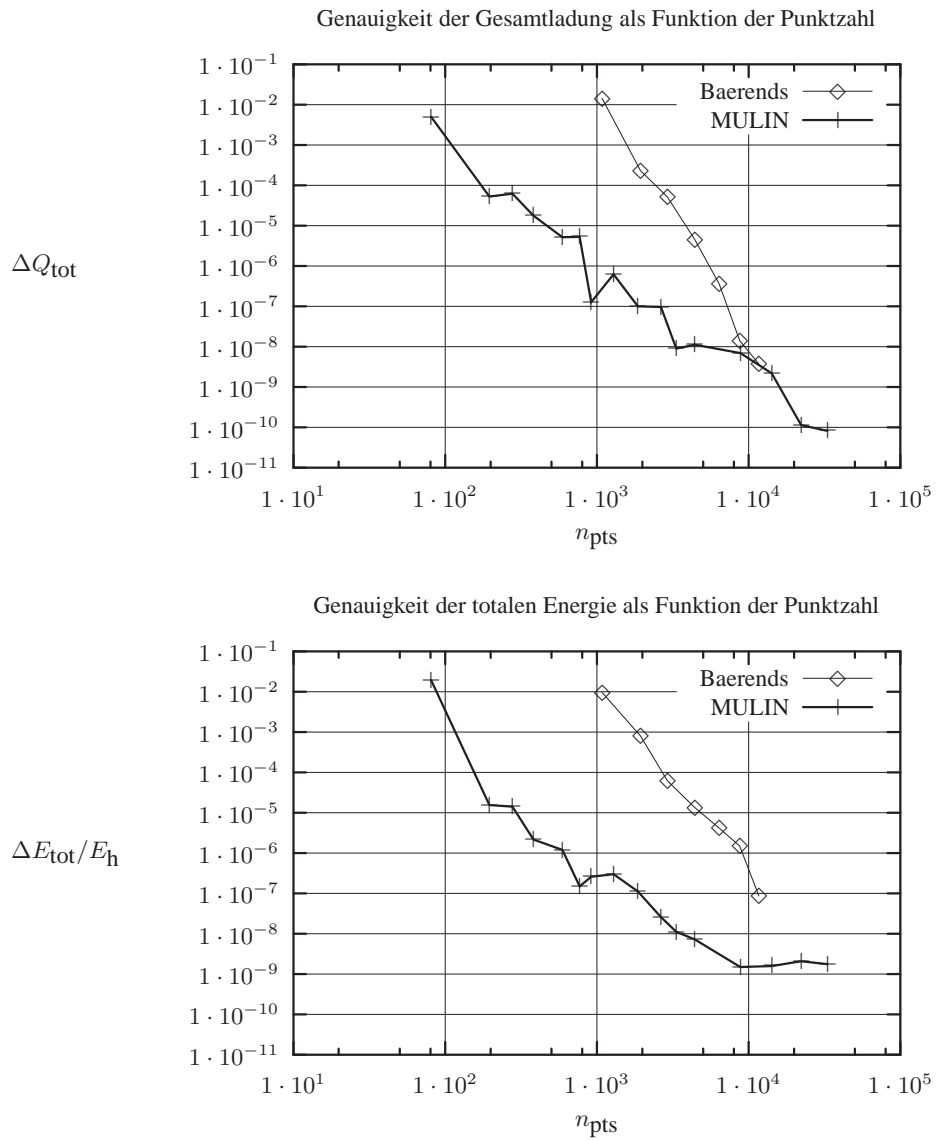


Abbildung 4.9.: RfCl_4 : Konvergenz von Gesamtladung und totaler Energie.

4.4. Cluster: Au@Au₉

Experimentell lässt sich Adsorption von Atomen an kristallinen Oberflächen mit hoher Genauigkeit untersuchen. Zu den gemessenen Größen gehören die Bindungsenergie der Ad-Atome, die Bindungsposition im Gleichgewichts-Zustand, das Potential der Bindung und das Schwingungsspektrum. Auf der theoretischen Seite sind Rechnungen zu diesen Systemen mit Schwierigkeiten verbunden. Sowohl durch die Oberfläche als auch durch das Ad-Atom wird die Translations-Symmetrie des Kristalls gebrochen, so dass Rechnungen mit einem periodischen Potential oder die Entwicklung der Wellenfunktion in ebene Wellen nicht mehr möglich ist. Bei Kristallen mit schweren Atomen kommt der erhöhte Rechenaufwand für die relativistische Berechnung und die Größe der Basissätze hinzu. Erst in jüngster Zeit sind daher realistische Berechnungen unter anderem in unserer Arbeitsgruppe [3] durchgeführt worden. Ziel des neuen Integrationsverfahrens soll daher auch sein, die langen Rechenzeiten für derartige Systeme zu vermindern.

Als Testsystem wurde ein Ausschnitt einer Au(100)-Oberfläche ausgewählt, auf der ein Au-Atom adsorbiert ist. Die Oberfläche wird durch neun Atome simuliert, von denen sich fünf Atome in der obersten Lage und vier Atome in der zweiten Lage befinden. Die Geometrie des Systems ist in Abbildung 4.10 dargestellt. Als Basissatz für Au ($Z=79$) wurden die Funktionen $1s_{1/2}$ bis $6s_{1/2}$ verwendet. Die Basis wurde ohne Ionisation berechnet.

Diskussion der Ergebnisse

Die Ergebnisse der Cluster-Rechnung sind in den Tabellen 4.4 und 4.8 sowie in graphischer Form in Abbildung 4.11 dargestellt.

Zum Vergleich der Genauigkeit wurden in den Graphiken jeweils zwei Geraden angefügt. Da die Werte doppelt-logarithmisch aufgetragen sind, gibt die Steigung m der Geraden die Konvergenzrate an, die sich in Einheiten von gesicherten Dezimalstellen ausdrücken lässt. Für die totale Energie hat der Exponent einen Wert von -6 , was bedeutet, dass sich die numerische Genauigkeit um $-m \log_{10}(2) = 1.8$ Dezimalstellen erhöht, wenn die Punktzahl verdoppelt wird, oder umgekehrt, dass die Punktzahl um einen Faktor $2^{1/1.8} = 1.46$ erhöht werden muss, um eine weitere Dezimalstelle zu erhalten. Bei der Gesamtladung hat der Exponent etwa den Wert 5 , was einen Gewinn von 1.5 Stellen bei einer Verdoppelung der Punktzahl bedeutet. Dies entspricht einer Erhöhung der Punktzahl um den Faktor $2^{1/1.5} = 1.58$, um die Genauigkeit um eine Stelle zu erhöhen.

Die Konvergenzraten der beiden verglichenen Integrationsverfahren sind etwa gleich, allerdings streuen die Werte des MULIN-Schemas stärker. Für dieses Schema wurde der Achsenabschnitt der Fitgeraden daher jeweils so gewählt, dass die errechnete Genauigkeit im interessanten Bereich immer höher ist, als die Fitgerade anzeigt. Ein direkter Vergleich der Achsenabschnitte auf der Abszisse zeigt, dass bei gleicher numerischer Genauigkeit unser Verfahren eine um den Faktor 4 geringere Punktzahl benötigt, um die Gesamtladung zu bestimmen. Bei der totalen Energie beträgt dieser Faktor sogar etwa 9 .

Aus den Achsenabschnitten der Ordinate lässt sich ablesen, dass bei gleicher Punktzahl ein um drei Stellen genauerer Wert der Gesamtladung berechnet wird, der Wert der totalen Energie ist um etwa sechs Stellen genauer.

Hier muss allerdings wieder auf die Grenzen der Konvergenz hingewiesen werden. Eine

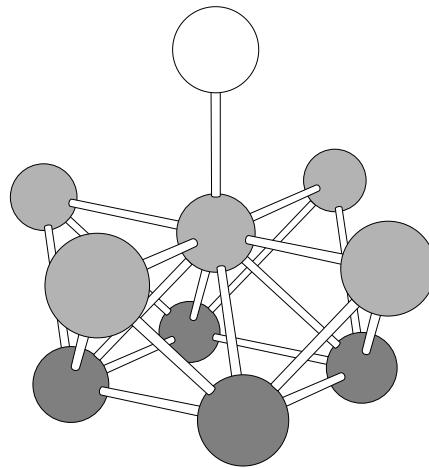
Genauigkeit besser als 10^{-10} bei der Gesamtladung von 10^{-7} bei der totalen Energie scheint nicht möglich zu sein. Trotz dass die Punktzahl erhöht wird, steigt die Genauigkeit nicht weiter an, oder verringert sich sogar wieder drastisch, wie im Fall der totalen Energie. Bei der totalen Energie zeigt der Vergleich mit den anderen Testsystemen, dass die erreichbare Genauigkeit mit zunehmender Größe des Moleküls geringer wird. Diese Beschränkung scheint vom Integrationsverfahren unabhängig zu sein und muss daher im Molekülstrukturprogramms vermutet werden. Dieser Verdacht wird auch dadurch gestützt, dass beide Integrationsverfahren ihre Genauigkeit an Testfunktionen messen, was schon bei der Aufstellung der Regel fehlschlagen würde, wenn die Vorgaben nicht erreicht werden würden.

MULIN				
ϵ_{Ω}	n_{rad}	n_{tot}	Q	$E_{\text{tot}}/E_{\text{h}}$
$1.0 \cdot 10^{-1}$	4	95	752	-1 75000
$0.6 \cdot 10^{-1}$	8	220	810	-2 54000
$0.4 \cdot 10^{-1}$	12	334	706.0	-1 20000
$2.5 \cdot 10^{-2}$	16	447	790.137	-19 3200
$1.3 \cdot 10^{-2}$	21	613	789.85	-190 180
$1.6 \cdot 10^{-2}$	20	626	790.21	-1899 75
$0.8 \cdot 10^{-2}$	25	788	789.915	-1899 59
$0.6 \cdot 10^{-2}$	27	871	789.89	-18998 5.6
$0.4 \cdot 10^{-2}$	31	1052	789.906	-18998 6.5
$2.5 \cdot 10^{-3}$	35	1288	789.944	-18998 7.8
$1.6 \cdot 10^{-3}$	39	1541	789.976	-189988.36
$0.6 \cdot 10^{-3}$	46	2079	790.0035	-189989.065
$2.5 \cdot 10^{-4}$	54	2713	789.9979	-189989.001
$1.6 \cdot 10^{-4}$	58	3158	789.99950	-189989.035
$1.0 \cdot 10^{-4}$	61	3583	790.0027	-189989.113
$0.4 \cdot 10^{-4}$	69	4796	790.00087	-189989.0809
$1.6 \cdot 10^{-5}$	77	6335	790.00020	-189989.0625
$1.0 \cdot 10^{-5}$	80	7104	790.000047	-189989.058
$0.4 \cdot 10^{-5}$	88	9529	790.000099	-189989.0604
$1.6 \cdot 10^{-6}$	96	12065	790.000042	-189989.0640
$1.0 \cdot 10^{-6}$	100	13897	790.000029	-189989.0774
$0.4 \cdot 10^{-6}$	107	19089	790.0000053	-189989.21
$1.0 \cdot 10^{-7}$	119	33810	790.00000121	-18999 4.4
$1.0 \cdot 10^{-8}$	138	88048	790.00000024	-190 240

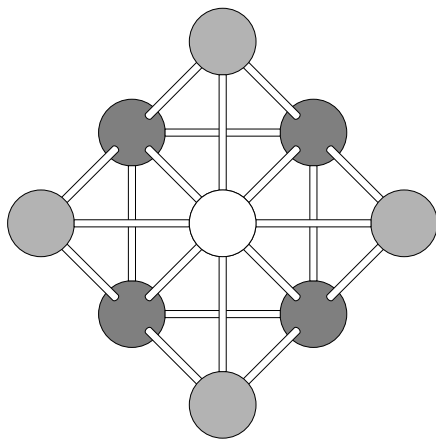
Tabelle 4.7.: Au an Au₉(100): Gesamtladung in Abhängigkeit der Zahl der Stützstellen (MULIN)

Baerends			
ϵ_{B}	n_{tot}	Q	$E_{\text{tot}}/E_{\text{h}}$
$3.0 \cdot 10^0$	4441	79 2.4	-2 17000
$4.0 \cdot 10^0$	6930	790.68	-189 240
$5.0 \cdot 10^0$	11055	789.921	-190 130
$6.0 \cdot 10^0$	16327	790.0095	-1899 75
$7.0 \cdot 10^0$	22323	789.99916	-1899 91
$8.0 \cdot 10^0$	30582	790.000110	-189989.44
$9.0 \cdot 10^0$	41135	790.00000049	-189989.115
$10.0 \cdot 10^0$	54219	789.99999989	-189988.97
$11.0 \cdot 10^0$	94910	789.99999986	-18999 9.2

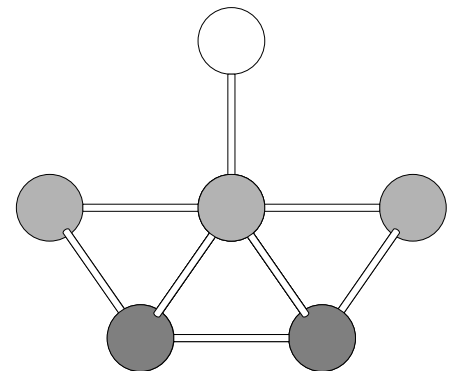
Tabelle 4.8.: Au an Au₉(100): totale Energie in Abhängigkeit der Zahl der Stützstellen (Baerends)



(a) Perspektivische Darstellung



(b) Projektion in XY-Ebene



(c) Projektion in XZ-Ebene

Abbildung 4.10.: Adsorption von Au an Au₉(100): Die oberste Lage ist durch hellgraue, die zweite Lage durch dunkelgraue Kugeln dargestellt. Das Ad-Atom sitzt über dem zentralen Atom der obersten Lage.

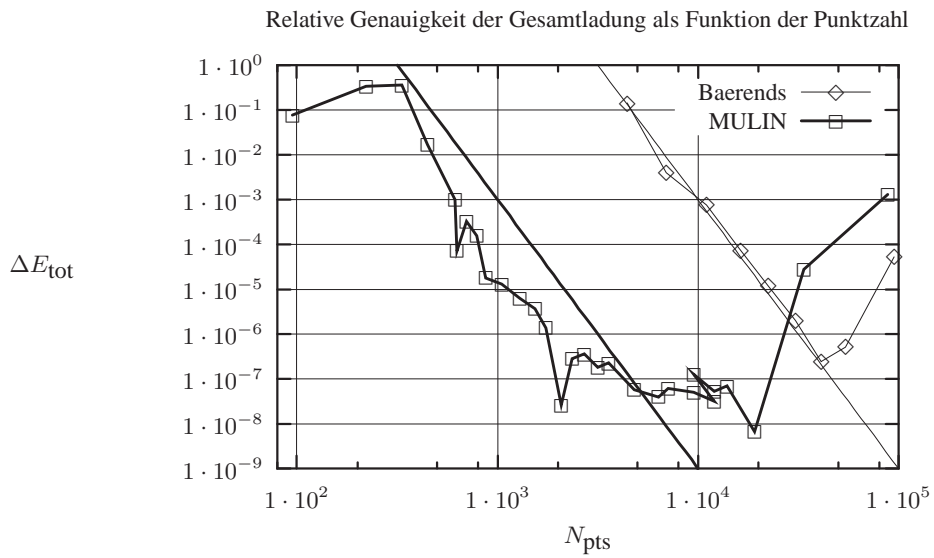
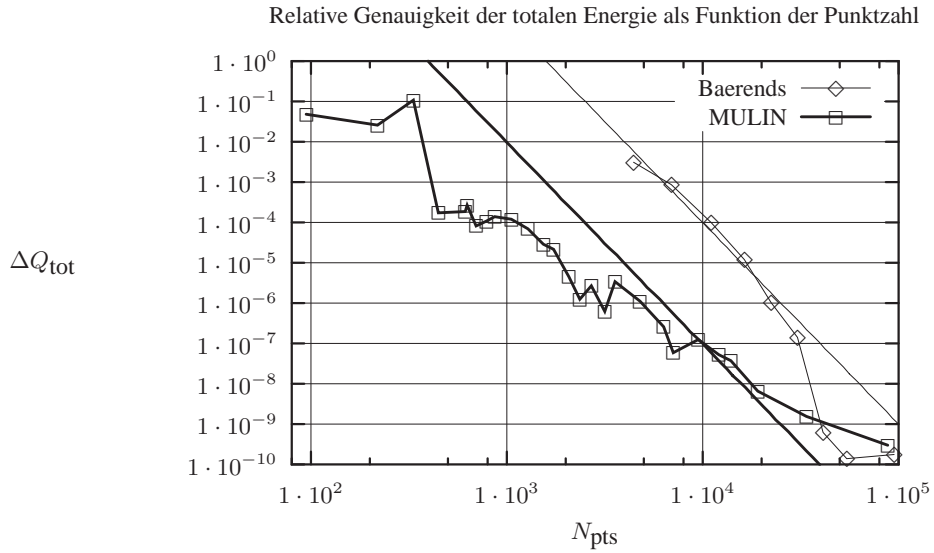


Abbildung 4.11.: Adsorption von Au an Au₉: Genauigkeit der Größen Gesamtladung (oben) und totale Energie (unten) in Abhängigkeit der Gesamtpunktzahl.

5. Zusammenfassung und Ausblick

Die vorliegende Arbeit entstand aus dem Bedürfnis, Rechnungen mit der Dichtefunktional-Methode für immer größere molekulare Systeme durchzuführen. Die Lösung der Dirac-Gleichung mittels der DFT bedingt die Auswertung der Matrixelemente des Hamilton-Operators in der Basis der Kohn-Sham-Orbitale. Der Einsatz eines numerischen Verfahrens dafür ist unumgänglich, nicht nur, weil in unserer Implementation eine numerische Basis von atomaren Orbitalen verwendet wird, sondern auch weil die Matrixelemente des Operators der Austausch-Korrelations-Wechselwirkung nicht-linear in den Basisfunktionen sind.

Die Rechenzeit des Molekülstruktur-Programms wird von der Integration der Matrixelemente dominiert und skaliert linear in der Anzahl der Stützstellen des numerischen Integrationsgitters. In dieser Arbeit wurde der Ansatz verfolgt, die Rechnungen durch Verringerung der Zahl der Integrationspunkte zu beschleunigen.

Das bisherige Verfahren zur Erstellung der numerischen Integrationsregel von Baerends et al. wurde ersetzt durch einen flexiblen Algorithmus, der auf dem Verfahren der Raumaufteilung in atomare Integrationszellen mittels einer Partitionsfunktion beruht. Durch geeignete Transformationen auf angepasste Koordinaten wird das Problem der dreidimensionalen Integration über den \mathbb{R}^3 auf geschachtelte Integrale niedrigerer Dimension zurückgeführt. Für die numerische Integration wurden ein- oder mehrdimensionale Integrationsregeln mit hoher Konvergenzrate verwendet. Die Zahl der Stützstellen der Integrationsregeln wurde mit der Methode der automatischen numerischen Integration so bestimmt, dass die erwarteten Integrale mit vorgegebener Genauigkeit berechnet wurden. Dazu wurden passende Testfunktionen benutzt, welche die Integranden des molekularen Problems repräsentieren. Schließlich wurde die Anzahl der Integrationspunkte reduziert, indem nur die unter der Punktgruppe des Moleküls symmetrie-inäquivalenten Stützstellen verwendet werden.

Die Testrechnungen für verschieden große Systeme zeigen, dass das neue Verfahren gegenüber dem bisherigen Integrationssystem mit einer deutlich verringerten Zahl der Integrationspunkte auskommt. Atomare Rechnungen können nun unter direkter Ausnutzung der Kugelsymmetrie schnell ausgeführt werden. Bei den Molekülen zeigt sich die Tendenz, dass das neue Verfahren im Vergleich zum Baerends'schen Algorithmus, mit zunehmender Größe der Moleküle deutlich effizienter wird. Die Testrechnungen zeigen weiterhin, dass eine bestimmte Kombination der Zellenfunktion und der Parameter der Transformationen für alle behandelten Moleküle gute Ergebnisse liefern.

Die unterschiedliche Anzahl der Stützstellen beider Integrationsverfahren ist auf die Art der Raumaufteilung zurückzuführen. Das Verfahren von Baerends teilt den Raum in sehr viele nicht-überlappende Raumgebiete auf und stellt eine eigene Integrationsregel für jeden dieser Bereiche auf. Wie viele dieser Bereiche für jedes Atom erstellt werden, hängt davon ab, wie viele

nächste Nachbarn im Sinne der Voronoi-Zerlegung des \mathbb{R}^3 jedes Atom besitzt, ist aber in jedem Fall größer als eins für Moleküle. Bei der Integration dieser Gebiete wird nicht vorausgesetzt, dass der Integrand stetig zwischen den getrennten Zellen verläuft. Die Integrationsregel benötigt dementsprechend eine höhere Gesamtpunktzahl.

Die Raumaufteilung in atomare Zellen hingegen benötigt lediglich eine einzelne Integrationsregel pro Atom, unabhängig davon wie viele Nachbarzellen ein Atom besitzt. Der Vorteil zeigt sich, wenn das System eine größere Anzahl von Atomen enthält. Da auch beim Barends'schen Verfahren die Anzahl der Zellen pro Atom bei großen Systemen konstant bleibt, wird sich die Zahl der Stützstellen in unserem Verfahren in diesem Limes um einen konstanten Faktor unterscheiden, wenn die Ergebnisse bei gleicher numerischer Genauigkeit verglichen werden. Im Falle des Gold-Clusters lag dieser Faktor bei der totalen Energie in etwa bei neun.

Es hat sich im Verlauf der Testrechnungen gezeigt, dass sich die numerische Genauigkeit der Ergebnisse nicht auf mehr als etwa 10^{-8} steigern lässt. Dieses Verhalten scheint unabhängig vom Integrationsverfahren zu sein und daher sind die numerischen Probleme im Molekülcode zu vermuten. Eine Verbesserung der Rechengenauigkeit des Molekülcodes, insbesondere der totalen Energie, ist wichtig, da Bindungs- und Ionisationsenergien immer nur aus Differenzen der totalen Energien bestimmt werden können. Für große Cluster steigt die totale Energie aber ungefähr linear mit der Zahl der (gleichartigen) Atome, während die Bindungsenergien etwa konstant bleiben. Dies begrenzt die Clustergröße, bei der Bindungsenergie-Rechnungen noch sinnvolle Werte liefern.

Das Problem der Skalierbarkeit der Dichtefunktionaltheorie hängt eng mit der Berechnung der Matrixelemente des Hamilton-Operators in der Basis der Kohn-Sham-Orbitale zusammen. Die Anzahl der zu berechnenden Matrixelemente steigt quadratisch mit der Zahl der elektronischen Freiheitsgrade des Systems, die wiederum linear mit der Zahl der beteiligten Atome anwächst. Zusätzlich vergrößert sich der Raumbereich, über den integriert werden muss, ebenfalls etwa linear mit der Anzahl der Atome. Insgesamt skaliert das Verfahren etwa mit der dritten Potenz der Zahl der beteiligten Elektronen.

Um dieses schlechte Skalenverhalten zu verbessern, müssen sowohl die Berechnung der Matrixelemente, als auch der Dichtefit verändert werden. Die in der Literatur bereits diskutierten neueren Verfahren übertragen die Raumaufteilung auf den Hamilton-Operator [20, 94] oder führen durchgehend Cut-Off-Radien für die Integration der Matrixelemente ein [46, 95]. Dadurch wird verhindert, dass die lokalisierten Basisfunktionen immer im gesamten \mathbb{R}^3 ausgewertet werden müssen. Der Radius der Integration der Zellenfunktionen muss in großen Clustern ebenfalls beschränkt werden. Dazu wählt man Zellenfunktionen, die nicht nur am Ort der Kerne benachbarter Zellen exakt verschwinden, sondern schon in einem bestimmten Radius um diese Zellen. Integriert man dann in einem dicht gepackten Festkörper, so stößt man in jeder Richtung sehr schnell auf diese Kugeln, so dass die Integration einer einzelnen Zelle nicht mehr bis $r \rightarrow \infty$ laufen muss. Geeignete Zellenfunktionen sind im Integrationsprogramm bereits implementiert (3.26), (3.27).

Die Raumaufteilung des Integrationsgebietes in atomare Zellen führt eine zusätzliche Winkelvariation der Integranden in den einzelnen Zellen ein. Dies macht sich dadurch bemerkbar, dass für die Integration über die Winkelvariablen eine höhere Integrationsordnung notwendig wird. Dieser Nachteil des Integrationsverfahrens, lässt sich durch zweidimensionale Koordinatentransformationen für die Winkelvariablen teilweise wieder ausgleichen. Solche Transforma-

tionen lassen sich durch Integration eines Funktions-Prototyps ermitteln, der die Integrationspunkte auf Winkelrichtungen fokussiert, die relevante Beiträge zum Gesamtintegral liefern. Ein entsprechender Algorithmus für die Berechnung einer numerischen Transformation ist schon im Programm für eindimensionale Integrationsgitter implementiert. Eine Erweiterung auf analytische Transformationen für zweidimensionale numerische Regeln könnte allerdings noch bessere Ergebnisse liefern.

Durch die in dieser Arbeit vorgestellten Methode der numerischen Integration lässt sich die Größe der berechenbaren Moleküle in etwa verdoppeln oder die Rechenzeit bei gleicher Größe um einen Faktor 4–8 verringert werden. In Verbindung mit der Methode der Cluster-Einbettung und der Frozen-Core-Näherung können Festkörper-Oberflächen nun realistisch modelliert werden. Um noch größere Systeme berechnen zu können, muss das Skalenverhalten des Molekülstrukturprogramms, wie oben beschrieben, verbessert und eine Parallelisierung des Programmcodes vorgenommen werden.

A. Physikalische Konstanten und Definitionen

Für alle Rechnungen wurden durchgehend atomare Einheiten benutzt ($\hbar = e = m = 4\pi\epsilon_0 = 1$). Die atomare Längeneinheit a_0 entspricht dem klassischen Bahnradius für $n = 1$ des Elektrons im Wasserstoffatom. Die atomare Energieeinheit E_h entspricht der doppelten Bindungsenergie des Grundzustandes.

In der Literatur werden Größen oft auch in anderen Einheiten angegeben. Für Energien ist die Einheit eV gebräuchlich, für Bindungslängen das Ångström und für Wellenzahlen die Einheit cm^{-1} . Tabelle A.1 stellt die verwendeten Umrechnungsfaktoren zusammen und listet auch abgeleitete Einheiten auf.

Physikalische Größe	Definition	atom. Einh.	S. I. und andere Einheiten
Plancksche Konstante	$\hbar = h/2\pi$	1	$1.054571596 \cdot 10^{-34}$ Js [96]
Ladung des Elektrons	e	1	$1.602176462 \cdot 10^{-19}$ C [96]
Ruhemasse des Elektrons	m_e	1	$9.10938188 \cdot 10^{-31}$ kg [96]
Permittivität des Vakuums	ϵ_0	$1/4\pi$	0.510998902 MeV/ c^2 $8.854187817 \cdot 10^{-12}$ F/m [96]
Atomare Energieeinheit (Hartree)	$1 E_h = \frac{m_e e^4}{(4\pi\epsilon_0 \hbar)^2}$	1	$4.35974383 \cdot 10^{-18}$ kg m ² /s ² 27.2113836 eV
Atomare Längeneinheit (Bohr)	$1 a_0 = \frac{\hbar}{e^2 m_e}$	1	$5.29177208 \cdot 10^{-11}$ m 0.529177208 Å
Atomare Zeiteinheit	$\frac{\hbar^3 (4\pi\epsilon_0)^2}{m_e e^4}$	1	$2.41888433 \cdot 10^{-17}$ s
Atomare Geschwindigkeitseinheit	$\frac{e^2}{\hbar(4\pi\epsilon_0)}$	1	$2.18769125 \cdot 10^6$ m/s
Lichtgeschwindigkeit im Vakuum	c	137.0359895	$2.99792458 \cdot 10^8$ m/s [96]
Vereinheitlichte Atom. Masse	$1 u = \frac{1}{12} m(^{12}\text{C})$	1822.88848	$1.66053873 \cdot 10^{-27}$ kg [96] $9.31494013 \cdot 10^8$ eV/ c^2
Masse des Au-Atoms	196.96655 u [96]	359048.06	$3.2707058 \cdot 10^{-25}$ kg

Abbildung A.1.: Verwendete Einheiten und Umrechnungsfaktoren. Sämtliche Referenzgrößen wurden den Tabellen der Particle Data Group [96] entnommen. Alle anderen Werte wurden daraus berechnet und auf neun signifikante Stellen gerundet.

B. Details zu den Rechnungen

B.0.1. Trapezregel für periodische Integranden

Satz B.1 (Trapezregel) Die Trapezregel mit n Punkten integriert periodische Funktionen exakt, falls deren Reihenentwicklung bei Harmonischen der Ordnung n abbricht.

Zum Beweis betrachten wir eine Funktion $f : \mathbb{R} \rightarrow \mathbb{C}$ mit der Eigenschaft $f(x) = f(x + 2\pi)$ und der Reihenentwicklung

$$f(x) = \sum_{k=0}^{\infty} a_k \exp(ikx).$$

Führt man die Integration aus, so erhält man den ersten Fourier-Koeffizienten:

$$\int_0^{2\pi} f(x) dx = 2\pi a_0$$

Eine n -Punkt Trapezregel für das Intervall $[0, 2\pi[$ hat an den Stützstellen $x_j = 2\pi j/n$ die Gewichte $w_j = 2\pi/n$ für $0 \leq j < n$. Die numerische Integration von $f(x)$ mit der Trapezregel liefert:

$$\tilde{I} = \sum_{j=0}^{n-1} w_j f(x_j) = \sum_{j=0}^{n-1} 2\pi/n \sum_{k=0}^{\infty} a_k \exp(2\pi i k j/n)$$

Nachdem die Summanden mit $k = 0$ und $k = n$ abgespalten wurden, kann die Summe über j ausgeführt werden, da es sich um eine geometrische Reihe handelt:

$$\tilde{I} = 2\pi a_0 + 2\pi a_n + \sum_{k=1, k \neq n}^{\infty} 2\pi a_k/n \frac{\exp(2\pi i k) - 1}{\exp(2\pi i k/n) - 1}$$

Für ganzzahlige k ist aber der Zähler in der Summe gleich Null, so dass gilt:

$$\tilde{I} = 2\pi a_0 + 2\pi a_n$$

Der numerische Fehler wird also durch den Fourier-Koeffizienten a_n der zu integrierenden Funktion bestimmt, falls eine n -Punkt Trapezregel verwendet wird. Dies bedeutet, dass die numerische Regel genau dann exakt ist, wenn alle Koeffizienten a_k mit $k \geq n$ verschwinden.

C. Das Integrationsprogramm

C.1. Beschreibung der Eingabe-Datei

Die vielen unterschiedlichen Parameter und Betriebsarten des Integrationsprogramms werden durch eine Eingabe-Datei eingestellt. Diese Datei stellt die Verbindung mit dem Molekülcode her, bestimmt, wie die Integrationsregel generiert wird und welche zusätzlichen Ausgabe-Dateien erzeugt werden.

Die Eingabe-Datei besteht aus einzelnen Zeilen, die jeweils eine einzelne Einstellung vornehmen oder eine Berechnung auslösen. Der Inhalt der Datei ist nicht case-sensitiv, d. h. Groß-/Kleinschreibung ist für die Schlüsselworte nicht von Bedeutung. Im Gegensatz dazu muss bei Dateinamen entsprechend der Konventionen des Betriebssystems auf Groß-/Kleinschreibung geachtet werden.

Die Eingabe-Datei wird zeilenweise von oben nach unten eingelesen, mit Hilfe des INCLUDE-Kommandos lässt sich jedoch auch in eine externe Datei verzweigen.

In der folgenden Beschreibung der Eingabe-Datei werden Befehle in Courier-Schrift, Schlüsselworte in *Schrägschrift*, "Dateinamen" zusammen mit den einzugebenden doppelten Anführungszeichen in *Schrägschrift* und numerische Werte oder Parameter im Mathematiksatz dargestellt. Optionale Parameter sind in eckige Klammern gesetzt.

C.1.1. Eingabe

INCLUDE "Dateiname"

Einlesen einer Include-Datei. Nach dem Einlesen der Include-Datei wird das ursprüngliche Input-File weiter abgearbeitet. Include-Dateien bieten die Möglichkeit, gemeinsame Parametersätze zusammenzufassen und vom Molekülprogramm während des Programmlaufes erzeugte Dateien einzulesen.

Drei spezielle Dateien werden vor dem Aufruf der Integrationsroutinen vom Molekülcode erzeugt und können durch den INCLUDE-Befehl eingelesen werden. Diese Dateien enthalten einen Befehl zum Einlesen der atomaren Basissätze (Datei "*base.inp*"), die Geometrie (Datei "*centers.inp*") sowie die Symmetrie (Datei "*symops.inp*") des Moleküls. Durch das Einbinden dieser Dateien kann der Integrationsinput unabhängig vom zu berechnenden Molekül gemacht werden, so dass ein einziger Input für alle Moleküle verwendet werden kann. Da das Einbinden dieser Dateien aber optional ist, lassen sich aber weiterhin Experimente mit eigenen Parametern für den Integrationscode durchführen.

READ BASE "Dateiname" n_b Byte-Order Version

Einlesen von atomaren Basissätzen. Von der Datei werden n_b Basissätze eingelesen. Die Basissatzdateien sind Binärdateien, die nicht plattformunabhängig sind. Damit auch Basissatzdateien gelesen werden können, die nicht auf dem gleichen Rechnersystem erzeugt wurden, muss der Parameter *Byte-Order* auf die entsprechende Prozessorarchitektur gesetzt werden (Tabelle C.1)

<i>Byte-Order</i>	Prozessorarchitektur
BIG POWER POWERPC	Big-Endian Architektur z. B. IBM Power oder PowerPC
LITTLE X86	Little-Endian Architektur z. B. Intel
HOST	Datei vom eigenen Rechnersystem

Tabelle C.1.: Erlaubte Werte für *Byte-Order*

Der Parameter *Version* trägt den Änderungen im Format der Basissatzdateien Rechnung. Da die Dateien keine eigene Versionsinformation enthalten, muss diese zusätzlich angegeben werden. Tabelle C.2 zeigt, welche Parameter in den Basissatzdateien enthalten sind und auf welche Werte die Parameter anderenfalls gesetzt werden. Dabei bedeutet der Eintrag „#“, dass die Variable aus der Basissatzdatei gelesen wird.

Innerhalb der Umgebung des Molekülstrukturprogramms ist die Angabe der Basissatz-Datei redundant, da sie schon im RELMOS-Input spezifiziert ist. RELMOS erzeugt daher vor dem Aufruf des Integrationsprogramms die Datei "*base.inp*", in der der READ BASE-Befehl mit den richtigen Parametern enthalten ist. Im Integrationsinput genügt daher der Befehl INCLUDE "*base.inp*", um die Basissatz-Datei einzulesen.

Version	NPRIN	CLL	DensityOpt	XCCase	IDIRAC	RMAX[ao]
V1	#	<i>c</i>	0	0.0	1	30
V2	0	<i>c</i>	#	0.0	#	30
V3	0	<i>c</i>	0	# (INT)	#	#
V4	0	<i>c</i>	0	# (DOUBLE)	#	#
V5	0	#	0	# (DOUBLE)	#	30

Tabelle C.2.: Erlaubte Werte für *Version*

C.1.2. Erzeugung der numerischen Integrationsregel

Der zentrale Befehl zur Erzeugung der Integrationsregel lautet

```
PW GENERATE
```

Hiermit werden alle bisher gesetzten Parameter benutzt, um die Punkte und Gewichte zu generieren. Diese werden noch nicht unmittelbar ausgegeben, sondern zunächst intern gespeichert. Die Ausgabe kann später mit dem in Abschnitt C.1.3 beschriebenen Befehl `PW DUMP ...` erfolgen.

C.1.3. Ausgabe

`PW DUMP "Dateiname" BINARY Byte-Order`

`PW DUMP "Dateiname" UNFORMATTED Byte-Order`

Ausgabe der Punkte und Gewichte in eine für das Molekülstrukturprogramm lesbare Binärdatei. Der Parameter *Byte-Order* (Tabelle C.1) bestimmt die Prozessorarchitektur, auf der das Molekülstrukturprogramm abläuft. Normalerweise laufen beide Programme auf der gleichen Plattform ab, so dass der Parameter auf `HOST` gesetzt werden kann.

`PW DUMP "Dateiname" ASCII`

Ausgabe der Punkte und Gewichte in eine ASCII-Datei. Jeweils eine Zeile enthält die durch Leerzeichen getrennten Koordinaten und Gewichte eines Integrationspunktes im Format:

$$x \quad y \quad z \quad w$$

Die Zahlen werden mit doppelter Genauigkeit in Exponentialschreibweise ausgegeben.

`PW COPY "Input" "Output" BINARY Byte-Order`

`PW COPY "Input" "Output" UNFORMATTED Byte-Order`

`PW COPY "Input" "Output" ASCII`

Kopieren und konvertieren einer Integrationsdatei. Mit diesem Befehl können vorberechnete Integrationsdateien aus der Datei *"Input"* gelesen und in ein anderes Binärformat für die Prozessorarchitektur *Byte-Order* oder in eine ASCII-Darstellung umgewandelt werden. Die konvertierte Datei wird nach *"Output"* geschrieben.

`PW STATISTICS "Dateiname"`

Ausgabe einer Datei mit statistischen Informationen über die generierte Integrationsregel. Die Datei enthält für jedes Integrationszentrum u. a. die Anzahl der Integrationspunkte vor der Symmetrie-Reduktion.

`PW INTEGRATE "Dateiname"`

Integration der Testfunktionen und Ausgabe in eine Datei. In die Ausgabe-Datei wird eine Kurzbeschreibung jeder Testfunktion und deren mit der numerischen Regel berechnetes Integral über den \mathbb{R}^3 ausgegeben.

POVRAY CENTER *"Dateiname"*

Ausgabe einer Beschreibungs-Datei für die dreidimensionale Darstellung der Integrationszentren. Die Datei kann als Eingabe für das Raytracing-Programm POV-Ray [97] verwendet werden.

POVRAY LEBEDEV *L "Dateiname"*

Ausgabe einer POV-Ray Beschreibungs-Datei für die dreidimensionale Darstellung der Lebedev-Integrationsregel der Ordnung *L* für die Oberfläche der Einheitskugel.

POVRAY LOBATTO2D *L "Dateiname"*

Ausgabe einer POV-Ray Beschreibungs-Datei für die dreidimensionale Darstellung der Produktregel der Ordnung *L* für die Oberfläche der Einheitskugel.

POVRAY IRULE *"Dateiname"*

Ausgabe einer POV-Ray Beschreibungs-Datei für die dreidimensionale Darstellung der erzeugten dreidimensionalen numerischen Integrationsregel..

C.1.4. Diagnose und Fehlersuche

Das Integrationsprogramm kann umfangreiche Informationen zur Diagnose und Fehlersuche liefern. Dabei lässt sich angeben, bis zu welcher Schachtelungstiefe der Unterprogramme Ausgaben erzeugt und welche Informationen ausgegeben werden sollen. Sämtliche Ausgaben enthalten eine vorangestellte Markierung (siehe Tabelle C.3), die den Typ der Ausgabe und deren Länge der Schachtelungstiefe des erzeugenden Unterprogramms entspricht. Danach folgt der Name des Unterprogramms und die eigentliche Meldung. Meldungen vom Typ ERROR lassen sich nicht unterdrücken und führen zum Abbruch des Programmablaufes.

SET OUTPUT_LEVEL *Output-Type* ON

SET OUTPUT_LEVEL *Output-Type* OFF

Schaltet die Ausgabe für einen bestimmten Ausgabebetyp ein oder aus. Der Parameter *Output-Type* ist in Tabelle C.3 aufgelistet. Voreingestellt ist, dass das Programm keine Ausgaben macht.

SET OUTPUT_DEPTH *depth*

Bestimmt, bis zu welcher Schachtelungstiefe der Unterprogramme Debugging-Ausgaben erfolgen. Voreinstellung: 0.

SET DEBUG_PARSER *pdebug*

Schaltet Debugging-Ausgaben des Eingabeparsers ein (*pdebug* ≠ 0) oder aus (*pdebug* = 0).

Art der Ausgabe	Markierung	<i>Output-Type</i>
Name der Funktion	+	NAME
Parameter beim Aufruf	>	PARAMETERS
Rückgabewert beim Beenden	<	RESULT
Rückkehr aus der Funktion	-	END
Information	I	INFO
Warnung	W	WARN
Fehler (führt zum Abbruch)	E	ERROR
Debug-Information	D	DEBUG
Umfangreiche Datensätze in separate Dateien		DUMP

Tabelle C.3.: Kennzeichnung der Fehlermeldungen und erlaubter Werte für den Parameter *Output-Type*

C.1.5. Interpolation der numerischen Radialfunktionen

Für die Auswertung von Testfunktionen und zur Konstruktion bestimmter Zellenfunktionen können die numerischen atomaren Basisfunktionen herangezogen werden. Da diese Funktionen nur an diskreten Stützstellen tabelliert sind, ist es notwendig, sie zu interpolieren. Die voreingestellte Interpolationsmethode ist die Lagrange-Interpolation [47] der Ordnung 8.

SET INTERPOLATION CUBIC

Schaltet die kubische Interpolation für die numerische atomare Basis ein. Werte zwischen den Stützstellen des logarithmischen Radialgitters werden durch Polynome dritter Ordnung interpoliert.

SET INTERPOLATION LAGRANGE [*n*]

Schaltet die Lagrange-Interpolation für die numerische atomare Basis ein. Die Ordnung der Lagrange-Interpolation wird durch den optionalen Parameter *n* angegeben. Der voreingestellte Wert von *n* ist 8.

C.1.6. Zellenfunktion

Die Wahl der Zellenfunktion bestimmt die Raumaufteilung des Integrationsgebietes. Einige Zellenfunktionen besitzen zusätzliche Parameter bzw. verwenden die numerischen Basisfunktionen für die Zerlegung.

SET CELL.TYPE VORONOI

Raumaufteilung in Voronoi-Zellen (Gleichung (3.19)).

SET CELL_TYPE BECKE3

Zellenfunktion von Becke [13, 71]. Die Abschneidefunktion wird dabei dreimal iteriert (Gleichung (3.13)).

SET CELL_TYPE BECKE4

Zellenfunktion von Becke [13, 71]. Die Abschneidefunktion wird dabei viermal iteriert (Gleichung (3.13)).

SET CELL_TYPE STRATMANN1 *a*

Zellenfunktion von Stratmann et al. [20]. Die Abschneidefunktion $z(\mu, a)$ wurde einmal iteriert (Gleichung 3.26) und der Parameter a muss im Intervall $(0, 1)$ liegen (Gleichung (3.26)).

SET CELL_TYPE STRATMANN3 *a*

Zellenfunktion von Stratmann et al. [20]. Die Abschneidefunktion $z(\mu, a)$ wurde dreimal iteriert (Gleichung 3.27) und der Parameter a muss im Intervall $(0, 1)$ liegen (Gleichung (3.26)).

SET CELL_TYPE DELLEY1

Zellenfunktionen von Delley [14]. Die Dichte $\rho_i(r)$ wird aus den atomaren Basissätzen berechnet (Gleichung (3.21)).

SET CELL_TYPE DELLEY3

Zellenfunktionen von Delley [14]. Die Dichte $\rho_i(r)$ wird aus den atomaren Basissätzen berechnet. Der Parameter r_0 hat dabei den Wert $0.5 a_0$ (Gleichung (3.23)).

SET ATOMIC_SIZE_ADJUSTMENT ATYP

Schalter für die Änderung der Zellgröße in Abhängigkeit der Kernladung (Tabelle C.4).

<i>ATYP</i>	Referenz
BECKE	[13], Gleichung (3.14)
TREUTLER	[18]
POPLE	[98]
MURRAY	[76]
NONE	keine Änderung der Zellgröße

Tabelle C.4.: Erlaubte Werte für *ATYP*

C.1.7. Symmetrie

Um die Punktzahl durch Ausnutzung der Molekülsymmetrie zu reduzieren, verwaltet das Programm intern eine Liste von Symmetrieoperatoren, welche die Orte der Kerne in sich selbst überführen und die Punktgruppe des Moleküls bilden. Die Operatoren wirken auf kartesische Vektoren des \mathbb{R}^3 und werden durch einen Rotations- und einen Translationsanteil dargestellt, der als Matrix bzw. Vektor angegeben wird.

SYMOP R_{11} R_{12} R_{13} R_{21} R_{22} R_{23} R_{31} R_{32} R_{33} t_1 t_2 t_3

Hinzufügen eines Symmetrieoperators, der aus der Rotationsmatrix \mathbf{R} und dem Translationsvektor \mathbf{t} besteht. Ein Vektor \mathbf{r} des \mathbb{R}^3 wird durch den Symmetrieoperator in den Vektor \mathbf{r}'

$$r'_i = \sum_j R_{ij} r_j + t_i$$

überführt. Die Symmetrieoperatoren müssen eine Gruppe bilden und der $\mathbb{1}$ -Operator muss explizit angegeben werden.

Wird das Integrationsprogramm als Unterprogramm des Molekülprogramms benutzt, so erzeugt der Molekülcode die Datei "symops.inp", welche die Symmetrieoperatoren enthält und in der Input-Datei mit dem Befehl

```
INCLUDE "symops.inp"
```

eingelassen werden kann.

C.1.8. Parameter der Integrationszentren

Jedes einzelne Integrationszentrum besitzt einen eigenen Satz von Parametern, die den Ort und Orientierung des Integrationszentrums, die Art der Integrationsregel für die Zelle, deren Symmetrieklasse und anderen Informationen beinhalten

Neben der Liste der Integrationszentren verwaltet das Integrationsprogramm einen Satz von Standardparametern, der zu jedem Zeitpunkt geändert werden kann und als Ausgangspunkt für die Parameter von neu spezifizierten Zellen dient. Dadurch ist es möglich, die Input-Datei übersichtlich zu halten und einheitliche Vorgaben für alle vom Molekülprogramm erzeugten Integrationszentren zu machen.

BEGIN CENTER

Spezifikation eines neuen Integrationszentrums. Bis zum Befehl **END CENTER** (s. u.) können nur die in diesem Abschnitt beschriebenen Befehle benutzt werden.

BEGIN CENTER DEFAULT

Spezifikation der Standardparameter für neue Integrationszentren. Durch diesen Befehl wird kein neues Integrationszentrum angelegt, die hier gesetzten Parameter dienen lediglich als Vorgabe für alle Parameter in nachfolgenden **BEGIN CENTER/END CENTER**-Blöcken und können dort überschrieben werden.

END CENTER

Ende der aktuellen Deklaration eines Integrationszentrums. Dieser Befehl beendet auch die Deklaration `BEGIN CENTER DEFAULT`.

ORIGIN n_x n_y n_z

Definition des Kernortes. Der Vektor \mathbf{n} bestimmt das Zentrum für lokale Transformation der Integrationsregel von Kugelkoordinaten auf kartesische Koordinaten, den Ursprung der Basisfunktionen die mit diesem Integrationszentrum assoziiert ist, den Ursprungsvektor der Zelle für die Zellenfunktion, sowie das Zentrum der Rotation des Koordinatensystems der Zelle, die mit dem `ROTATION`-Befehl festgelegt wird. In Verbindung mit dem Molekülprogramm wird der Kernort in der Datei `"centers.inp"` vor dem Aufruf des Integrationsprogramms abgelegt.

ROTATION r_{11} r_{12} r_{13} r_{21} r_{22} r_{23} r_{31} r_{32} r_{33}

Rotationsmatrix für das Koordinatensystem des Integrationszentrums. Durch Angabe der Rotationsmatrix kann das lokale Koordinatensystem so gedreht werden, dass eine höhere Symmetrie bezüglich der neuen Z-Achse erreicht wird. Dadurch lässt sich die Anzahl der Integrationspunkte reduzieren. Die gesamte Transformation von Kugelkoordinaten auf kartesische Koordinaten ist in Gleichung (3.30) dargestellt.

BASE i_b

Definition des Basissatzes. Die Zahl n_b gibt die Nummer des Basissatzes an, der mit dem Integrationszentrum verbunden ist. i_b liegt im Intervall $[1, n_b]$, wobei n_b die mit dem Befehl `READ BASE` eingelesene Zahl der Basissätze angibt. Die Nummer des Basissatzes wird benötigt, um Transformationsparameter der Radialtransformation festzulegen und um Testfunktionen auszuwerten, die die numerischen Radialfunktionen benötigen.

CLASS i_c

Definition der Symmetrieklasse des Integrationszentrums. Die Zahl i_c gibt die Nummer der Klasse an. Alle Integrationszentren mit der gleichen Nummer gehören zu einer Klasse. Die Einteilung in Symmetrieklassen ist der Ausgangspunkt der Reduktion der Zahl der Integrationspunkte aufgrund der Molekülsymmetrie.

COORD c_1 c_2 ...

Festlegung Reihenfolge der Koordinaten für die Integration in Kugelkoordinaten. Durch diesen Befehl wird festgelegt, welche Bedeutung die nummerierten intrinsischen Koordinaten haben, deren Parameter durch nachfolgende Befehle festgelegt werden. Erlaubte Schlüsselworte für c_i sind in Tabelle C.5 aufgelistet.

c_i	Bedeutung der intrinsischen Koordinate i
R	Radialkoordinate r
$THETA$	Kosinus des Winkels θ
PHI	Azimutalwinkel ϕ
$OMEGA$	Zweidimensionale Integrationsregel für das Doppelintegral $\iint d\Omega$

Tabelle C.5.: Erlaubte Werte für c_i

Das Schlüsselwort *OMEGA* erzeugt eine zweidimensionale kombinierte Integrationsregel für die Kugeloberfläche:

$$\iint f(r, \cos \theta, \phi) d\Omega \approx \sum_i f(r, \cos \theta_i, \phi_i) w_i$$

C.1.9. Transformation der intrinsischen Koordinaten auf Kugelkoordinaten

Die folgenden Befehle der Form `TRANS i ...` definieren die Transformation der intrinsischen Koordinaten u_i auf die Kugelkoordinaten x_i . Die Zuordnung der Koordinaten x_i zu den Kugelkoordinaten $(r, \cos \theta, \phi)$ muss zuvor durch den Befehl `COORD` festgelegt worden sein. Die intrinsischen Koordinaten u_i liegen immer im Intervall $[-1, 1]$, die Kugelkoordinaten $(r, \cos \theta, \phi)$ in den entsprechenden Intervallen $[0, \infty)$, $[-1, 1]$ und $[0, 2\pi]$.

TRANS i EXPONENTIAL r_{\min} r_{\max}

Exponentielle Transformation. Die intrinsische Koordinate u wird mit der Transformation

$$x = r_{\max} \exp\left(\alpha \frac{u-1}{2}\right)$$

$$\alpha = \log_e \frac{r_{\max}}{r_{\min}}$$

auf die Koordinate x abgebildet. Diese Transformation ist nur für die Radialkoordinate sinnvoll.

TRANS i EXPONENTIAL2 n r_{\max} h_1

Exponentielle Transformation. Die Parameter der Transformation entsprechen den Parametern des Atomstruktur-Programms von Ellis et al. Die Transformation `EXPONENTIAL2` entspricht der Transformation `EXPONENTIAL` mit dem Parameter

$$\alpha = \frac{n-1}{h_1},$$

wobei n der Anzahl der Stützstellen des Gitters der atomaren Basis entspricht und h_1 dem Verhältnis zweier aufeinanderfolgender Abszissen.

TRANS *i* ATOMEXP *i_b*

Exponentielle Transformation. Mit dieser Syntax lassen sich die Parameter der Transformation aus der Basissatz-Datei verwenden, indem mit i_b die Nummer des Basissatzes angegeben wird. Dabei wird vorausgesetzt, dass die Basis zuvor mit dem Befehl `READ BASE` eingelesen wurde.

TRANS *i* BECKE r_m

Transformation von Becke [13]. Diese Transformation für die Radialkoordinate bildet die intrinsische Koordinate u nach Gleichung (3.31) auf das Intervall $[0, \infty)$. Der Parameter r_m gibt dabei die eine Längenskala vor und wird üblicherweise in der Größenordnung von $1 a_0$ gewählt.

TRANS *i* UNIT

Einheitstransformation. Diese Transformation lässt die intrinsische Koordinate unverändert:

$$x = u$$

TRANS *i* KOLB $t_0 m_1 m_2 \beta_1 \beta_2$

Radialtransformation nach Gleichung (3.42). Die Parameter der Transformation werden mit dem Befehl direkt angegeben.

TRANS *i* KOLB2 $t_0 m_1 x_{mm} \beta_1 \beta_2 r_{mm} r_{\min} r_{\max}$

Radialtransformation nach Gleichung (3.42). Der Parameter m_2 der Transformation wird durch Gleichung (3.46) bestimmt. Dies entspricht der Vorgehensweise in [66].

TRANS *i* THETA

Transformation für die Winkel-Koordinate $\cos \theta$. Die intrinsische Koordinate u entspricht dem Kosinus des Winkels θ .

$$x = u$$

TRANS *i* THETA2

Transformation für die Winkel-Koordinate $\cos \theta$. Die intrinsische Koordinate u wird als $1 - \frac{2\theta}{\pi}$ interpretiert.

$$x = \cos \frac{(1 - u)\pi}{2}$$

TRANS *i* PHI

Transformation für die Winkel-Koordinaten ϕ . Die intrinsische Koordinate u wird als $\frac{\phi}{\pi} - 1$ interpretiert.

$$x = (u + 1)\pi$$

TRANS *i* **LEBEDEV**
TRANS *i* **LOBATTO2D**
TRANS *i* **AHLRICHS**

Transformation für eine zweidimensionale Integrationsregel für die Winkel-Koordinaten θ und ϕ . Diese Integrationsregeln liefern als intrinsische Koordinaten die Parameter $u^1 = \cos \theta$ und $u^2 = \phi$, so dass diese drei Befehle lediglich die intrinsischen Koordinaten übernehmen:

$$\begin{aligned}x^1 &= u^1 \\x^2 &= u^2\end{aligned}$$

C.1.10. Numerische Integrationsregeln in den intrinsischen Koordinaten

In den intrinsischen Koordinaten wird eine ein- oder zweidimensionale numerische Integrationsregel aufgestellt. Für eindimensionale Integrationsregel liegen die Abszissen immer im Intervall $u \in [-1, 1]$, wobei das Intervall offen, halboffen oder geschlossen ist. Zweidimensionale Integrationsregel liefern Paare $u^{(1,2)}$ von Abszissen in den Intervallen $u^{(1)} \in [-1, 1]$ und $u^{(2)} \in [0, 2\pi]$. Die Anzahl der Abszissen wird durch den Befehl **IRULE** festgelegt.

NRULE *i* **TRAPEZ_OPEN**

Erweiterte Trapezregel für das offene Intervall $(-1, 1)$. Die Abszissen und Gewichte haben die in Gleichung (3.58) beschriebenen Werte.

NRULE *i* **TRAPEZ_SEMI**

Erweiterte Trapezregel für das halboffene Intervall $(-1, 1)$. Die Abszissen und Gewichte haben die in Gleichung (3.59) beschriebenen Werte.

NRULE *i* **TRAPEZ_CLOSED**

Erweiterte Trapezregel für das halboffene Intervall $(-1, 1)$. Die Abszissen und Gewichte haben die in Gleichung (3.60) beschriebenen Werte.

NRULE *i* **SIMPSON**

Erweiterte Simpson-Regel für das geschlossene Intervall $[-1, 1]$ nach Gleichung (3.60). Die Simpson-Regel ist nur für ungerade n definiert.

NRULE *i* **CUBIC_CLOSED**

Integration durch Polynom-Interpolation dritter Ordnung auf äquidistanten Stützstellen nach Gleichung (3.62).

NRULE *i* CUBIC_OPEN

Integration durch Polynom-Interpolation dritter Ordnung auf äquidistanten Stützstellen nach Gleichung (3.63).

NRULE *i* GAUSS_LEGENDRE

Gauß-Legendre-Quadratur (Abschnitt 3.4.1.5).

NRULE *i* GAUSS_CHEBYSHEV

NRULE *i* GAUSS_CHEBYSHEV1

Gauß-Tschebyscheff-Quadratur erster Art (3.69).

NRULE *i* GAUSS_CHEBYSHEV2

Gauß-Tschebyscheff-Quadratur zweiter Art (3.72).

C.1.11. Anzahl der Stützstellen der numerischen Integrationsregel

Die Zahl der Stützstellen der numerischen Integrationsregeln kann entweder direkt vorgegeben werden oder das Programm versucht mittels Testfunktionen eine vorgegebene Genauigkeit zu erreichen. Die Deklaration der Testfunktionen wird in Abschnitt 3.4.3.3 behandelt.

IRULE *i* FIXED *n*

Vorgabe einer festen Anzahl von Stützstellen n für die Koordinate i . Bei bestimmtem Integrationsregel kann es Einschränkungen bezüglich der Anzahl der Punkte geben. Diese Einschränkungen werden vom Programm abgefangen.

IRULE *i* ADAPTIVE $n_0 \epsilon$

Vorgabe einer Mindestanzahl n_0 von Stützstellen und einer Genauigkeit ϵ . Das Programm integriert die Testfunktionen und erhöht iterativ die Anzahl der Integrationspunkte solange, bis alle Testfunktionen mit der Genauigkeit ϵ integriert werden können.

Die Anzahl der Punkte bei eindimensionalen Integrationsregeln wird in jedem Iterationsschritt um den Faktor f erhöht, der mit dem Befehl `SET_ADAPTIVE_INCFAC` f eingestellt wird. Dabei wird berücksichtigt, dass bei bestimmten Integrationsregeln nicht alle Punktzahlen definiert sind. In diesem Fall wird auf die nächst höhere erlaubte Punktzahl zugegriffen. Um zu verhindern, dass durch die Rundung auf ganze Zahlen die Zahl der Integrationspunkte auf den vorhergehenden Wert abgerundet wird, wird die Zahl n der Punkte in der i -ten Iteration auf

$$n_i = n_0 f^i$$

gesetzt. Ergibt sich für n_i die gleiche Zahl n_{i-1} der Integrationspunkte, so wird n_i auf $n_{i-1} + 1$ gesetzt. Bei der Simpson-Regel wird zudem auf die nächsthöhere (ungerade) Zahl aufgerundet falls n_i gerade ist.

Bei zweidimensionalen Integrationsregel sind die Punkte und Gewichte teilweise tabelliert. Es sind also nur vorgegebene Werte für n_i erlaubt. Das Programm geht dann so vor, dass in jedem Iterationsschritt der nächste tabellierte Wert verwendet wird.

C.1.12. Variation der Zellengröße bei heteronuklearen Systemen

Ursprünglich von Becke [13] wurde vorgeschlagen, die Größe der Integrationszellen eines Systems vom Typ des Atoms abhängig zu machen, welches im Zentrum der Zelle liegt. Dazu wird für jede Zelle eine Längenskala definiert, welche z. B. dem Erwartungswert $\langle r \rangle$ der radialen Elektronendichte entspricht. Einige Zellenfunktionen verwenden diesen Parameter, um die Zellengröße zu modifizieren.

SIZE BRAGG_SLATER Z

Definition einer Längenskala r_m für die Ausdehnung der Integrationszelle. Der Wert von r_m wird in der Tabelle der Bragg-Slater-Radien [73] für das angegebene Element der Kernladungszahl Z nachgeschlagen. Der Radius r_m wird in einigen Zellenfunktionen verwendet, um die Zellgröße in heteronuklearen Systemen anzupassen. Eine solche Korrektur findet nur statt, falls mit dem Befehl `SET ATOMIC_SIZE_ADJUSTMENT ATYP` der Parameter *ATYP* auf einen Wert verschieden von NONE gesetzt wurde.

Das Programm meldet einen Fehler, falls für das Element Z kein Eintrag in der Tabelle vorhanden ist. In diesem Fall kann der Wert von r_m mit dem Befehl `SIZE AU` direkt vorgegeben werden.

SIZE AU r_m

Definition einer Längenskala r_m für die Ausdehnung der Integrationszelle. Der Wert des Parameters r_m wird in atomaren Längeneinheiten angegeben. Dieser Befehl kann alternativ zu `SIZE BRAGG_SLATER Z` verwendet werden und hat die gleichen Auswirkungen.

C.1.13. Testfunktionen

Die Anpassung der Integrationsregel mit Hilfe der automatischen numerischen Integration 3.4.3 erfordert die Vorgabe von Testfunktionen für die Abschätzung der numerischen Genauigkeit. Die im Programm implementierten Funktionen sind in Abschnitt 3.4.3.3 vorgestellt. In der Eingabe-Datei werden die Testfunktionen innerhalb eines `BEGIN TEST/END TEST`-Blockes definiert. Jeweils eine einzelne Testfunktion oder eine Gruppe von Testfunktionen wird durch eine Zeile in der Eingabe-Datei wie folgt beschrieben:

CENTER A ATOM i

Atomare Elektronendichte bezüglich des A -ten Kerns gebildet aus dem i -ten Basissatz (3.83). Es werden die atomaren Besetzungszahlen dieser Basis verwendet, um die Dichte zu bilden.

DENSITY

Summe der kugelsymmetrischen, atomaren Dichten (3.84).

OVERLAP ($A, i_A, (nl2j)_A, (m_j)_A, B, i_B, (nl2j)_B, (m_j)_B$)

Überlapp-Testfunktion zweier atomarer Basisfunktionen (3.85). Die Kombinationen der Quantenzahlen $nl2j$ werden als Blöcke mit jeweils drei Ziffern angegeben. Die dritte Ziffer entspricht der mit zwei multiplizierten halbzahligen Drehimpulsquantenzahl j .

SLATER

Approximation von allen Basissatzfunktionen durch Slater-Funktionen für alle Atome des Moleküls (3.87).

GAUSS (A, r_m)

Die Gauß-Funktion mit dem Radius r_m , zentriert am Ort des A -ten Kerns (3.90).

UNIT

Die konstante Funktion $f(\mathbf{r}) = 1$.

EXP01 (A)

Die Exponentialfunktion $f(\mathbf{r}) = \exp(-|\mathbf{r} - \mathbf{r}_A^0|)$, zentriert am Ort des A -ten Kerns (3.92).

C.2. Eingabe-Datei für atomare Rechnungen

Bei atomaren Rechnungen muss lediglich eine Integrationsregel für die Radialkoordinate erzeugt werden. Dies wird gewährleistet, indem im Input die Zahl der Integrationspunkte für die Winkel-Integration jeweils auf eins gesetzt wird. Die Zahl der Radialpunkte bestimmt die Integrationsgenauigkeit. Da bei atomaren Rechnungen die Rechenzeit kurz ist, kann immer eine große Zahl von Radialpunkte verwendet werden.

```
SET INTERPOLATION LAGRANGE 8
SET ITRANS NONE
SET ATOMIC_SIZE_ADJUSTMENT NONE
SET ADAPTIVE_INCFACOR 1.1

INCLUDE "base.inp"

BEGIN CENTER DEFAULT
COORD R THETA PHI
TRANS 0 KOLB2 1.0 2.0 4.0 2.0 0.8 20.0 0 55.0
TRANS 1 THETA
TRANS 2 PHI
IRULE 0 FIXED 250
IRULE 1 FIXED 1
IRULE 2 FIXED 1
NRULE 0 GAUSS_LEGENDRE
NRULE 1 TRAPEZ_SEMI
NRULE 2 TRAPEZ_SEMI
END CENTER

INCLUDE "symops.inp"
INCLUDE "centers.inp"

BEGIN TEST
  DENSITY
  SLATER
END TEST

PW GENERATE
PW DUMP "points.ext" BINARY HOST
```

C.3. Eingabe-Datei für lineare Moleküle

Bei linearen Molekülen orientiert RELMOS das Molekül so, dass alle Atome auf der x -Achse liegen. Da das Integrationsprogramm jedoch die z -Achse als ausgezeichnete Achse für die Winkel-Integration verwendet, wird mit dem Schlüsselwort ROTATION eine (eigentliche) Drehung der x -Achse auf die z -Achse ausgeführt. Danach wird für die Integration über die Winkel-Koordinate ϕ nur noch ein Integrationspunkt benötigt, da die Integranden rotations-symmetrisch bezüglich der Molekülachse sind.

```
SET INTERPOLATION LAGRANGE 8
SET ITRANS NONE
SET ATOMIC_SIZE_ADJUSTMENT NONE
SET ADAPTIVE_INCFACOR 1.1
SET CELL_TYPE DELLEY3

INCLUDE "base.inp"

BEGIN CENTER DEFAULT
ROTATION 0 0 1 0 -1 0 1 0 0
COORD R THETA PHI
TRANS 0 KOLB2 1.0 2.0 4.0 2.0 0.8 20.0 0 55.0
TRANS 1 THETA
TRANS 2 PHI
IRULE 0 FIXED 61
IRULE 1 ADAPTIVE 1 0.0001
IRULE 2 FIXED 1
NRULE 0 GAUSS_CHEBYSHEV
NRULE 1 GAUSS_LEGENDRE
NRULE 2 TRAPEZ_SEMI
END CENTER

INCLUDE "symops.inp"
INCLUDE "centers.inp"

BEGIN TEST
  DENSITY
  SLATER
END TEST

PW GENERATE
PW DUMP "points.ext" BINARY HOST
```

C.4. Eingabe-Datei für dreidimensionale Moleküle

```
SET INTERPOLATION LAGRANGE 8
SET ITRANS NONE
SET ATOMIC_SIZE_ADJUSTMENT NONE
SET ADAPTIVE_INCFACOR 1.1
SET CELL_TYPE DELLEY3

INCLUDE "base.inp"

BEGIN CENTER DEFAULT
COORD R OMEGA
TRANS 0 KOLB2 1.0 2.0 4.0 2.0 0.8 20.0 0 55.0
TRANS 1 LOBATTO2D
IRULE 0 FIXED 70
IRULE 1 ADAPTIVE 6 1e-6
NRULE 0 GAUSS_CHEBYSHEV
NRULE 1 AHLRICHS
END CENTER

INCLUDE "symops.inp"
INCLUDE "centers.inp"

BEGIN TEST
  DENSITY
  SLATER
END TEST

PW GENERATE
PW DUMP "points.ext" BINARY HOST
```


Literaturverzeichnis

- [1] E. ENGEL, S. KELLER, A. FACCO BENETTI, H. MÜLLER und R. M. DREIZLER: *Local and nonlocal relativistic exchange-correlation energy functionals: Comparison to relativistic optimized-potential-model results*. Phys. Rev. A, 52:2750–2764, 1995.
- [2] E. ENGEL, S. KELLER und R. M. DREIZLER: *Generalized gradient approximation for the relativistic exchange-only energy functional*. Phys. Rev. A, 53:1367–1374, 1996.
- [3] D. GESCHKE: *Adsorption von CO an Plation - Eine theoretische Untersuchung mit einer Cluster-Näherung im Rahmen einer relativistischen Dichtefunktionaltheorie*. Dissertation, Universität Kassel, November 1999.
- [4] C. C. J. ROOTHAAN: *New Developments in Molecular Orbital Theory*. Rev. Mod. Phys., 23(2):69–89, April 1951.
- [5] W. J. HEHRE, L. RADOM, P. V. R. SCHLEYER und J. A. POPL: *Ab Initio Molecular Orbital Theory*. John Wiley & Sons, 1986.
- [6] W.-D. SEPP und B. FRICKE: *Implicit Projection Operators in Basis-Set Expansions of the Molecular Dirac-Fock-Slater Problem*. In: H. P. KELLY und Y.-K. KIM (Herausgeber): *Atomic Theory Workshop on Relativistic and QED Effects in Heavy Atoms*, Band 136 der Reihe *AIP Conference Proceedings*, Seiten 20–25, New York, 1985. National Bureau of Standards, Gaithersburg, MD, American Institute of Physics.
- [7] T. BAŞTÜĞ: *Genaue Berechnung der totalen Energien für kleine Moleküle und Atomcluster mit der Dirac-Fock-Slater Methode*. Dissertation, Universität Kassel, January 1994.
- [8] T. JACOB: *Einbettungsverfahren zur Simulation von Festkörperoberflächen*. Diplomarbeit, Universität Kassel, 1999.
- [9] P. M. BOERRIGTER, G. TE VELDE und E. J. BAERENDS: *Three-dimensional Numerical Integration for Electronic Structure Calculations*. Int. J. Quant. Chem., 33:87, 1988.
- [10] AMSTERDAM THEORETICAL CHEMISTRY, VRIJE UNIVERSITEIT: *Amsterdam density functional, ADF release 2.1*.
- [11] H. CONROY: *Molecular Schrödinger Equation. VIII. A New Method for the Evaluation of Multidimensional Integrals*. J. Chem. Phys., 47(12):5307–5318, December 1967.
- [12] D. E. ELLIS. Int. J. Quant. Chem., 25:35, 1968.

- [13] A. D. BECKE: *A multicenter numerical integration scheme for polyatomic molecules.* J. Chem. Phys., 88(4):2547, February 1988.
- [14] B. DELLEY: *An all-electron numerical method for solving the local density functional for polyatomic molecules.* J. Chem. Phys., 92:508–517, 1990.
- [15] W. YANG: *A local projection method for the linear combination of atomic orbital implementation of density-function theory.* J. Chem. Phys., 94(2):1208–1214, January 1991.
- [16] J. M. PÉREZ-JORDÁ, EMILIO SAN-FABIÁN und FEDERICO MOSCARDÓ: *A simple, reliable and efficient scheme for automatic numerical integration.* Comp. Phys. Comm., 70:271–284, 1992.
- [17] J. M. PÉREZ-JORDÁ: *Automatic numerical integration techniques for polyatomic molecules. Backward trimming.* J. Chem. Phys., 101(2):738–1740, July 1994.
- [18] O. TREUTLER und R. AHLRICHS: *Efficient molecular numerical integration schemes.* J. Chem. Phys., 102(1):346–354, January 1995.
- [19] M. E. MURA und P. J. KNOWLES: *Improved radial grids for quadrature in molecular density-functional calculations.* J. Chem. Phys., 104(24):9848–9858, June 1995.
- [20] M. J. FRISCH R. ERIC STRATMANN, G. E. SCUSERIA: *Achieving linear scaling in exchange-correlation density functional quadratures.* Chem. Phys. Lett., 257:213–223, July 1996.
- [21] T. ZHU, W. PAN und W. YANG: *Structure of solid-state systems from embedded-cluster calculations: A divide-and-conquer approach.* Phys. Rev. B, 53(19):12713–12724, May 1996.
- [22] V. TERMATH und J. SAUER: *Optimized molecular integration schemes for density function theory ab initio molecular dynamics simulations.* Chem. Phys. Lett., 255:187–194, June 1996.
- [23] K. YAMAMOTO, H. ISHIKAWA, K. FUJIMA und M. IWASAWA: *An accurate single-center three-dimensional numerical integration and its application to atomic structure calculation.* J. Chem. Phys., 106(21):8769–8777, June 1997.
- [24] H. ISHIKAWA, K. YAMAMOTO, K. FUJIMA und M. IWASAWA: *An Accurate Numerical Multicenter Integration for Molecular Orbital Theory.* Int. J. Quant. Chem., 72:509–523, 1999.
- [25] P. A. M. DIRAC: *Proc. Roy. Sec. In: The Quantum Theory of the Electron*, Band 117, Seite 610, 1928.
- [26] J. D. BJORKEN und S. D. DRELL: *Relativistische Quantenmechanik.* BI-Wiss.-Verlag, 1990.

- [27] P. A. M. DIRAC: *Proc. Roy. Soc. In: A Theory of Electrons and Protons*, Band 126, Seiten 360–365, 1930.
- [28] J. SUCHER: *Reduction of the Dirac Equation*. *Phys. Rev.*, 103:468, 1956.
- [29] J. SUCHER: *Relativistic Invariance and the Square Root Klein-Gordon-Equation*. *J. Math. Phys.*, 4:17, 1963.
- [30] G. E. BROWN und D. G. RAVENHALL: *On the interaction of two electrons*. *Proc. Roy. Soc. London A*, September 1951.
- [31] M. BORN und R. OPPENHEIMER: *Zur Quantentheorie der Molekeln*. *Ann. Phys.*, 84:457, 1927.
- [32] S. VARGA: *Relativistische Molekülstruktur – Implementierung der Geometrieoptimierung und neuer Dichtefunktionale*. Dissertation, Universität Kassel, 1998.
- [33] P. HOHENBERG und W. KOHN: *Inhomogeneous Electron Gas*. *Phys. Rev.*, 136(3B):864, 1964.
- [34] M. LEVY. In: *Proc. Natl. Acad. Sci. USA*, Band 76, Seite 6062, 1979.
- [35] A. K. RAJAGOPAL und J. CALLAWAY. *Phys. Rev. B*, 10:1, 1973.
- [36] W. KOHN und L. J. SHAM: *Self-Consistent Equations Including Exchange and Correlation Effects*. *Phys. Rev.*, 140(4A):1133, 1965.
- [37] J. C. SLATER: *A simplification of the Hartree-Fock method*. *Phys. Rev.*, 81:385–390, 1951.
- [38] R. GÁSPÁR. *Acta Physica Acad. Sci. Hung.*, 3:263, 1954.
- [39] J. C. SLATER: *Statistical exchange-correlation in the self-consistent field*. *Adv. Quant. Chem.*, 6:1–92, 1972.
- [40] J. W. D. CONOLLY: *The $X\alpha$ Method*. In: GERALD A. SEGAL (Herausgeber): *Semiempirical Methods of Electronic Structure Calculation, Part A: Techniques*, Band 7 der Reihe *Modern Theoretical Chemistry*, Kapitel 4, Seiten 105–132. Plenum Press, New York and London, 1977.
- [41] R. M. DREIZLER und E. K. GROSS: *Density Functional Theory*. Springer-Verlag, 1990.
- [42] E. ENGEL und R. M. DREIZLER: *Relativistic density functional theory*. *Top. Curr. Chem.*, 181:1–80, 1996.
- [43] D. E. ELLIS, H. ADACHI und F. W. AVERILL. *Surf. Sci.*, 58:497, 1976.
- [44] A. ROSÉN, D. E. ELLIS, H. ADACHI und F. W. AVERILL. *Chem. Phys.*, 65:3629, 1976.
- [45] B. DELLEY und D. E. ELLIS. *J. Chem. Phys.*, 76(4):1949, 1982.
- [46] G. TE VELDE und E. J. BAERENDS. *Phys. Rev. B*, 44:7888, 1991.

- [47] W. H. PRESS, B. P. FLANNERY, S. A. TEUKOLSKY und W. T. VETTERLING: *Numerical Recipes in C*. CAMBRIDGE, 1988.
- [48] IMSL, INC.: *IMSL Library. FORTRAN subroutines for Mathematics and Statistics*, 9.2 Auflage, November 1984.
- [49] P. O. LÖWDIN: *Quantum Theory of Many-Particle Systems. I. Physical Interpretations by Means of Density Matrices, Natural Spin-Orbitals, and Convergence Problems in the Method of Configurational Interaction*. Phys. Rev., 97:1474–1489, 1955.
- [50] P. O. LÖWDIN: *Quantum Theory of Many-Particle Systems. II. Study of the Ordinary Hartree-Fock Approximation*. Phys. Rev., 97:1490–1508, 1955.
- [51] P. O. LÖWDIN: *Quantum Theory of Many-Particle Systems. III. Extension of the Hartree-Fock Scheme to Include Degenerate Systems and Correlation Effects*. Phys. Rev., 97:1509–1520, 1955.
- [52] K. H. JOHNSON. J. Chem. Phys., 45:3085, 1966.
- [53] D. E. ELLIS und G. S. PAINTER: *Discrete Variational Method for the Energy-Band Problem with General Crystal Potentials*. Phys. Rev. B, 2(8):2887–2898, October 1970.
- [54] D. E. ELLIS und G. S. PAINTER. In: P. M. MARCUS, J. F. JANAK und A. R. WILLIAMS (Herausgeber): *Computational Methods in Band Theory*, Seite 271. Plenum Press, New York, 1971.
- [55] E. J. BAERENDS, D. E. ELLIS und P. ROS: *Self-consistent molecular Hartree-Fock-Slater calculations. I. The computational procedure*. Chem. Phys., 2:41, 1973.
- [56] A. ROSÉN: *Relativistic symmetry orbitals for the double groups C_{2v} , $C_{\infty v}$, $D_{\infty h}$ and O_h* . Int. J. Quant. Chem., 13:509, 1978.
- [57] J. P. ELLIOTT und P. G. DAWBER: *Symmetry in Physics*, Band 1. The Macmillan Press Ltd., London and Basingstoke, 1979.
- [58] J. B. DIAMOND: *Symmetry and the Scattered-Wave Model*. Chem. Phys. Lett., 20(1):63–66, January 1973.
- [59] J. MEYER, W.-D. SEPP und B. FRICKE: *Computation of relativistic symmetry orbitals for finite double point groups*. Comp. Phys. Comm., 54:55–73, 1989.
- [60] J. MEYER, W.-D. SEPP, B. FRICKE und A. ROSÉN: *A new version of the program TSYM generating relativistic molecular symmetry orbitals for finite double point groups*. Comp. Phys. Comm., 96:263–287, 1996.
- [61] H. SAMBE und R. H. FELTON. J. Chem. Phys., 62:1122, 1975.
- [62] B. I. DUNLAP, J. W. CONOLLY und J. R. SABIN. J. Chem. Phys., 71:3396, 1979.

- [63] B. I. DUNLAP, J. W. CONOLLY und J. R. SABIN. *J. Chem. Phys.*, 71(12):4993, 1979.
- [64] J. W. MINTMIRE und B. I. DUNLAP. *Phys. Rev. A*, 25(1):88, 1982.
- [65] J. D. JACKSON: *Klassische Elektrodynamik*. de Gruyter, Berlin and New York, 2 Auflage, 1982.
- [66] T. BAŞTÜĞ, W.-D. SEPP, D. KOLB, B. FRICKE, E. J. BAERENDS und G. TE VELDE: *All-electron Dirac-Fock-Slater SCF calculations for electronic and geometric structures of the Hg₂ and Hg₃ molecules*. *J. Phys. B*, 28:2325–2331, 1995.
- [67] P. MOON und D. E. SPENCER: *Field Theory Handbook*. Springer, 2. Auflage, 1971.
- [68] S. F. BOYS und P. RAJAGOPAL. *Adv. Quant. Chem.*, 2:1, 1965.
- [69] C. B. HASELGROVE. *Math. Comput.*, 15:323, 1961.
- [70] F. W. AVERILL und D. E. ELLIS. *J. Chem. Phys.*, 59:6412, 1973.
- [71] A. D. BECKE und R. M. DICKINSON: *Numerical Solution of Poisson's equation in polyatomic molecules*. *J. Chem. Phys.*, 89(5):2993–2997, September 1988.
- [72] G. TE VELDE und E. J. BAERENDS: *Numerical integration for polyatomic systems*. *J. Comp. Phys.*, 99:84–98, 1992.
- [73] J. C. SLATER: *Quantum Theory of Molecules and Solids*, Band 2. McGraw-Hill, New York, 1965.
- [74] P. M. W. GILL, B. G. JOHNSON und J. A. POPLE. *Chem. Phys. Lett.*, 209:506, 1993.
- [75] J. M. PÉREZ-JORDÁ und A. D. BECKE: *Automatic numerical integration techniques for polyatomic molecules*. *J. Chem. Phys.*, 100(9):6520–6534, May 1994.
- [76] C. W. MURRAY, N. C. HANDY und G. J. LAMING: *Quadrature schemes for integrals of density functional theory*. *Mol. Phys.*, 78:997, 1993.
- [77] J. C. SLATER. *J. Chem. Phys.*, 41:3199, 1964.
- [78] M. ABRAMOWITZ und I. A. STEGUN (Herausgeber): *Handbook of Mathematical Functions*. Dover, New York, 1970.
- [79] A. H. STROUD: *Approximate Calculation of Multiple Integrals*. Prentice-Hall, Englewood Cliffs, NJ, 1971.
- [80] G. H. GOLUB und J. H. WELSCH: *Calculation of Gaussian Quadrature Rules*. *Mathematics of Computation*, 23(221–230), 1969.
- [81] S. L. SOBOLEV: *Cubature formulas on the sphere invariant under finite groups of rotations*. *Sov. Math. Dokl.*, 3, 1962.

- [82] S. L. SOBOLEV: *Mechanical quadrature formulas on spherical surfaces*. Sib. Math. J., 3(5), 1962.
- [83] V. I. LEBEDEV: *Values of nodes and weights for spherical quadrature formulas of Gauss-Markov type exact to order 9–17 and invariant under the octahedral group with inversion*. Zh. Vychisl. Mat. Mat. Fiz., 15:48–54, 1975.
- [84] V. I. LEBEDEV: *On a type of quadrature formulas of increased algebraic accuracy for a sphere*. Sov. Math. Dokl., 17(6):1515–1517, 1976.
- [85] V. I. LEBEDEV: *Quadratures on the sphere*. Zh. Vychisl. Mat. Mat. Fiz., 16(2):293–306, 1976.
- [86] V. I. LEBEDEV: *Spherical Quadrature Formulas exact to Orders 25–29*. Sib. Math. J., 18(1):99–107, September 1977. Übersetzung von Sibirskii Matematicheskii Zhurnal, Vol. 18, No. 1, pp. 132–142.
- [87] V. I. LEBEDEV: *Theory of Cubature of Formulas and Numerical Mathematics*. Diff. Eqn. Numer. Math., Seite 110, 1980.
- [88] V. I. LEBEDEV und A. L. SKOROKHODOV: *Quadrature formulas of orders 41, 47, and 53 for the sphere*. Russian Acad. Sci. Dokl. Math., 45(3):587–592, 1992. Übersetzung von Dokl. Akad. Nauk Vol. 324(1992), 519–524.
- [89] V. I. LEBEDEV: *A quadrature formula for the sphere of 59th algebraic order of accuracy*. Russian Acad. Sci. Dokl. Math., 50(2):283–286, 1995. Übersetzung von Dokl. Akad. Nauk Vol. 338(1994), No. 4.
- [90] P. J. DAVIS und P. RABINOWITZ: *Methods of Numerical Integration*. Academic, San Diego, 2. Auflage, 1984.
- [91] K. E. IVERSON: *A Programming Language*. Wiley, 1962.
- [92] F. N. FRITSCH, D. K. KAHANER und J. N. LYNESS: *Adaptive Double Integration*. ACM Transactions on Mathematical Software, 7(1):46–75, March 1981.
- [93] K. SCHULZE: *Quasimolekulare Röntgenstrahlung aus Schwerionenstößen mit hochionisierten Projektilen*. Dissertation, Universität Kassel, May 1998.
- [94] J. M. PÉREZ-JORDÁ und W. YANG: *An algorithm for 3D numerical integration that scales linearly with the size of the molecule*. Chem. Phys. Lett., Seiten 469–476, July 1995.
- [95] C. FONSECA GUERRA, J. G. SNIJDERS, G. TE VELDE und E. J. BAERENDS: *Towards an order- N DFT method*. Theoret. Chim. Acta, 99:391–403, September 1998.
- [96] C. CASO und OTHERS. Europ. Phys. J., C3(1), 1998.
- [97] D. WELLS und C. YOUNG: *The Waite Group's Ray Tracing Creations*. Waite Group Press, 1993.

- [98] P. M. W. GILL, B. G. JOHNSON, J. A. POPLE und M. J. FRISCH: *The performance of the Becke-Lee-Yang-Parry (B-LYP) density functional theory with various basis sets.* Chem. Phys. Lett., 197(4,5):499–505, September 1998.

Danksagung

Ich bedanke mich sehr herzlich bei Herrn Prof. Dr. B. Fricke für die Aufnahme in die Arbeitsgruppe und die Ermöglichung dieser Arbeit unter hervorragenden Bedingungen.

Bei Herrn Dr. Wolf-Dieter Sepp möchte ich mich für die zahlreichen Diskussionen bedanken, die sich nicht immer nur um die Grundlagen der Physik drehten, die mir oft aber überraschende Einblicke und Sichtweisen eröffneten. Dass dabei manchmal die Zeit in Vergessenheit geriet, zeigt nur, wie interessant die Gespräche waren.

Besonders danken möchte ich auch Herrn Dr. Sven Varga und Herrn Dr. Josef Anton, die mir immer die manchmal verschlungenen Wege durch der Molekülstrukturcode zeigen konnten.

Ich möchte weiterhin all denjenigen danken, die mir bei meinen Fragen mit Rat und Tat beiseite standen. Den Mitgliedern meiner Arbeitsgruppe möchte ich nicht nur für die Hilfe bei physikalischen und mathematischen Problemen, sondern auch für die angenehme Arbeitsatmosphäre danken.

Bei den Rechenzentren der Technischen Universität Darmstadt, der Gesellschaft für Schwerionenforschung (GSI) in Darmstadt, der Universität Frankfurt und der Universität Kassel bedanke ich mich für die zur Verfügung gestellten Möglichkeiten der Rechnernutzung im Höchstleistungs-Rechnerverbund (HHLR).

Der Deutschen Forschungsgemeinschaft (DFG) danke ich für die finanzielle Unterstützung.

Lebenslauf

Anschrift: Andreas Heitmann
Am Klosterhof 1
34132 Kassel

Persönliche Daten: Geb. am 17. Oktober 1969 in Wolfhagen, ledig

Eltern: August Heitmann
Erika Heitmann, geb. Reis

Ausbildung: 1976-1980 Grundschule in Emstal
1980-1989 König-Heinrich Schule in Fritzlar mit Abschluß Abitur

Studium: 1989-1996 Hochschulstudium an der Technischen Hochschule
Darmstadt mit Abschluß Diplom

Berufstätigkeit: seit Oktober 1996 wissenschaftlicher Mitarbeiter in der Arbeitsgruppe
Theoretische Physik der Universität Kassel

Erklärung

Hiermit versichere ich, dass ich die vorliegende Dissertation selbstständig und ohne unerlaubte Hilfe angefertigt und andere als die in der Dissertation angegebenen Hilfsmittel nicht benutzt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen sind, habe ich als solche kenntlich gemacht. Kein Teil dieser Arbeit ist in einem anderen Promotions- oder Habilitationsverfahren verwendet worden.