



Massiv parallele Implementierung eines zustandsselektierenden MRCI-Algorithmus

Dissertation

zur Erlangung des Grades eines
Doktors der Naturwissenschaften
der Abteilung Physik
der Universität Dortmund

vorgelegt von
Philipp Stampfuß

April 2002

Tag der mündlichen Prüfung	24. Mai 2002
Vorsitzender und Prodekan	Prof. Dr. K. Wille
1. Berichterstatter	Priv.-Doz. Dr. W. Wenzel
2. Berichterstatter	Prof. Dr. H. Keiter
Vertreter der promovierten wissenschaftlichen Mitarbeiter	Dr. D. Lüke

Inhaltsverzeichnis

1	Einleitung	1
2	Theoretische Grundlagen	3
2.1	Die Multireference Configuration-Interaction Methode	3
2.2	Iteration des Zustandsvektors	9
2.3	Partitionierung des Orbitalraums	11
2.4	Basis Set Reduction	14
2.5	Extensivität	17
2.6	Ablauf quantenchemischer Rechnungen	25
3	Der massiv parallele MRCI-Algorithmus	28
3.1	Der Residuenbaum	28
3.2	Integraltriebener Algorithmus	35
3.3	Parallele Implementierung	39
3.4	Programmierung in C++	43
4	Benchmarkrechnungen	49
4.1	Leistung des nicht integralgetriebenen MRD-CI Algorithmus	49
4.2	Leistung des integralgetriebenen Algorithmus und des Gesamtpakets	56
4.3	Performance-Daten	58

5 Anwendungen	61
5.1 Elektronenaffinität von BO, NO und O ₂	61
5.2 Endiine	70
5.3 VF ₂ und VCl ₂	73
5.4 Konische Überschneidung von NO ₂	83
6 Zusammenfassung	90
Literaturverzeichnis	93

Kapitel 1

Einleitung

Die Quantenchemie hat es sich zur Aufgabe gemacht, die physikalischen und chemischen Eigenschaften von Molekülen mit Mitteln der Quantenmechanik vorausszusagen. Von ihrem Ziel, experimentelle Studien auf ein Minimum zu reduzieren oder sogar überflüssig zu machen, ist sie jedoch noch weit entfernt. Der Grund hierfür liegt in den komplexen Vielteilchenwechselwirkungen zwischen den Elektronen und den Atomkernen des Moleküls. Den Ausgangspunkt für quantenchemische Rechnungen bildet naturgemäß die Vielteilchen-Schrödinger-Gleichung, die vielen quantenchemischen Methoden durch die Born-Oppenheimer-Näherung vereinfacht wird.

Der immense numerische Aufwand, den eine approximative direkte Lösung der Schrödingergleichung erfordert, wird vermieden durch die heutzutage häufig benutzten *Dichtefunktionalmethoden (DFT)*, in denen das Vielteilchenproblem auf ein effektives Einteilchenproblem reduziert wird. Es hat sich gezeigt, dass diese relativ effizienten Rechnungen in vielen Fällen eine ausreichende Genauigkeit liefern. So sind sie beispielsweise hervorragend geeignet zur Bestimmung von Molekülgeometrien. Schwierigkeiten hat die DFT bei der Behandlung stark korrelierter Systeme und bei der Berechnung von Anregungsenergien.

Ebenfalls eine effektive Einteilchentheorie ist die Hartree-Fock-Näherung, die die Grundlage für die meisten weiterführenden quantenchemischen Verfahren bildet. Sie entspricht der einfachen Modellvorstellung, dass jedes Elektron ein molekulares Orbital besetzt. Für hochgenaue quantenchemische Berechnungen haben sich in den letzten Jahren zwei Familien von Methoden etabliert, die jeweils Verbesserungen der Hartree-Fock-Näherung sind: Zum einen die *Coupled Cluster (CC)*-Methode, zum anderen die *Multireference Configuration-Interaction*-Methode, mit der sich diese Arbeit beschäftigt. Von beiden Methoden existiert eine Vielzahl abgeleiteter Varianten. Beide Verfahren gehen von einer expliziten Darstellung der Vielteilchenwellenfunktion als Linearkombination von Slater-Determinanten aus. Beide nutzen die Tatsache aus, dass nur einige wenige Determinanten signifikant zur Wellenfunktion beitragen.

Zur Diagonalisierung des Hamilton-Operators wird in der MRCI-Methode das Variationsprinzip benutzt. Das MRCI-Verfahren ist daher konzeptuell einfach, aber numerisch

komplex: Zwar tragen nur einige wenige Determinanten signifikant zur Wellenfunktion bei, dennoch konvergiert die Entwicklung so langsam, dass eine Verwendung von bis zu einigen Millionen (bei großen Molekülen sogar einigen Milliarden) Determinanten notwendig ist. Herkömmliche MRCI-Implementierungen stoßen daher auf heutigen Computern bei Systemen einer Größe von 20 Elektronen an ihre Grenzen. Von Buenker und Peyerimhoff wurde aus diesem Grund das zustandsselektierende MRCI-Verfahren entwickelt. In dieser Variante wird nur ein Teil der Determinanten des MRCI-Hilbertraums für die variationelle Wellenfunktion selektiert, der Rest wird in zweiter Ordnung Rayleigh-Schrödinger-Störungstheorie behandelt. Doch auch in dieser Variante bleibt das Verfahren numerisch aufwändig.

Um die Anwendung des zustandsselektierenden MRCI-Verfahrens auf größere Systeme zu ermöglichen ist es daher wünschenswert, die heutzutage leistungsfähigsten verfügbaren Computer zu nutzen. Dabei handelt es sich um sog. *distributed memory* Parallelrechner, eine Rechnerarchitektur, in der eine Vielzahl von Einzelrechnern (Knoten) durch ein schnelles Netzwerk verbunden ist. Selbst kleinere Parallelrechner verfügen über einen Hauptspeicher von mehreren Gigabytes, kosten dabei aber nur den Bruchteil eines vergleichbaren Vektorrechners. Theoretisch können die Leistungsfähigkeit und der Hauptspeicher eines Parallelrechners durch Hinzufügen weiterer Knoten beliebig gesteigert werden. Der Nachteil des Parallelrechners liegt darin, dass bestehende Implementierungen nicht einfach auf einen Parallelrechner übertragen werden können. Es werden spezielle Algorithmen benötigt, die der Tatsache Rechnung tragen müssen, dass die einzelnen Recheneinheiten zunächst nur auf den lokal vorhandenen Speicher zugreifen können. Die Knoten müssen also effektiv auf den lokal vorhandenen Daten operieren können, um die ineffektiven Kommunikationsprozesse auf ein Minimum zu reduzieren. In dieser Arbeit wird ein Algorithmus vorgestellt, der es erstmals erlaubt, das MRCI-Verfahren in seiner zustandsselektierenden Variante auf massiv parallelen Rechnerarchitekturen zur Anwendung zu bringen.

Die Arbeit ist dabei wie folgt gegliedert: In Kapitel 2 wird die MRCI-Methode und ihre zustandsselektierende Variante beschrieben. Anschließend wird in Kapitel 3 der massiv-parallele zustandsselektierende MRCI-Algorithmus vorgestellt. Die Leistungsfähigkeit des Algorithmus wird anhand einiger Benchmarkrechnungen in Kapitel 4 überprüft. Schließlich enthält Kapitel 5 einige Anwendungen des Programms auf physikalisch chemische Problemstellungen.

Kapitel 2

Theoretische Grundlagen

2.1 Die Multireference Configuration-Interaction Methode

In quantenchemischen *ab initio* Rechnungen wird die nichtrelativistische Schrödinger-Gleichung im Rahmen der Born-Oppenheimer Näherung gelöst. Der auf das elektronische System beschränkte Hamiltonoperator nimmt in atomaren Einheiten folgende Form an:

$$\hat{H} = -\sum_{i=1}^{n_e} \frac{1}{2} \nabla_i^2 - \sum_{i=1}^{n_e} \sum_{A=1}^M \frac{Z_A}{|\vec{r}_i - \vec{R}_A|} + \sum_{i=1}^{n_e} \sum_{j>i}^{n_e} \frac{1}{|\vec{r}_i - \vec{r}_j|} \quad (2.1)$$

$$+ \sum_{A=1}^M \sum_{B>A}^M \frac{Z_A Z_B}{|\vec{R}_A - \vec{R}_B|} \quad (2.2)$$

$$\vec{r}_i : \text{Position des Elektrons } i. \quad (2.3)$$

$$n_e : \text{Anzahl der Elektronen} \quad (2.4)$$

$$\vec{R}_A : \text{Position des Atomkerns } A. \quad (2.5)$$

$$Z_A : \text{Kernladungszahl des Atomkerns } A. \quad (2.6)$$

$$M : \text{Anzahl der Atomkerne} \quad (2.7)$$

Die Positionen der Atomkerne gehen lediglich als Parameter ein. Die Verwendung atomarer Einheiten bedeutet, dass die Energien in Hartree (H) und Längen in Einheiten des Bohrschen Radius a_0 gemessen werden.

Im obigen Hamiltonoperator treten Ein- und Zweiteilchenterme auf; in zweiter Quantisierung lässt er sich daher darstellen durch

$$\hat{H} = \sum_{ij} V_{ij} c_i^\dagger c_j + \sum_{ijkl} V_{ijkl} c_i^\dagger c_j^\dagger c_k c_l. \quad (2.8)$$

Die Vielteilchenwellenfunktion wird innerhalb unseres Programms als Linearkombination von Slater-Determinanten dargestellt, die das antisymmetrische äußere Produkt von Spinorbitalen sind. Unter einem Spinorbital versteht man eine Einteilchenwellenfunktion, die sowohl die räumliche Verteilung als auch den Spin eines Elektrons beschreibt. Man erhält das Spinorbital als Produkt einer räumlichen Wellenfunktion $\psi(\vec{r})$ mit einer der orthogonalen Spinfunktionen $\alpha(\omega)$ (Spin up) oder $\beta(\omega)$ (Spin down):

$$\begin{aligned}\chi_{\uparrow}(\mathbf{x}) &= \psi(\vec{r})\alpha(\omega) \\ \chi_{\downarrow}(\mathbf{x}) &= \psi(\vec{r})\beta(\omega)\end{aligned}\quad \mathbf{x} = (\vec{r}, \omega). \quad (2.9)$$

Bei der *Multireference Configuration-Interaction* Methode handelt es sich im Prinzip um eine Verbesserung der Hartree-Fock Näherung (auch *self-consistent-field approximation (SCF)*). Letztere approximiert die elektronische Wellenfunktion durch die einfachste vollständig antisymmetrische Wellenfunktion überhaupt, nämlich durch eine einzige Slater-Determinante:

$$|\Psi_0\rangle = |\chi_1\chi_2 \dots \chi_N\rangle. \quad (2.10)$$

Die bestmögliche Energie im Rahmen dieser 1-Determinanten-Näherung erhält man durch Variation des Energiefunktional

$$E_0 = \frac{\langle \Psi_0 | H | \Psi_0 \rangle}{\langle \Psi_0 | \Psi_0 \rangle} \quad (2.11)$$

nach den Orbitalen χ_i . Nebenbedingung bei der Variation ist die Orthogonalität der Spinorbitale χ_i .

In der Praxis werden Hartree-Fock Rechnungen effizient durchgeführt, indem die Orbitale durch eine Linearkombination von Gaußfunktionen dargestellt werden. Diese Darstellung der Orbitale geht auf *John Pople et al.* [1] zurück; der Name des wohl bekanntesten Quantenchemiepakets *Gaussian* leitet sich davon ab. Den Satz an verwendeten Gaußorbitalen bezeichnet man als Basissatz.

In einer *minimalen Basis* beschränkt man sich, wie der Name schon sagt, auf die absolut notwendigen Orbitale. Das sind in der Regel die Orbitale der besetzten Schalen, zum Beispiel bei Sauerstoff die Orbitale der ersten und zweiten Schale. Nimmt man für die Valenzorbitale (bei Sauerstoff also 2s und 2p) jeweils eine weitere Funktion hinzu (3s und 3p), spricht man von einer *double zeta-Basis* (DZ). Bei Hinzunahme von zwei weiteren Funktionen pro Valenzorbital (3s,3p,4s und 4p) spricht man von einer *triple zeta-Basis* (TZ), usw..

In der Praxis hat es sich jedoch nicht bewährt, zu größeren Basissätzen überzugehen, ohne gleichzeitig auch Funktionen mit höherem Drehimpuls, sog. Polarisationsfunktionen, hinzuzunehmen. Die Basissätze werden daher immer um komplette Schalen vergrößert, wie es in Tabelle 2.1 am Beispiel von Sauerstoff illustriert ist. Eine weitere Möglichkeit zur Vergrößerung der Basissätze ist die Hinzunahme s-artiger, diffuser Funktionen. Man

minimal	1s 2s 2p
DZP	+3s 3p 3d
TZP	+4s 4p 4d 4f
QZP	+5s 5p 5d 5f 5g

Tabelle 2.1: Basissätze für eine quantenchemische Rechnung am Beispiel von Sauerstoff. In einer minimalen Basis beschränkt man sich auf die absolut notwendigen Orbitale (1s, 2s, 2p). Um die Basis zu vergrößern, nimmt man weitere Schalen hinzu und spricht von einer double zeta Basis (+3s3p3d), triple zeta Basis (+4s4p4d4f), ...

Basis	SD	TQ	FCI
DZP	3821	$1 \cdot 10^6$	$9 \cdot 10^6$
TZP	39571	$1 \cdot 10^8$	$4 \cdot 10^{10}$
QZP	137133	$3 \cdot 10^8$	$8 \cdot 10^{14}$

Tabelle 2.2: Größe des Hilbertraums verschiedener Anregungsstufen in Abhängigkeit vom verwendeten Basissatz am Beispiel Sauerstoffs. SD: Ein- und Zweifachanregungen, TQ: Drei- und Vierfachanregungen, FCI: Full-CI. Basissätze: DZP : *double zeta*, TZP: *triple zeta*, QZP: *quadruple zeta*.

spricht dann von *augmentierten* Basissätzen. Der Übergang zu ein- und zweifach augmentierten Basissätzen ist insbesondere bei der Behandlung schwach gebundener Elektronen, z.B. bei Anionen, sinnvoll.

Die Hartree-Fock Rechnung liefert nun einen Satz von Molekülorbitalen, von denen $n_e/2$ besetzt sind und $N - n_e/2$ unbesetzt sind, wobei n_e die Anzahl der Elektronen und N die Größe der Basis ist (Abb. 2.1). Die Anzahl der unbesetzten Orbitale ist in praktischen Rechnungen meist viel größer als die der besetzten Orbitale.

Ausgehend vom Hartree-Fock Grundzustand kann man die CI-Entwicklung der Wellenfunktion vornehmen: Man bildet zunächst alle Einfachanregungen, in denen ein besetztes Orbital a durch ein unbesetztes Orbital r ersetzt wird, dann alle Zweifachanregungen, in denen zwei besetzte Orbitale a und b durch zwei unbesetzte Orbitale r und s ersetzt werden, usw.:

$$\begin{aligned}
 |\Phi_0\rangle &= c_0 |\Psi_0\rangle + \sum_{ar} c_a^r |\Psi_a^r\rangle + \sum_{\substack{a<b \\ r<s}} c_{ab}^{rs} |\Psi_{ab}^{rs}\rangle \\
 &+ \sum_{\substack{a<b<c \\ r<s<t}} c_{abc}^{rst} |\Psi_{abc}^{rst}\rangle + \sum_{\substack{a<b<c<d \\ r<s<t<u}} c_{abcd}^{rstu} |\Psi_{abcd}^{rstu}\rangle + \dots
 \end{aligned} \tag{2.12}$$

Die Diagonalisierung des Hamiltonoperators in den so erhaltenen Konfigurationen bezeichnet man als *Full CI (FCI)*. Sie ist exakt im Rahmen der gewählten Basis.

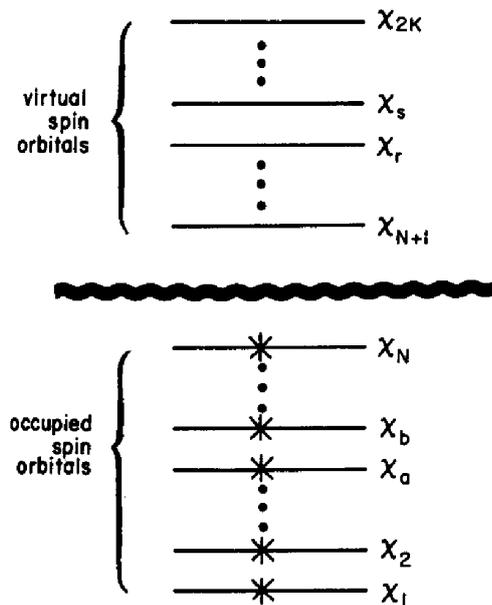


Abbildung 2.1: Der Hartree-Fock Grundzustand. Die ersten $n_e/2$ Orbitale sind doppelt besetzt, die weiteren $N - n_e/2$ Orbitale sind unbesetzt. Die Anzahl der unbesetzten Orbitale ist in der Regel viel größer als die Anzahl der besetzten Orbitale.

Praktisch sind FCI-Rechnungen nur für sehr kleine Moleküle und unter Verwendung sehr kleiner Basissätze durchzuführen. In Tabelle 2.1 ist die Größe des Hilbertraums für verschiedene Anregungsstufen am Beispiel des Sauerstoffmoleküls aufgelistet. Schon der Raum der Drei- und Vierfachanregungen wird zu schnell zu groß, so dass man sich in der Regel auf die ein- und zweifach angeregten Zustände beschränkt (*singles-doubles CI* (*SDCI*)). Insgesamt wächst die Anzahl der n -fach angeregten Konfigurationen mit

$$\binom{n_e}{n} \binom{N - n_e}{n} \approx (N n_e)^n, \quad (2.13)$$

also exponentiell.

Der Rechenaufwand für eine SDCI lässt sich folgendermaßen abschätzen: Der Hauptaufwand entfällt auf die Berechnung der Matrixelemente zwischen den doppelt angeregten Konfigurationen. Davon gibt es nach Gleichung (2.13) ungefähr $(N n_e)^2$. Der Zweiteilchen Hamiltonoperator vernichtet zwei Orbitale und erzeugt zwei neue; pro Doppelanregung gibt es also etwa N^2 zu berechnende Matrixelemente. Der Rechenaufwand skaliert also ungefähr mit

$$\text{SDCI} \propto n_e^2 N^4. \quad (2.14)$$

Eine erhebliche Verbesserung der Genauigkeit von *SDCI*-Rechnungen erreicht man, wenn Ein- und Zweifachanregungen nicht nur der Hartree-Fock Konfiguration, sondern mehrerer *Referenzkonfigurationen* zugelassen werden. Man spricht dann von *Multireference*

Configuration-Interaction (MRCI). Der Rechenaufwand muss im Vergleich zur SDCI mit der vergleichsweise geringen Zahl an Referenzen multipliziert werden:

$$\text{SD-MRCI} \propto N_{\text{ref}} n_e^2 N^4. \quad (2.15)$$

Die Referenzen werden nach chemischen Gesichtspunkten ausgewählt. Hiermit befasst sich auch Abschnitt 2.3.

Trotz der Beschränkung auf ein- und zweifach angeregte Zustände, bleibt der Rechenzeitaufwand für eine MRCI recht hoch, so dass solche Rechnungen bisher auf kleinere Systeme beschränkt waren. Aus diesem Grund wurde von Buenker und Peyerimhoff [2][3][4] die *zustandsselektierende MRCI (MRD-CI)* eingeführt. In dieser Variante wird nur ein Teil der Konfigurationen des Hilbertraums für die variationelle Bestimmung der Grundzustandsenergie selektiert. Der Energiebeitrag der übrigen Konfigurationen wird durch Rayleigh-Schrödinger Störungstheorie zweiter Ordnung abgeschätzt. Dazu wird zunächst für jede Konfiguration $|\phi_x\rangle$ aus dem Raum der Ein- und Zweifachanregungen das zweidimensionale Eigenwertproblem

$$\begin{pmatrix} \langle \Psi_p^{(0)} | H | \Psi_p^{(0)} \rangle & \langle \Psi_p^{(0)} | H | \phi_x \rangle \\ \langle \phi_x | H | \Psi_p^{(0)} \rangle & \langle \phi_x | H | \phi_x \rangle \end{pmatrix} \begin{pmatrix} 1 \\ c_x \end{pmatrix} = (E^{(0)} + \delta E_x) \begin{pmatrix} 1 \\ c_x \end{pmatrix} \quad (2.16)$$

gelöst. $\Psi_p^{(0)}$ ist hierbei die Wellenfunktion der Referenzkonfigurationen. Eine Konfiguration wird in die variationelle Wellenfunktion ϕ_v aufgenommen, wenn der berechnete Koeffizient c_x (oder die berechnete Energiekorrektur δE_x) größer als ein gegebener Schwellwert λ_c (λ_E) ist. Ansonsten addiert man nur die Energiekorrekturen δE_x auf:

$$E_P = \sum_x \delta E_x \quad \text{mit } c_x < \lambda_c \quad (2.17)$$

Die totale Energie ergibt sich dann als Summe von variationeller Energie und perturbativer Korrektur und wird für $\lambda \rightarrow 0$ extrapoliert.

Das MRD-CI Verfahren ist in Abb. 2.2 am Beispiel einer Rechnung an Sauerstoff illustriert. Aufgetragen sind die totale Energie und die Energie des variationellen Subraums gegen den Schwellwert. Man sieht, dass die perturbative Energiekorrektur bei großen Schwellwerten zu groß ausfällt. Das ist auch zu erwarten, da Energien in zweiter Ordnung Rayleigh-Schrödinger Störungstheorie in der Regel überschätzt werden. Diese Überschätzung ist jedoch meistens systematisch, wie man am Beispiel in Abb. 2.3 erkennt. Aufgetragen sind hier die totalen und variationellen Grundzustandsenergien (E_T und E_V) von Endiin und didehydro-para-Benzol gegen die perturbative Energiekorrektur. Auf diese Moleküle wird in Abschnitt 5.2 noch näher eingegangen werden. Die berechneten Energien liegen auf einer Gerade, d.h. dass die perturbative Energiekorrektur um einen konstanten Faktor überschätzt wird:

$$E_T(E_P) = E_0 + cE_P \quad (2.18)$$

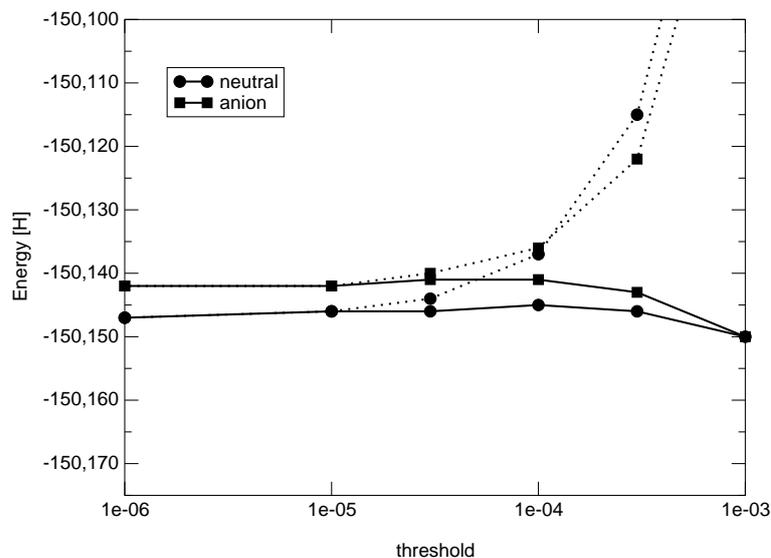


Abbildung 2.2: MRD-CI Rechnung an Sauerstoff. Aufgetragen sind die totalen Energien (durchgezogene Linien) und die variationellen Energien (gestrichelte Linien) des Sauerstoffmoleküls und seines Anions gegen den Schwellwert für die Koeffizienten. Bei großen Schwellwerten erfolgt eine große Überschätzung des Energiebeitrags der nichtselektierten Konfigurationen.

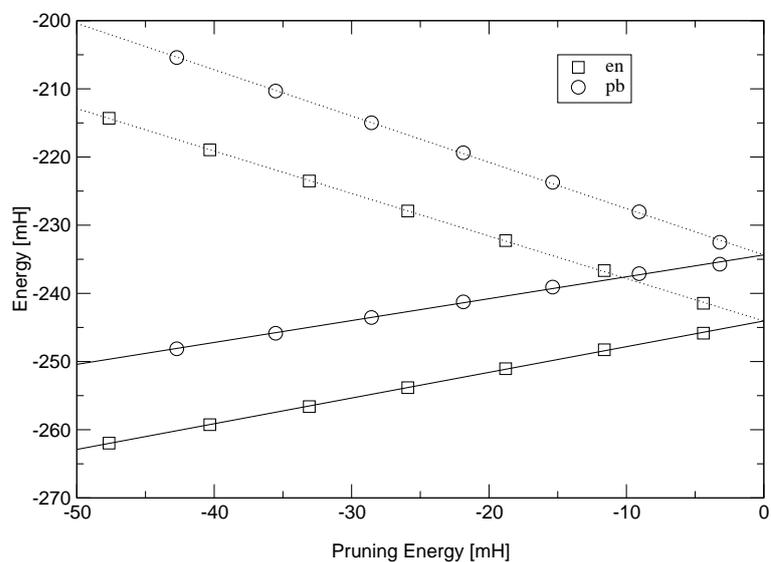


Abbildung 2.3: MRD-CI Rechnung an Endiin und didehydro-para-Benzol. Aufgetragen sind die totalen Energien (durchgezogene Linien) und die variationellen Energien (gestrichelte Linien) gegen die perturbative Energiekorrektur E_P . Die Überschätzung der perturbativen Energiekorrektur ist systematisch und kann für $E_P \rightarrow 0$ extrapoliert werden.

Die Geraden, auf denen die Energien der (einander ähnlichen) Systeme liegen, verlaufen fast parallel. In beiden Systemen wird also die Energiekorrektur um fast denselben Faktor überschätzt. Man sieht auch, dass es zu Fehlern in der Größenordnung der perturbativen Energiekorrektur führen kann, Differenzen von Energien zu berechnen, die bei verschiedenen Schwellwerten berechnet wurden. Es sind auch vereinzelte Fälle bekannt, in denen die Extrapolation $\lambda \rightarrow 0$ fehlschlägt, in der überwiegenden Mehrheit der Anwendungen erhält man jedoch eine bemerkenswert gute Auflösung der Potentialenergieoberflächen. In der Regel erhält man gute Ergebnisse bei einer Selektion von ca. 1% der Konfigurationen des Hilbertraums.

Mit der MRD-CI lassen sich große Hilberträume behandeln. Das von uns implementierten Programm ist in der Lage, bis zu 10^{10} Konfigurationen perturbativ zu behandeln. Im variationellen Schritt kann ein Hilbertraum von bis zu 20 Mio. Konfigurationen diagonalisiert werden. Ein Nachteil der Methode ist, dass der Teilraum der selektierten Konfigurationen keinerlei Struktur aufweist. Die Matrixelemente zwischen diesen Konfigurationen sind daher schwierig zu berechnen. Abhilfe schafft hier das von uns entwickelte residuenbasierte Verfahren zur Auswertung der Matrixelemente [5]. Dieses erlaubt zum einen eine sehr effiziente Berechnung der Matrixelemente auch im seriellen Betrieb des Programms und ist zum anderen der Schlüssel zur massiven Parallelisierung. Das Verfahren wird in Abschnitt 3.1 vorgestellt.

2.2 Iteration des Zustandsvektors

Zur Berechnung der variationellen Grundzustandsenergie wird innerhalb des Programms die von Davidson [6] vorgeschlagene, iterative Methode zur Lösung großer Eigenwertprobleme verwendet. Ausgangspunkt des Davidson-Verfahrens ist die gegenwärtige Approximation des Eigenvektors. Diese sei in der k -ten Iteration gegeben durch

$$|c^k\rangle = \sum_i c_i^k |\phi_i\rangle \quad (2.19)$$

Der approximative Energieeigenwert ist der Erwartungswert des Hamiltonoperators in diesem Zustand:

$$E_0^k = \frac{\langle c^k | \hat{H} | c^k \rangle}{\langle c^k | c^k \rangle} = \frac{\langle c^k | q^k \rangle}{\langle c^k | c^k \rangle} = \frac{\sum_i c_i^k q_i^k}{\sum_i c_i^k c_i^k} \quad (2.20)$$

Die Grundidee der Davidson-Iteration besteht nun darin, die Approximation des Eigenvektors durch Hinzunahme eines Testzustands $|t\rangle$ zu verbessern:

$$|c^{k+1}\rangle = |c^k\rangle + \beta |t^k\rangle, \quad (2.21)$$

wobei die Koeffizienten des Testzustands $|t\rangle$ durch

$$t_i^k = \frac{E_0^k c_i^k - q_i^k}{\langle \phi_i | \hat{H} | \phi_i \rangle - E_0^k} \quad (2.22)$$

gegeben sind. Dieses Ergebnis erhält man, wenn man für jede Konfiguration $|\phi_i\rangle$ aus dem Zustandsvektor $|c^k\rangle$ das zweidimensionale Eigenwertproblem

$$\begin{pmatrix} E_0^k & q_i^k \\ q_i^k & \langle \phi_i | \hat{H} | \phi_i \rangle \end{pmatrix} \begin{pmatrix} 1 \\ t_i^k \end{pmatrix} = E \begin{pmatrix} 1 \\ t_i^k \end{pmatrix} \quad (2.23)$$

löst und dabei die in t_i^k quadratischen Terme vernachlässigt.

Durch Gleichung (2.21) wird ebenfalls ein leicht lösbares zweidimensionales Eigenwertproblem definiert, bei dem jedoch der Tatsache Rechnung getragen werden muss, dass die Vektoren $|c^k\rangle$ und $|t^k\rangle$ in der Regel nicht orthogonal sind:

$$\begin{pmatrix} \langle c^k | \hat{H} | c^k \rangle & \langle c^k | \hat{H} | t^k \rangle \\ \langle t^k | \hat{H} | c^k \rangle & \langle t^k | \hat{H} | t^k \rangle \end{pmatrix} \begin{pmatrix} 1 \\ \beta \end{pmatrix} = E_0^{k+1} \begin{pmatrix} \langle c^k | c^k \rangle & \langle t^k | c^k \rangle \\ \langle c^k | t^k \rangle & \langle t^k | t^k \rangle \end{pmatrix} \begin{pmatrix} 1 \\ \beta \end{pmatrix}. \quad (2.24)$$

Dieses Eigenwertproblem wird exakt gelöst und daraus der Startvektor $|c^{k+1}\rangle$ für die nächste Iteration bestimmt. Es hat sich gezeigt, dass man die Konvergenz des Davidson-Verfahrens erheblich beschleunigen kann, wenn man auch die Vektoren $|c^k\rangle$ und die Felder $|q^k\rangle$ aus vorangegangenen Iterationen zur Berechnung des neuen Zustandsvektors verwendet. Gleichung (2.21) ist dann durch

$$|c^{k+1}\rangle = |c^k\rangle + \beta_1 |t^k\rangle + \beta_2 |t^{k-1}\rangle + \dots \quad (2.25)$$

zu verallgemeinern; statt des Eigenwertproblems (2.24) erhält man ein entsprechendes höherdimensionales Eigenwertproblem. Die Anzahl der erforderlichen Iterationen schwankt in praktischen Rechnungen zwischen 6 und 10 bei einer Genauigkeit von 10^{-5} .

Hat man erst einmal das Vielteilchenfeld

$$|q^k\rangle = \hat{H} |c^k\rangle \quad (2.26)$$

bestimmt, so erfolgt die Berechnung des Testzustands $|t^k\rangle$ nach Gleichung (2.22) und die Berechnung des neuen Zustandsvektors $|c^k\rangle$ in Bruchteilen von Sekunden. Auf die Berechnung des Feldes $|q^k\rangle$ entfällt damit der Hauptrechenaufwand im Iterationsschritt.

Das Davidson-Verfahren findet an zwei Stellen im Programmablauf Verwendung: Zum einen muß vor der Expansion der Referenzraum diagonalisiert werden, zum anderen nach der Expansion die variationelle Wellenfunktion. In beiden Fällen steht man vor dem Problem, einen geeigneten Startvektor für die Davidson-Iteration zu wählen. Nach der Expansion sind das natürlich die perturbativ bestimmten Koeffizienten der selektierten Konfigurationen. Zur Diagonalisierung des Referenzraums sind wir zunächst so vorgegangen, nur den Koeffizienten der Konfiguration mit der niedrigsten Energie auf 1.0 zu setzen und alle übrigen Koeffizienten auf 0.0. Das hat jedoch in einigen Fällen zu numerischen Instabilitäten geführt, insbesondere wenn es mehrere Konfigurationen mit derselben Energie gab. Es hat sich daher als sinnvoll erwiesen, einen Raum von maximal 100 Konfigurationen exakt zu diagonalisieren. Durch die Wahl von maximal 100 Konfigurationen ist die Zeit, die dieser Schritt benötigt, vernachlässigbar klein.

2.3 Partitionierung des Orbitalraums

Die Dichtematrix in ihrer diskreten Darstellung

$$\rho_{ji} = \langle \Phi | c_i^\dagger c_j | \Phi \rangle \quad (2.27)$$

ist in der Basis der kanonischen Hartree-Fock Spinorbitale nicht diagonal. Da ρ hermitesch ist, lässt sich jedoch eine unitäre Transformation finden, die die Dichtematrix diagonalisiert. Die solchermaßen transformierten Spinorbitale wurden von Löwdin [7] eingeführt und werden als *natürliche Orbitale (NO)* bezeichnet.

Der Vorteil natürlicher Orbitale liegt darin, dass die Konvergenz der CI-Entwicklung beschleunigt wird. Zur Energie tragen hauptsächlich die Konfigurationen bei, die aus natürlichen Orbitalen mit großer Besetzungszahl gebildet werden. Es werden daher weniger Konfigurationen als in jeder anderen orthonormalen Basis benötigt, um dieselbe Genauigkeit in der Energie zu erzielen.

Strenggenommen können natürliche Orbitale erst *nach* der MRCI Rechnung bestimmt werden. Bender und Davidson [8] schlugen daher eine iterative NO Berechnung vor, in der nach jeder MRCI Rechnung neue natürliche Orbitale berechnet werden, bis Selbstkonsistenz erreicht ist.

In der Praxis wird Selbstkonsistenz nach 5–10 NO Iterationen erreicht. Das würde einen erheblichen Mehraufwand an Rechenleistung bedeuten, wenn jedesmal eine volle MRCI Rechnung durchgeführt würde. Es zeigt sich jedoch, dass natürliche Orbitale, die mit der perturbativen *Basis Set Reduction (BSR)* Methode (s. Abschnitt 2.4) bestimmt wurden, qualitativ genauso gut sind wie aus MRCI Rechnungen bestimmte natürliche Orbitale. Der Rechenaufwand ist jedoch erheblich geringer.

In Abb. 2.3 sind die Besetzungszahlen der natürlichen Orbitale des Sauerstoffmoleküls, also der diagonalen Elemente der Dichtematrix, in einer cc-pVTZ Basis dargestellt, wie sie nach 10 MRCI Iterationen berechnet wurden. Die Besetzungszahlen unterscheiden sich um mehrere Größenordnungen: Während die ersten fünf Orbitale in der MRCI Wellenfunktion fast doppelt besetzt sind, fallen Anregungen in höhere Orbitale kaum ins Gewicht.

Die Orbitale werden nun, entsprechend ihres Gewichts, in *interne* und *externe* Orbitale eingeteilt. Aus den internen Orbitalen werden die Referenzkonfigurationen gebildet, indem alle Slater-Determinanten mit n_e Elektronen und beliebigen Kombinationen aus internen Orbitalen konstruiert werden. Im Raum, der durch die internen Orbitale aufgespannt wird, wird also praktisch eine Full CI durchgeführt. Den Raum der Ein- und Zweifachanregungen erhält man, indem maximal zwei interne Orbitale durch zwei externe Orbitale ersetzt werden.

Häufig ist es sinnvoll, die besetzten Elektronenschalen und die Valenzelektronen als interne Orbitale zu wählen. Für Sauerstoff sind das die 1s-, 2s- und 2p-Orbitale. Diese Wahl würde

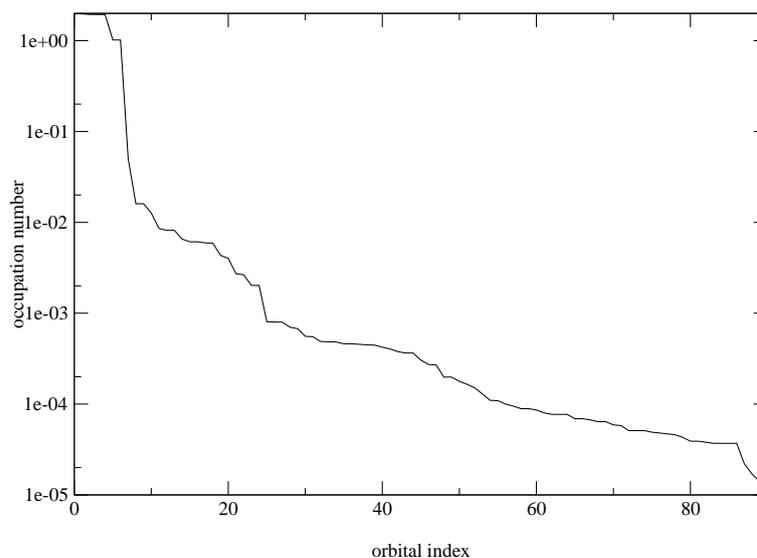


Abbildung 2.4: Besetzungszahlen der natürlichen Orbitale von Sauerstoff in einer cc-pVTZ Basis. Aufgetragen ist die Besetzungszahl gegen den Index des jeweiligen Orbitals. Die Besetzungszahlen unterscheiden sich um mehrere Größenordnungen.

auf 2025 Referenzen führen, was sich aber durch Ausnutzung von Symmetrien auf 212 Referenzen reduzieren läßt.

Anregungen aus der geschlossenen 1s-Schale spielen jedoch praktisch keine Rolle. Diese Orbitale werden daher nach der Hartree-Fock Rechnung *eingefroren* und ihr Beitrag zu dem der Kerne addiert. Die Zahl der Referenzen reduziert sich durch die eingefrorenen Orbitale weiter auf 96.

Bei großen Molekülen führt das beschriebene Verfahren jedoch zu sehr großen Referenzräumen. Um die Zahl der Referenzen weiter einzuschränken, teilt man die internen Orbitale wiederum in zwei Klassen, in *inaktive* und *aktive* Orbitale. Hierbei gilt für die inaktiven Orbitale, dass sie innerhalb des Referenzraums doppelt besetzt sein müssen. Bei einem so gewählten Referenzraum spricht man von einem *complete active space (CAS)*. Durch die Einführung von *inaktiven* Orbitalen entsteht auch ein neuer Typ von zweifach angeregten Zuständen, bei denen zwei inaktive Orbitale durch zwei aktive Orbitale ersetzt werden, ohne dass Anregungen in externe Orbitale beteiligt sind.

Neben inaktiven, aktiven und externen Orbitalen unterstützt unser Programmpaket drei weitere Orbitaltypen, die hier der Vollständigkeit halber hier erwähnt werden sollen: *low* Orbitale sind spezielle aktive Orbitale. In einer Referenzkonfiguration dürfen höchstens zwei Orbitale vom Typ *low* besetzt sein. Dieser Orbitaltyp dient dazu, bestimmte Drei- und Vierfachanregungen einzuschalten. Orbitale vom Typ *high* sind spezielle externe Orbitale. Sie sind die Grundlage einer orbitalbasierten Selektion, in der nur Anregungen in die *high* Orbitale für die variationelle Wellenfunktion selektiert werden. *Virtuelle* Orbitale

```

(iA0 aA1 aB0 aC0 aD0 aE0 )(iA0 aA1 aB0 aC0 aE0 aF0 ) -0.688179
(iA0 aA1 aB0 aC0 aD0 aF0 )(iA0 aA1 aB0 aD0 aE0 aF0 ) 0.162445
(iA0 aA1 aB0 aC0 aE0 aF0 )(iA0 aA1 aB0 aC0 aD0 aE0 ) 0.688179
(iA0 aA1 aB0 aD0 aE0 aF0 )(iA0 aA1 aB0 aC0 aD0 aF0 ) -0.162445
.....
(iA0 aB0 aB1 aC0 aD0 aF0 )(iA0 aA1 aB0 aC0 aE0 aF0 ) -0.002743
(iA0 aB0 aB1 aC0 aE0 aF0 )(iA0 aA1 aB0 aC0 aD0 aF0 ) 0.001264
(iA0 aB0 aB1 aC0 aD0 aF0 )(iA0 aB0 aB1 aD0 aE0 aF0 ) -0.000161

```

Abbildung 2.5: Ein Teil der 96 Referenzen einer MRCI-Rechnung an Sauerstoff. Eine Reihe steht für eine Konfiguration. Die linke Klammer beschreibt den *up*-Teil der Konfiguration, die rechte Klammer den *down*-Teil. In der letzten Spalte findet man den Koeffizienten der Konfiguration. Die Orbitale werden durch Angabe von Typ (a,i,...), Symmetrie (A,B,...) und Index beschrieben.

Typ	Symbol	Beschreibung
core	c	Orbitale, die nach der Hartree-Fock Rechnung eingefroren werden.
inactive	i	Interne Orbitale, die in den Referenzen doppelt besetzt sein müssen.
active	a	Interne Orbitale, die in den Referenzen beliebig besetzt sein können.
extern	d	Externe Orbitale, die in den Referenzen nicht vorkommen dürfen.
low	l	Spezielle aktive Orbitale, von denen in einer Referenz höchstens zwei enthalten sein dürfen.
high	h	Spezielle externe Orbitale. Bei Orbitalselektion werden nur Anregungen in high Orbitale selektiert.
virtual	v	Für die MRCI verbotene Orbitale

Tabelle 2.3: Übersicht über die vom vorliegenden Programmpaket unterstützten Orbitaltypen. Die drei letztgenannten Typen finden nur in speziellen Fällen Verwendung.

Symbol	yz	xz	xy
A	gerade	gerade	gerade
B	gerade	gerade	ungerade
C	gerade	ungerade	gerade
D	gerade	ungerade	ungerade
E	ungerade	gerade	gerade
F	ungerade	gerade	ungerade
G	ungerade	ungerade	gerade
H	ungerade	ungerade	ungerade

Tabelle 2.4: Bezeichnung der Symmetrien im vorliegenden Programmpaket bei Invarianz gegen Spiegelungen an der yz-, xz- und xy-Ebene.

schließlich sind Orbitale, die für die MRCI Rechnung verboten sind. Einen Überblick über alle Orbitaltypen gibt Tabelle 2.3.

Abb. 2.5 zeigt einen Teil der Referenzen mit ihren Koeffizienten der MRCI Rechnung an Sauerstoff. Der erste Block steht dabei für den Spin-„up“ Anteil des Zustandes, der zweite für den Spin-„down“ Anteil. Das kleine „a“ steht für *aktiv*. Im gewählten Beispiel wurde keines der Orbitale als inaktiv gewählt. Die Großbuchstaben A, B, ... kennzeichnen das Verhalten des Orbitals bei Spiegelung an den drei Symmetrieebenen (s. Tab. 2.4). Ein Molekül, das sich bei Spiegelungen an allen drei Symmetrieebenen symmetrisch verhält, hat die Symmetriegruppe D_{2h} . Höhere Symmetrien, z.B. die Rotationsinvarianz des Sauerstoffmoleküls für beliebige Drehungen, werden von unserem Programm nicht unterstützt. Die Ziffer ist schließlich ein laufender Index für Orbitale gleichen Typs und gleicher Symmetrie. Die Fließkommazahl gibt den Koeffizienten der Konfiguration an, wie er nach der Diagonalisierung des Referenzraums berechnet wurde.

Man sieht, dass sich die Gewichte der einzelnen Konfigurationen innerhalb der Gesamtwellenfunktion um Größenordnungen unterscheiden. Das legt eine Referenzselektion nahe, bei der aus dem ursprünglichen CAS-Raum nur die Referenzen selektiert werden, deren Koeffizient größer als ein gegebener Schwellwert — typischerweise 0.1 oder 0.05 — ist. Auf diesem Weg läßt sich die Zahl der Referenzen gegenüber einer CAS-Rechnung deutlich reduzieren, ohne einen signifikanten Verlust an Genauigkeit.

2.4 Basis Set Reduction

Eine Näherung der MRCI ist die von Wenzel und Wilson [9] entwickelte *Basis Set Reduction (BSR)*. Der Name ist dadurch motiviert, dass in der BSR Methode die variationellen Freiheitsgrade auf den Subraum der Valenzelektronen reduziert werden. Der Rest des Hilbertraums wird rein perturbativ behandelt.

Der Hilbertraum, der mit dem einer MRCI identisch ist, wird dazu in zwei orthogonale Subräume zerlegt: den *Valenzraum* (P-Raum) und den komplementären *Korrelationsraum* (Q-Raum). Der Valenzraum enthält in der Regel die Referenzen sowie einige zusätzliche (selektierte) Konfigurationen, die rein aus internen Orbitalen bestehen.

Man definiert nun

$$|\Psi_p\rangle = P|\Psi\rangle \quad \text{und} \quad (2.28)$$

$$\begin{aligned} |\Psi_q\rangle &= Q|\Psi\rangle \\ &= \sum c_x |\phi_x\rangle \end{aligned} \quad (2.29)$$

als die *Valenz-* und die *Korrelationswellenfunktion*. P und Q sind hierbei die Projektoren auf den P- bzw. den Q-Raum.

Das BSR Verfahren besteht nun konkret aus folgenden Schritten: In einem ersten Schritt wird der P-Raum exakt diagonalisiert. Für jede Konfiguration $|\phi_x\rangle$ aus dem Q-Raum wird dann die Eigenwertgleichung

$$\begin{pmatrix} \langle \Psi_p^{(0)} | H | \Psi_p^{(0)} \rangle & \langle \Psi_p^{(0)} | H | \phi_x \rangle \\ \langle \phi_x | H | \Psi_p^{(0)} \rangle & \langle \phi_x | H | \phi_x \rangle \end{pmatrix} \begin{pmatrix} 1 \\ c_x \end{pmatrix} = (E^{(0)} + \delta E_x) \begin{pmatrix} 1 \\ c_x \end{pmatrix}, \quad (2.30)$$

gelöst. Die Beiträge der einzelnen Konfigurationen zur Energie

$$E^{(Q)} = \sum_q \delta E^{(q)} = \sum_q c_x^2 \langle \phi_x | H | \phi_x \rangle \quad (2.31)$$

und zur Norm

$$N^{(Q)} = \sum_q N^{(q)} = \sum_q c_x^2 \quad (2.32)$$

werden sofort aufsummiert. Ebenso werden die Komponenten des Feldes $P H |\Psi\rangle$

$$q_i = \langle \phi_i | H | \Psi \rangle. \quad (2.33)$$

sofort aufsummiert, da sie den wichtigsten Beitrag zu einer Relaxation der Valenzwellenfunktion im dritten Schritt der BSR liefern.

Schließlich wird die Energie der Gesamtwellenfunktion durch

$$E^{(n)} = \frac{\langle \Psi_p^{(n)} | H | \Psi_p^{(n)} \rangle + (\langle \Psi_p^{(n)} | H | \Psi_x^{(n)} \rangle + h.c.) + E_Q^{(n)}}{\langle \Psi_p | \Psi_p \rangle + N_Q^{(n)}} \quad (2.34)$$

bestimmt. Das entspricht einer variationellen Bestimmung der Energie des in Abb. 2.6 dargestellten Modell Hamiltonoperators, in dem alle nichtdiagonalen Matrixelemente, die Konfigurationen des Korrelationsraums untereinander koppeln, vernachlässigt werden.

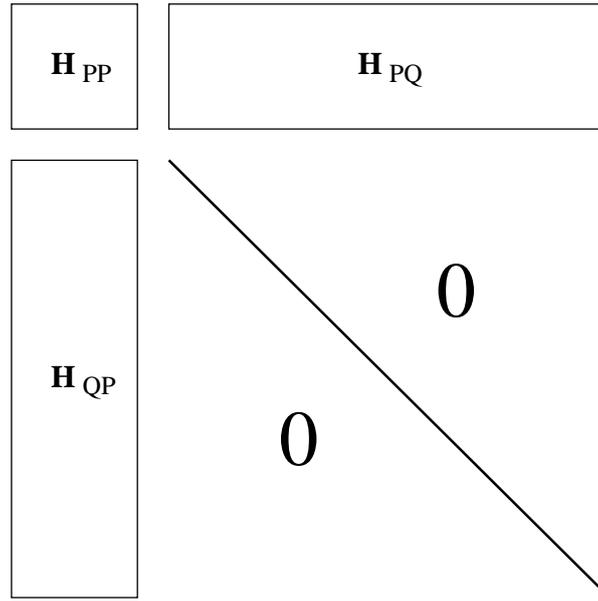


Abbildung 2.6: Schematische Darstellung der von Null verschiedenen Matrixelemente des BSR Hamiltonoperators. Alle nichtdiagonalen Matrixelemente zwischen Konfigurationen des Q-Raums wurden auf Null gesetzt.

Weitere Iterationen sind möglich, indem mit der jeweils relaxierten Wellenfunktion die Schritte zwei und drei wiederholt werden. In der Praxis zeigt sich jedoch, dass die auskonvergierte Energie nur um Bruchteile von milli-Hartree von der BSR Energie nach der ersten Iteration abweicht. Das Verfahren wird daher nach einer Iteration beendet.

Bis auf den Relaxationsschritt ist das Verfahren im wesentlichen mit der selektierenden MRD-CI identisch, sofern man auch in dieser keine weiteren Konfigurationen oder nur rein interne Konfigurationen selektiert. Eine wesentliche Idee des BSR Verfahrens ist eine deutliche Reduzierung des Speicherbedarfs, da die nach Gleichung (2.30) berechneten Koeffizienten der Konfigurationen des Q-Raums nicht abgespeichert werden müssen.

Um die BSR zur Berechnung von natürlichen Orbitalen oder von Erwartungswerten von Einteilchenoperatoren zu verwenden, ist die Berechnung der reduzierten Dichtematrix notwendig. Die Berechnung der vollen Dichtematrix

$$\rho_{ij}^{B_k} = \langle \Phi^{(P)} + \Phi^{(Q)} | c_i^\dagger c_j | \Phi^{(P)} + \Phi^{(Q)} \rangle \quad (2.35)$$

würde jedoch ein Abspeichern der Koeffizienten c_x erfordern. In Konsistenz mit dem Hamiltonoperator in Abb. 2.6 werden wiederum alle Nebendiagonalelemente zwischen Konfigurationen des Q-Raums vernachlässigt, so dass eine vereinfachte Dichtematrix durch

$$\begin{aligned} \rho_{ij}^{BSR} &= \langle \Phi^{(P)} | c_i^\dagger c_j | \Phi^{(P)} \rangle + (\langle \Phi^{(P)} | c_i^\dagger c_j | \Phi^{(Q)} \rangle + h.c.) \\ &\quad + \delta_{ij} \langle \Phi^{(Q)} | c_i^\dagger c_j | \Phi^{(Q)} \rangle \end{aligned} \quad (2.36)$$

gegeben ist. Tatsächlich zeigt sich in der Praxis, dass es keinen Unterschied für die Qualität der natürlichen Orbitale macht, ob man sie mit der BSR Methode generiert, oder ob man eine MRCI Rechnung durchführt, in der die Kopplungen zwischen den Konfigurationen des Q-Raums enthalten sind. Die Ersparnis an Rechenzeit ist hingegen enorm.

2.5 Extensivität

Ein Mangel der MRCI Verfahren liegt im subextensiven Skalierungsverhalten der Korrelationsenergie. So ist beispielsweise die Korrelationsenergie eines dissoziierten Moleküls nicht identisch mit der Summe der atomaren Korrelationsenergien, die mit der gleichen Methode bestimmt wurden. In einem solchen Fall gehen zwar in die atomare Rechnung auch nur maximal Doppelanregungen der Referenzen ein, jedoch enthält die zugehörige Molekülwellenfunktion, gegeben als äußeres Produkt der Atomfunktionen, auch höhere Anregungen. Diese sind in der molekularen MRCI nicht enthalten, was sich in einer betragsmäßig kleineren Korrelationsenergie niederschlägt. Im Grenzfall unendlicher Atomzahlen bei konstant gehaltenem Referenzraum verschwindet die Korrelationsenergie pro Atom sogar gänzlich.

Um das Problem der Nichtextensivität zu beheben, wurden eine Reihe von *a priori* und *a posteriori* Korrekturen entwickelt. Die erste *a posteriori* Methode wurde von Davidson [10] [11] vorgeschlagen. Sie gehört heute noch zu den meistbenutzten Korrekturen. Die Davidson-Korrektur kann abgeleitet werden durch eine asymptotische Entwicklung der SDCI bzw. der FCI Korrelationsenergie für nichtwechselwirkende Cluster. Sie beseitigt den Fehler in der SDCI Korrelationsenergie in niedrigster Ordnung:

$$\Delta E_D = (1 - c_0^2) (E_{\text{SDCI}} - E_{\text{SCF}}) \quad (2.37)$$

c_0 ist hier der Koeffizient der Referenzkonfiguration in der SDCI Wellenfunktion. Die Davidson-Korrektur existiert in der sog. *renormierten* Form:

$$\Delta E_{D,\text{ren.}} = \frac{1 - c_0^2}{c_0^2} (E_{\text{SDCI}} - E_{\text{SCF}}). \quad (2.38)$$

Diese Form der Davidson-Korrektur wurde entwickelt, da Gleichung (2.37) auch im Falle eines Zweielektronensystems eine von Null verschiedene Korrektur liefert, obwohl die SDCI in diesem Fall extensiv ist. Pople [12] schlug eine Korrektur der Form

$$\Delta E'_{D,\text{ren.}} = \left(1 - \frac{2}{n_e}\right) \Delta E_{D,\text{ren.}} \quad (2.39)$$

vor.

Davidsons Methode wurde erweitert auf den Multireferenzfall [13] [14] [15]:

$$\Delta E_D = \left(1 - \sum_{\text{ref}} c_i^2\right) (E_{\text{MRCI}} - E_{\text{ref}}) \quad (2.40)$$

r_{OH}	1.0	1.5	2.0	2.5	3.0	Δ
full CI	-76.24186	-76.07234	-75.95166	-75.91798	-75.91194	0
SDCI	14.72	32.66	72.90	146.36	136.34	131.6
+ ΔE_{D}	4.73	9.52	11.60	(-34.87)	—	6.9
+ $\Delta E_{\text{D,ren.}}$	4.21	7.11	-4.70	(-306.61)	—	11.8
+ ΔE_{P}	6.84	13.50	14.70	(-193.36)	—	7.9
MRCI	6.64	5.83	5.04	4.65	4.56	2.1
+ ΔE_{D}	0.74	0.42	0.84	0.99	1.03	0.6
+ $\Delta E_{\text{D,ren.}}$	0.52	0.21	0.69	0.88	0.92	0.7
+ ΔE_{P}	2.05	1.62	1.78	1.82	1.83	0.4

Tabelle 2.5: Eine illustrative Rechnung zur Dehnung der Bindungslänge von Wasser. r_{OH} bezeichnet den Dehnungsfaktor der Gleichgewichtsbindungslänge $r_0 = 0.975512\text{\AA}$. Der Bindungswinkel betrug $110,6^\circ$, die Rechnungen wurden in einer cc-pVDZ Basis von Dunning durchgeführt. Die FCI Ergebnisse stammen von Olsen *et al.* [16]. Die FCI Energien sind in H, die übrigen Energien in mH relativ zu den FCI Energien angegeben. Δ ist die maximale Differenz der absoluten Fehler.

Die Renormierung nach Gleichung (2.38) oder nach Pople (Gl. (2.39)) erfolgt entsprechend.

All diese Methoden können aus asymptotischen Entwicklungen der SDCI Energie abgeleitet werden. Sie haben den Vorteil, dass keine Änderung des Programmcodes notwendig ist und keine zusätzliche Rechenleistung erforderlich wird. Andererseits verschlechtert sich ihre Qualität mit zunehmender Systemgröße, auch für nicht wechselwirkende Cluster.

Die symmetrische Dehnung der zwei O–H Bindungen von Wasser bei gleichbleibendem Bindungswinkel wurde auf Full-CI Niveau von Olsen *et al.* untersucht. Bei diesen Rechnungen wurde eine cc-pVDZ Basis von Dunning [17] verwendet. Als Gleichgewichtsgeometrie wurde $r_{\text{OH}} = 0.975512\text{\AA}$ und ein Bindungswinkel von $110,6^\circ$ festgesetzt. Um unsere Resultate mit denen dieser Studie zu vergleichen, haben wir Rechnungen in derselben Basis und derselben Geometrie durchgeführt. Die Ergebnisse sind in Tabelle 2.5 dargestellt. Die *a posteriori* Korrekturen verbessern die maximale Differenz der absoluten Fehler Δ einer SDCI Rechnung gegenüber der Full-CI signifikant bei Dehnungsfaktoren bis 2,0. Bei größeren Dehnungen versagen die Korrekturen völlig und führen zu unphysikalischen Werten. Der Grund dafür liegt darin, dass die gewählte Referenzkonfiguration nicht mehr zur Beschreibung des Grundzustands geeignet ist. Ihr Koeffizient c_0 nimmt mit wachsendem Dehnungsfaktor ab; andere Konfigurationen hingegen nehmen an Gewicht zu.

Für die MRCI Rechnung wurde eine CAS-Raum aus sechs aktiven Orbitalen verwendet. Die Sauerstoff 1s Orbitale wurden, wie auch in der SDCI Rechnung, eingefroren. Der absolute Fehler einer MRCI ist deutlich geringer als der der SDCI, die maximale Differenz der absoluten Fehler der MRCI Energien beträgt sogar nur 2.1 mH. Auch hier lässt sich durch die Anwendung der *a posteriori* Korrekturen noch eine Verbesserung erzielen: Δ reduziert sich auf 0,4–0,7 mH, je nach Art der Korrektur.

Eine zweite Familie von Näherungen versucht den Effekt höhere Anregungen in einem effektiven Hamiltonoperator zu berücksichtigen. In den in unserem Programm implementierten Methoden MR-ACPF [18] [19] und $(SC)^2$ [20] [21] werden die diagonalen Elemente des Hamiltonoperators durch sog. *diagonale Dressings* modifiziert. Zunächst wird dazu der Full-CI Raum in drei Subräume unterteilt. Der P Raum enthält die Referenzkonfigurationen ϕ_i^P , der Q Raum enthält alle Ein- und Zweifachanregungen ϕ_i^Q der Referenzen, und der R Raum enthält alle höheren Anregungen ϕ_i^R .

In der MR-ACPF-Methode bestimmt man die Korrelationsenergie durch folgendes Funktional:

$$E_c[\Psi_Q] = \frac{\langle \Psi_P + \Psi_Q | \hat{H} | \Psi_P + \Psi_Q \rangle}{\langle \Psi_P | \Psi_P \rangle + (2/n) \langle \Psi_Q | \Psi_Q \rangle} \quad (2.41)$$

Dabei bezeichnet $\Psi_P = \sum_i c_i^P \phi_i^P$ die Referenzwellenfunktion und $\Psi_Q = \sum_i c_i^Q \phi_i^Q$ die Korrelationswellenfunktion. Dadurch dass im Nenner die Norm der Korrelationswellenfunktion mit einem Faktor $\sim 1/n$ versehen wurde, ist Extensivität sichergestellt, falls auch die Energie E_0 des Referenzraums extensiv ist. Das gilt im Fall einer einzelnen Hartree-Fock Referenzkonfiguration aber auch für CAS-artige Referenzräume.

Variation des MR-ACPF Funktionals (2.41) nach den Konfigurationen ϕ_i führt auf

$$\langle \phi_i | \hat{H} - E + K_i | \Psi_P + \Psi_Q \rangle = 0 \quad \text{für alle } i. \quad (2.42)$$

Hierbei ist $E = E_0 + E_c$ die Gesamtenergie des Systems und K_i definiert durch

$$K_i = \begin{cases} 0 & \text{für } \phi_i \in P \\ \frac{n-2}{n} & \text{für } \phi_i \in Q \end{cases} \quad (2.43)$$

Gleichung (2.42) entspricht der Säkulargleichung der MRCI, nur dass hier der Hamiltonoperator durch einen effektiven Hamiltonoperator \hat{H}^{eff} zu ersetzen ist, dessen diagonale Elemente durch ein *diagonales Dressing* K_i modifiziert wurden:

$$\hat{H}_{ij}^{eff} = \hat{H}_{ij} + K_i \delta_{ij} \quad (2.44)$$

Um die MR-ACPF Methode zu implementieren, sind daher kaum Änderungen am Programmcode notwendig.

In der $(SC)^2$ -Methode wird versucht, dem Effekt der höheren Anregungen in etwas genauerer Weise Rechnung zu tragen. Man betrachte dazu zunächst den Erwartungswert der FCI Korrelationsenergie, der sich schreiben lässt als

$$E_c = \frac{\langle \Psi_P + \Psi_Q + \Psi_R | \hat{H} | \Psi_P + \Psi_Q + \Psi_R \rangle}{\langle \Psi_P + \Psi_Q + \Psi_R | \Psi_P + \Psi_Q + \Psi_R \rangle}, \quad (2.45)$$

mit $\Psi_P = \sum_i c_i^P \phi_i^P$, $\Psi_Q = \sum_i c_i^Q \phi_i^Q$ und $\Psi_R = \sum_i c_i^R \phi_i^R$.

Eine Variation der Koeffizienten der verschiedenen Konfigurationen führt auf die folgenden CI-Gleichungen:

$$\langle \phi_i | \hat{H} - E | \Psi_P + \Psi_Q + \Psi_R \rangle = 0 \quad \text{für alle } i, \quad (2.46)$$

mit $\phi_i \in \{P, Q, R\}$ und $E = E_0 + E_C$. In praktischen Anwendungen ist das obige FCI-Problem jedoch nicht lösbar und wird in der Regel nur für $\phi_i \in \{P, Q\}$ betrachtet. Es wird nun versucht, die Koeffizienten c_i^R der höheren Anregungen durch die bekannten Koeffizienten c_i^Q und c_i^P auszudrücken. Definiert man eine Konstante K_i durch

$$K_i = \frac{\langle \phi_i | \hat{H} | \Psi_R \rangle}{\langle \phi_i | \Psi_P + \Psi_Q + \Psi_R \rangle} = \frac{\langle \phi_i | \hat{H} | \Psi_R \rangle}{c_i}, \quad (2.47)$$

so lässt sich Gleichung (2.46) umschreiben zu

$$\langle \phi_i | \hat{H} - E + K_i | \Psi_P + \Psi_Q + \Psi_R \rangle = 0 \text{ für alle } i. \quad (2.48)$$

Das entspricht formal wiederum den Säkulargleichungen der MRCI mit einem effektiven Hamiltonoperator, der nur im Modellraum der MRCI wirkt und dessen Diagonalelemente durch diagonale Dressings modifiziert wurden. Wenn man die Koeffizienten der Konfigurationen aus dem R -Raum kennen würde, wäre Gleichung (2.48) immer noch exakt. Man versucht nun, die Dressingkonstante K_i zu approximieren. Das soll zunächst für den Fall einer einzigen Referenz ϕ_0 geschehen. Die Verallgemeinerung auf den Multireferenzfall geschieht später. Man betrachte noch einmal Gleichung (2.47) genauer:

$$K_i = \frac{\langle \phi_i | \hat{H} | \Psi_R \rangle}{c_i} = \frac{\sum_j c_j^R \langle \phi_i | \hat{H} | \phi_j^R \rangle}{c_i} \quad (2.49)$$

Die Matrixelemente $\langle \phi_i | \hat{H} | \Psi_R \rangle$ sind nur von Null verschieden, wenn ϕ_j^R eine Ein- oder Zweifachanregung von ϕ_i ist:

$$\phi_j^R = \hat{D}_k \phi_i = \hat{D}_k \hat{D}_i \phi_0, \quad (2.50)$$

wobei der zweifache Anregungsoperator \hat{D}_i definiert ist durch

$$\phi_i = \hat{D}_i \phi_0. \quad (2.51)$$

Die Dressingkonstante K_i lässt sich nun schreiben als

$$\begin{aligned} K_i &= c_i^{-1} \sum_{\substack{j, \\ \hat{D}_j \phi_i \neq 0}} \langle \phi_i | \hat{H} | \hat{D}_j \phi_i \rangle \\ &= c_i^{-1} \sum_{\substack{j, \\ \hat{D}_j \phi_i \neq 0}} \langle \phi_0 | \hat{H} | \phi_j \rangle \\ &= c_i^{-1} \sum_{\substack{j, \\ \hat{D}_j \phi_i \neq 0}} H_{0j}. \end{aligned} \quad (2.52)$$

Die Summation läuft über die Menge der Doppelanregungsoperatoren \hat{D}_j , die sich tatsächlich auf ϕ_i anwenden lassen. Das sind diejenigen Operatoren, welche das Ausschlussprinzip nicht verletzen, also weder Orbitale erzeugen, die bereits in ϕ_i vorhanden sind, oder Orbitale vernichten, die nicht in ϕ_i vorhanden sind.

Wenn man nun die höheren Anregungen nicht explizit behandeln will, muss man die Koeffizienten c_j^R durch Koeffizienten von Konfigurationen des Q -Raums ausdrücken. Dazu kann man die *Clusterbedingung*

$$c_j^R = c_i c_k \quad (2.53)$$

benutzen, die in der Regel eine gute Näherung ist. Gleichung (2.53) vereinfacht sich damit zu

$$K_i = \sum_{j, \hat{D}_j \phi_i \neq 0} c_j H_{0j}. \quad (2.54)$$

In intermediärer Normierung ($\langle \phi_0 | \Psi \rangle = 1$) lässt sich die Korrelationsenergie schreiben als

$$E_c = \sum_i c_i H_{0i}, \quad (2.55)$$

wobei die Summation über alle Ein- und Zweifachanregungen von ϕ_0 läuft. Gleichung (2.54) lässt sich dann umschreiben zu

$$\begin{aligned} K_i &= E_c - \sum_{j, \hat{D}_j \phi_i \neq 0} c_j H_{0j} \\ &= E_c - E_i^{\text{EPV}}. \end{aligned} \quad (2.56)$$

E_i^{EPV} bezeichnet dabei den Beitrag, bei dem das Ausschlussprinzip verletzt wird (*exclusion principle violating*).

Zur Verallgemeinerung auf den Multireferenzfall ist der Summation in Gleichung (2.57) eine weitere Beschränkung aufzuerlegen:

$$\begin{aligned} K_i &= E_c - \sum_{\substack{j, \hat{D}_j \phi_i \neq 0 \\ \hat{D}_j \phi_i \notin \mathcal{P} \cup \mathcal{Q}}} c_j H_{0j} \\ &= E_c - E_i^{\text{EPV}} - R_i. \end{aligned} \quad (2.57)$$

Da durch den Multireferenzcharakter der Wellenfunktion bereits höhere Anregungen der Hartree-Fock Konfiguration ϕ_0 in der Wellenfunktion enthalten sind, führen die Doppelanregungsoperatoren \hat{D}_i nicht in jedem Fall aus dem Modellraum $\mathcal{P} \cup \mathcal{Q}$ heraus. R_i bezeichnet den Beitrag, der zwar nicht das Ausschlussprinzip verletzt, jedoch auch nicht aus dem MRCI Modellraum herausführt.

Zu beachten ist, dass nun zwar die Wellenfunktion Multireferenz-, das Dressing jedoch weiterhin nur Einreferenzcharakter hat. In die Berechnung der Dressingkonstante K_i gehen nur die Hartree-Fock Konfiguration und ihre Ein- und Zweifachanregungen ein. Eine Verallgemeinerung auf ein Multireferenz-Dressing (MR(SC)² CI) findet man bei Malrieu *et al.* [22] [23]. Die MR(SC)²-Methode ist jedoch nicht in unserem Programm implementiert.

In der (SC)²-Methode wird Gleichung (2.58) ohne weitere Näherungen zur Berechnung der Dressingkonstante K_i benutzt. Da K_i nun abhängig von den Koeffizienten c_i ist, wird

ein iteratives Verfahren notwendig. Aus der MRCI-Wellenfunktion werden die Dressingkonstanten K_i berechnet. Der so erhaltene effektive Hamiltonoperator wird diagonalisiert, was einen neuen Satz von Koeffizienten c_i und Dressingkonstanten K_i liefert. Das Verfahren wird wiederholt bis Selbstkonsistenz erreicht ist — üblicherweise nach 2–3 Iterationen.

Der in unserem Programm verwendete Algorithmus zur Berechnung der Dressings geht auf Ben Amor und Maynau zurück. Durch das Speichern partieller Summen lassen sich die Dressingkonstanten effizient berechnen, so dass die CPU-Kosten die einer MRCI nicht wesentlich übersteigen. Ausgehend von Gleichung (2.58) kann man die Dressingkonstanten K_i folgendermaßen umschreiben, indem man den ersten und dritten Term zusammenfasst:

$$K_i = \sum_{j, \hat{D}_j \phi_i \notin \mathcal{P} \cup \mathcal{Q}} H_{0j} c_j - \sum_{j, \hat{D}_j \phi_i = 0} H_{0j} c_j. \quad (2.58)$$

Der erste Term ist einfach zu erhalten: Sei ϕ_i eine Konfiguration mit n_h Löchern, also fehlenden inaktiven Orbitalen, und mit n_p Teilchen, also externen Orbitalen ($n_h \leq 2, n_p \leq 2$). Die Doppelanregungen D_j , für die

$$D_j \phi_i \notin \mathcal{P} \cup \mathcal{Q} \quad (2.59)$$

gilt, werden die Anregungen mit e_h Löchern und e_p Teilchen sein, mit

$$n_h + e_h > 2 \quad \text{oder} \quad n_p + e_p > 2. \quad (2.60)$$

Definiert man nun

$$S_h^p = \sum_{j, \hat{D}_j(h,p)} H_{0j} c_j \quad (2.61)$$

als Summe von $H_{0j} c_j$ über alle Doppelanregungen mit h Löchern und p Teilchen, kann man für eine Konfiguration ϕ_i mit n_h Löchern und n_p Teilchen

$$\sum_{j, \hat{D}_j \phi_i \notin \mathcal{P} \cup \mathcal{Q}} H_{0j} c_j = \sum_{\substack{h,p \\ n_p+p > 2 \text{ oder } n_h+h > 2}} S_h^p \quad (2.62)$$

schreiben. Von den durch Gleichung (2.61) definierten Summen S_h^p gibt es acht Stück, da sowohl h als auch p die Werte 0, 1 und 2 annehmen kann. Diese Summen können vor der Iteration des Zustandsvektors berechnet werden, indem alle Konfigurationen mit ihren Koeffizienten einmal eingelesen werden. Es handelt sich hierbei um einen Prozess, dessen Aufwand mit der Größe des Zustandsvektors, also mit $n_e^2 N^2$ skaliert.

Die EPV-Terme aus Gleichung (2.58) sind schwieriger zu erhalten. Indem alle Konfigurationen einer Summe S_h^p nach Gleichung (2.62) zur Dressingkonstanten K_i addiert wurden, wurden einige Terme $H_{0j} c_j$ addiert, in denen i und j gemeinsame Orbitalindizes haben. Um diesen Fehler zu korrigieren, ist es notwendig alle $H_{0j} c_j$ zu entfernen, in denen j einen Orbitalindex von i enthält. Handelt es sich beispielsweise bei der Konfiguration ϕ_i um die Doppelanregung $(a, b \rightarrow r, s)$, so muss die Summe S_h^p folgendermaßen korrigiert werden:

$$S_h^p - S_h^p(a) - S_h^p(b) - S_h^p(r) - S_h^p(s). \quad (2.63)$$

Die Summe $S_h^p(x)$ ist hierbei genauso definiert wie S_h^p in Gleichung (2.61), außer dass die Summation hierbei nur über die Konfigurationen läuft, deren erzeugender Doppelanregungsoperator \hat{D}_j den Orbitalindex x beinhaltet. Die Summen $S_h^p(x)$ können zeitgleich mit den Summen S_h^p im selben $N^2 n_e^2$ Prozess berechnet werden.

Indem man die Korrektur (2.63) durchführt, entsteht ein neuer Fehler, da beispielsweise die Konfiguration ϕ_k , bei der \hat{D}_k die Indizes r und s enthält, zweimal abgezogen wird. Um diesen Fehler zu korrigieren, ist es notwendig Zweiindexsummen $S_h^p(x, y)$ zu definieren. Die Korrektur für die Konfiguration $\phi_i = (a, b \rightarrow r, s)$ nimmt dann folgende Form an:

$$S_h^p - \sum_x S_h^p(x) + \sum_{x,y} S_h^p(x, y), \quad (2.64)$$

wobei x und y über die Indizes des Operators \hat{D}_i , also a, b, r und s , läuft. Auch dieser Ausdruck ist fehlerhaft, da nun der Beitrag einer Konfiguration, die drei Indizes, z.B. a, b und r , enthält, zweimal addiert wird. Berücksichtigt man alle möglichen Korrekturen, so erhält man

$$S_h^p - \sum_x S_h^p(x) + \sum_{x,y} S_h^p(x, y) - \sum_{x,y,z} S_h^p(x, y, z) + \sum_{x,y,z,t} S_h^p(x, y, z, t). \quad (2.65)$$

Alle Summen S_h^p werden vor der Iteration des Zustandsvektors im selben $N^2 n_e^2$ Prozess berechnet. Sie müssen während des Iterationsschritts im Hauptspeicher gehalten werden. Das ist jedoch auch bei der Vierindexsumme $S_h^p(x, y, z, t)$ möglich, da in diesem Fall einerseits die Indizes x und y nur über die (zu vernichtenden) *internen* Orbitale laufen und andererseits die Indizes z und t nur über die (zu erzeugenden) externen Orbitale laufen.

Strenggenommen gilt der vorgestellte Algorithmus von Ben Amor und Maynau nur für CAS-artige Referenzräume. Andernfalls gibt es auch Anregungen mit weniger als zwei Teilchen und Löchern, die nicht mehr im Modellraum enthalten sind; Gleichung (2.62) für den ersten Term in Gleichung (2.58) gilt dann nicht mehr. Aber auch bei einer Referenzselektion ist man in der Regel nahe am CAS-Fall: Die nicht selektierten Referenzen und ihre Anregungen sollten auch in der Wellenfunktion der CAS-CI kaum ins Gewicht fallen.

Analog zu den in Tabelle 2.5 dargestellten Ergebnissen, haben wir Rechnungen zur symmetrischen Dehnung der O–H Bindung in Wasser mit der MR-ACPF und der (SC)² Methode durchgeführt, deren Ergebnisse in Tabelle 2.6 aufgeführt sind. Wiederum reicht eine Referenz nicht zur Beschreibung der Korrelationseffekte aus: Besteht der Referenzraum aus nur einer Referenz, divergieren die MR-ACPF und (SC)² Ergebnisse für Dehnungsfaktoren größer als 2,0. In den CAS-Rechnungen hingegen lassen sich durch die beiden *a priori* Korrekturen signifikante Verbesserungen erzielen: Δ sinkt von 2,1 mH für die MRCI auf 0.4 bzw. 0.5 mH für die MR-ACPF bzw. (SC)² Methode.

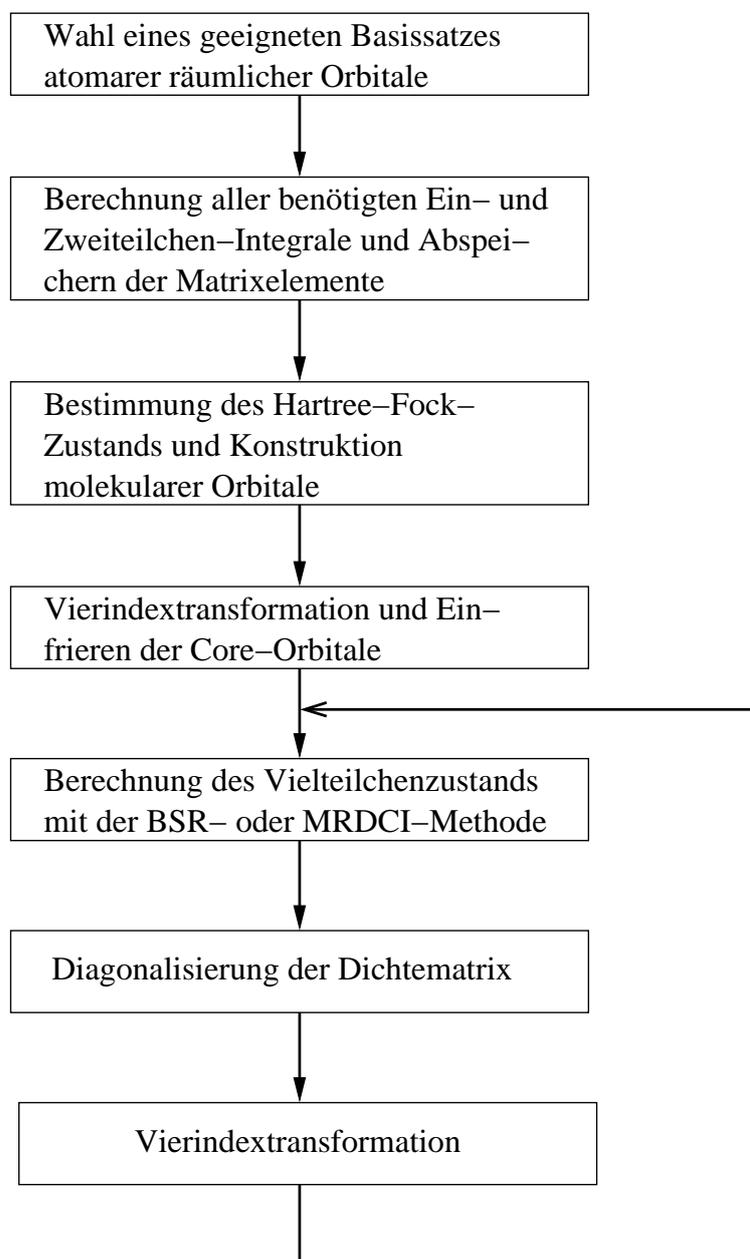


Abbildung 2.7: Vor Beginn der numerischen Rechnungen muss zunächst ein molekülspezifischer Basissatz vorgegeben werden, für den dann die erforderlichen Integrale berechnet werden. Damit wird stets eine Hartree-Fock Rechnung durchgeführt, auf der dann die MRD-CI- oder BSR-Methode aufbaut. Um natürliche Orbitale zu generieren, wird die Korrelationsrechnung iterativ durchgeführt und nach jedem Schritt die berechnete Dichtematrix diagonalisiert.

r_{OH}	1.0	1.5	2.0	2.5	3.0	Δ
full CI	-76.24186	-76.07234	-75.95166	-75.91798	-75.91194	0
SDCI	14.72	32.66	72.90	146.36	136.34	131.6
MR-ACPF	6.57	10.32	—	—	—	10.3
(SC) ²	8.67	18.17	—	—	—	18.2
MRCI	6.64	5.83	5.04	4.65	4.56	2.1
MR-ACPF	1.93	1.51	1.71	1.77	1.78	0.4
(SC) ²	3.98	3.57	3.48	3.72	3.68	0.5

Tabelle 2.6: (SC)² und MR-ACPF Rechnungen zur symmetrischen Dehnung der O–H Bindung in Wasser. r_{OH} bezeichnet den Dehnungsfaktor der Gleichgewichtsbindungslänge $r_0 = 0.975512\text{\AA}$. Der Bindungswinkel betrug $110,6^\circ$, die Rechnungen wurden in einer cc-pVDZ Basis von Dunning durchgeführt. Die FCI Ergebnisse stammen von Olsen *et al.* [16]. Die FCI Energien sind in H, die übrigen Energien in mH relativ zu den FCI Energien angegeben. Δ ist die maximale Differenz der absoluten Fehler.

2.6 Ablauf quantenchemischer Rechnungen

Der Weg zur Berechnung von Energien mit unserem Quantenchemie-Programmpaket ist grob in Abb. 2.7 skizziert. Im einzelnen sind folgende Schritte notwendig:

- Für jedes Atom, aus dem das zu berechnende Molekül besteht, werden zunächst geeignete atomare räumliche Orbitale $\psi_i(\vec{r})$ ausgewählt. Jedes Orbital stellt sich als eine Linearkombination von Gaußfunktionen dar. Die Koeffizienten und Exponenten dieser Linearkombinationen sind z.B. im Internet [] erhältlich. Die Anzahl der Orbitale pro Atom ist notgedrungen endlich. Die Orbitale bilden in ihrer Gesamtheit eine nichtorthogonale Basis für das Molekül.
- Für die Gleichgewichtsgeometrie des Moleküls werden dann alle benötigten Integrale zwischen sämtlichen Orbitalen berechnet und, auf die Knoten verteilt, im Hauptspeicher des Parallelrechners gehalten. Im einzelnen sind dies die Überlappintegrale

$$S_{ij} = \int d\vec{r}_1 \psi_i^*(\vec{r}_1) \psi_j(\vec{r}_1), \quad (2.66)$$

die Integrale der kinetischen Energie

$$T_{ij} = \int d\vec{r}_1 \psi_i^*(\vec{r}_1) \left[-\frac{1}{2} \nabla_{\vec{r}_1}^2 \right] \psi_j(\vec{r}_1), \quad (2.67)$$

der Kernanziehung

$$V_{ij}^{nucl} = \int d\vec{r}_1 \psi_i^*(\vec{r}_1) \left[-\sum_A \frac{Z_A}{|\vec{r}_1 - \vec{R}_A|} \right] \psi_j(\vec{r}_1), \quad (2.68)$$

die aufwändigen Integrale der Zweielektronwechselwirkung

$$V_{ijkl} = \int d\vec{r}_1 d\vec{r}_2 \psi_i^*(\vec{r}_1) \psi_j(\vec{r}_1) \frac{1}{|\vec{r}_1 - \vec{r}_2|} \psi_k^*(\vec{r}_2) \psi_l(\vec{r}_2) \quad (2.69)$$

und, falls man an der Berechnung von Erwartungswerten von Multipoloperatoren interessiert ist, die Multipolintegrale

$$M_{ij}(\vec{r}, \vec{n}) = \int d\vec{r}_1 \psi_i^*(\vec{r}_1) [r_x^{n_x} r_y^{n_y} r_z^{n_z}] \psi_j(\vec{r}_1). \quad (2.70)$$

Die Multipolintegrale sind zusätzlich durch den Vektor \vec{n} indiziert. So müssen z.B. für die drei Komponenten des Dipolmoments drei separate Matrizen für $\vec{n} = (1,0,0)$, $(0,1,0)$, $(0,0,1)$ berechnet werden. Bei der Berechnung nach Gleichung (2.70) wurde der Koordinatenursprung stets in den Schwerpunkt des Moleküls gelegt.

Außerdem müssen auch die konstanten Beiträge der Atomkerne zur Energie und zu den Multipolmomenten berechnet werden. Dies alles geschieht mit dem Programm IPACK (Integral Package).

- Unter Verwendung von S_{ij} , $H_{ij}^{\text{core}} = T_{ij} + V_{ij}^{\text{nucl}}$ und V_{ijkl} werden dann mit dem SCF-Programm der Hartree-Fock Zustand des Moleküls und die zugehörigen (orthonormalen) molekularen Orbitale bestimmt.
- Die umfangreiche Transformation der Integralmatrizen in die Basis der molekularen Orbitale vollzieht das Programm FOUR. Der aufwendigste Schritt ist dabei die 'Vierindextransformation' der Zweielektronintegrale. Wenn N die Anzahl der Basisfunktionen ist, so skaliert nämlich die Anzahl der Zweielektronintegrale ungefähr mit $N^4/8$, wogegen die Anzahl der Einteilchenintegrale nur mit N^2 geht. Insgesamt skaliert diese Transformation im Aufwand ca. mit N^5 . Es ist hierbei von großem Vorteil, dass die Integrale im Hauptspeicher gehalten werden können und nicht auf die Festplatte zugegriffen werden muss. In der seriellen Version des Programms ist die Vierindextransformation der limitierende Faktor für die Größe der Basissätze.
- Die chemischen Eigenschaften des Moleküls werden hauptsächlich durch die Valenzorbitale bestimmt. Die inneren, vollbesetzten Schalen (Core-Orbitale) spielen dabei eine vernachlässigbare Rolle. Deshalb kann man die Anzahl der Freiheitsgrade dadurch reduzieren, dass man die Core-Orbitale einfriert, d.h. sie stets als vollbesetzt annimmt und sie daher aus weiteren Korrelationsrechnungen herausnimmt. Diese Separation wird ebenfalls vom Programm FOUR ausgeführt.
- An die Hartree-Fock Rechnung schließen sich weiterführende Korrelationsrechnung mit dem MRD-CI- bzw. BSR-Programm an. Hierfür bilden zunächst die molekularen Orbitale der SCF die Ausgangsbasis. Aus ihnen kann in Form von Slaterdeterminanten die Basis des Vielteilchenhilbertraums konstruiert werden. Mittels der Korrelationsrechnungen wird ein verbesserter Vielteilchengrundzustand bestimmt.

- Mit der MRD-CI- oder BSR-Dichtematrix können nun auch natürliche Orbitale generiert werden. Dazu wird zunächst mit dem Programm VIRTUAL die Dichtematrix diagonalisiert. Sind mehrere Dichtematrizen für unterschiedliche elektronische Zustände vorhanden, so werden diese gemittelt. Anschließend erfolgt wieder eine Vierindextransformation (FOUR) sowie eine neue MRD-CI oder BSR.

Kapitel 3

Der massiv parallele MRCI-Algorithmus

3.1 Der Residuenbaum

Zur Berechnung der Matrixelemente des Hamiltonoperators zwischen zwei Konfigurationen ϕ_i und ϕ_j haben wir ein Ordnungsschema entwickelt, das von der Menge der Orbitale abhängt, die in ϕ_i und ϕ_j identisch sind. Diese Teilmenge von ϕ_i und ϕ_j wird im folgenden als *Übergangsresiduum* oder einfach *Residuum* bezeichnet.

Um das Ordnungsschema zu illustrieren, betrachtet man den auf die Zweiteilchenmatrixelemente beschränkten Hamiltonoperator

$$H = \sum_{\alpha\beta\gamma\delta} V_{\alpha\beta\gamma\delta} c_{\alpha}^{\dagger} c_{\beta}^{\dagger} c_{\gamma} c_{\delta}. \quad (3.1)$$

Es handelt sich hierbei nicht um eine Einschränkung, da die Einteilchenmatrixelemente völlig analog zu den Zweiteilchenmatrixelementen erfolgen kann. Der Hauptrechenaufwand geht jedoch in die Berechnung der Zweiteilchenmatrixelemente, denen daher hier größere Beachtung geschenkt wird. Offensichtlich können die Matrixelemente des Hamiltonoperators aus Gleichung (3.1) nur von Null verschieden sein, wenn sich die Konfigurationen ϕ_i und ϕ_j um höchstens zwei Orbitale unterscheiden.

Der kanonische Weg zur Berechnung der Matrixelemente ist in Abb. 3.1 dargestellt: Ausgehend von einer Konfiguration ϕ_i vernichtet man zunächst zwei Orbitale o_1 und o_2 und erhält so ein Residuum R . Fügt man zu R wieder zwei Orbitale o_3 und o_4 hinzu, so erhält man eine Konfiguration ϕ_j :

$$\Phi_i \xrightarrow{-o_1, o_2} R \xrightarrow{+o_3, o_4} \Phi_j. \quad (3.2)$$

Da es sich bei der hier implementierten Methode um die zustandsselektierende Variante der MRCI handelt, ist es nun nicht klar, ob die Konfiguration ϕ_j überhaupt im selektierten

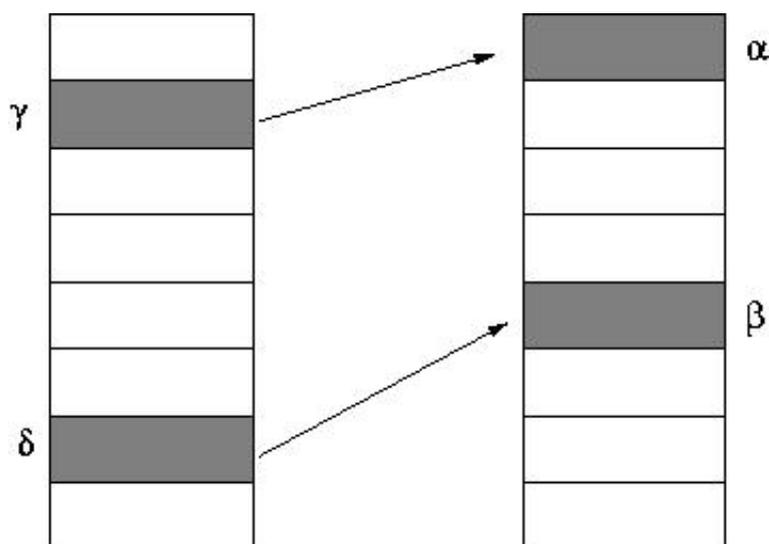


Abbildung 3.1: Herkömmliche Berechnung der Matrixelemente. In der Konfiguration ϕ_i (links) werden die Orbitale γ und δ vernichtet und die Orbitale α und β erzeugt, wodurch man eine Konfiguration ϕ_j erhält. Diese ist nicht notwendig im selektierten Subraum der MRD-CI enthalten.

Subraum vorhanden ist. In praktischen Rechnungen werden ca. 1% der Konfigurationen für die variationelle Rechnung selektiert, so dass die Konfiguration ϕ_j in 99% der Fälle *nicht* im Zustandsvektor enthalten ist. Der kanonische Weg, um die Matrixelemente zu berechnen, ist damit sehr ineffektiv.

Eine sehr effiziente Berechnung der Matrixelemente wird möglich, wenn man einen Baum aller Übergangsresiduen, die in einem gegebenen Zustandsvektor vorkommen, betrachtet. In Abb. 3.2 ist der Aufbau des Residuenbaums schematisch dargestellt. Für jedes Residuum R wird eine Liste von Residueneinträgen gespeichert. Diese bestehen aus dem Orbitalpaar (oder dem Orbital für ein Einteilchenresiduum), das zusammen mit dem Residuum R eine Konfiguration ϕ_i aus dem Zustandsvektor ergibt und dem Index der Konfiguration ϕ_i im Zustandsvektor. Weiterhin enthält der Residueneintrag noch Information über notwendige Permutationen in Form eines Vorzeichens: Wenn man ein Orbital o_1 aus der Konfiguration ϕ_i vernichten möchte, muss das Orbital o_1 dazu zunächst an die erste Position in der Konfiguration getauscht werden. Um das zu erreichen ohne die Ordnung der Konfiguration zu zerstören, sind — wenn sich das Orbital o_1 innerhalb der Konfiguration ϕ_i an Position $\text{pos}(o_1, \phi_i)$ befindet, $\text{pos}(o_1, \phi_i)$ paarweise Vertauschungen notwendig. Die Slater-Determinante ϕ_i ist dann mit einem Vorzeichen

$$\text{sign}(o_1, \phi_i) = (-1)^{\text{pos}(o_1, \phi_i)} \quad (3.3)$$

zu versehen. Vernichtet man in der Konfiguration ϕ_i zunächst o_1 und dann o_2 , also

$$\Phi_i \xrightarrow{-o_1} R_1 \xrightarrow{-o_2} R_2, \quad (3.4)$$

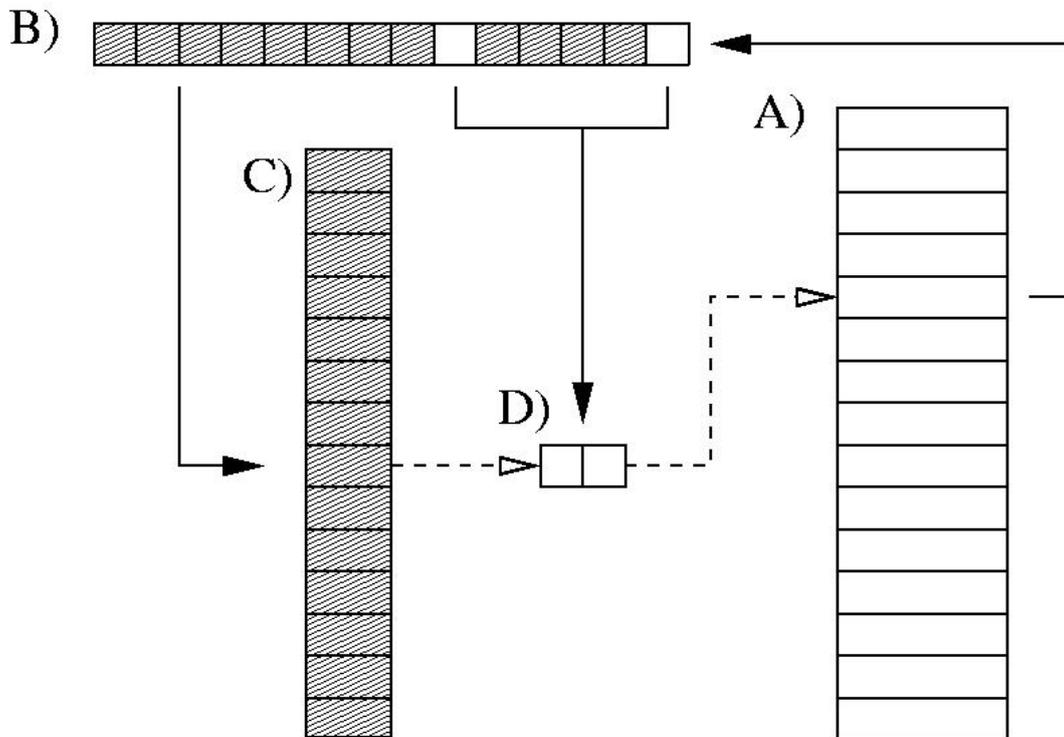


Abbildung 3.2: Konstruktion des Residuenbaums. Für jede Konfiguration ϕ_i aus dem Zustandsvektor (A) werden alle möglichen Zweiteilchenresiduen gebildet. Die Konfiguration ϕ_i ist in (B) illustriert: Jeder Kasten repräsentiert ein besetztes Orbital, der dunkel gezeichnete Bereich ist das Residuum R , das man erhält, wenn man die beiden hell dargestellten Orbitale o_1 und o_2 vernichtet. Das $(n_e - 2)$ Elektronen große Residuum R wird im Residuenbaum (C) gesucht. Dort wird ein Element (D) hinzugefügt, das die beiden vernichteten Orbitale, Informationen über die Permutationen (also die Positionen von o_1 und o_2 in ϕ_i) und den Index der ursprünglichen Konfiguration ϕ_i im Zustandsvektor (A) enthält. Die durchgezogenen Linien stellen logische Beziehungen da. Die gestrichelten Linien Korrespondieren mit Zeigern (Pointern) in der Datenstruktur. Der Residuenbaum muss nach jeder Vergrößerung des Zustandsvektors neu gebildet werden. Der Aufwand dazu ist proportional zum Produkt von n_e^2 mit der Anzahl der Konfigurationen. Die Anzahl der Matrixelemente, die durch ein Residuum R kodiert werden, ist proportional zum Quadrat der Einträge vom Typ (D), die zum Residuum R gehören.

so erhält man insgesamt ein Vorzeichen

$$\text{sign}(o_1, o_2, \phi_i) = \text{sign}(o_1, \phi_i) \cdot \text{sign}(o_2, R_1). \quad (3.5)$$

Falls sich die Position von o_2 durch das Vernichten von o_1 nicht verändert, gilt:

$$\text{sign}(o_1, o_2, \phi_i) = \text{sign}(o_1, \phi_i) \cdot \text{sign}(o_2, \phi_i). \quad (3.6)$$

Das Vorzeichen $\text{sign}(o_1, o_2, \phi_i)$ wird mit dem Orbitalpaar und dem Index der ursprünglichen Konfiguration ϕ_i im Residueneintrag abgespeichert.

Auch wenn die Anzahl der Residuen relativ klein ist, wächst die Anzahl der Residuen-einträge schnell mit $N_{\text{sel}} n_e^2$, wobei N_{sel} die Größe des Zustandsvektors und n_e die Anzahl der Elektronen ist. In einer nichtselektierenden MRCI ist es daher nicht möglich, den Residuenbaum während des ganzen Programmablaufs im Speicher zu halten. Es ist hier vielmehr nötig, Teile des Residuenbaums temporär zu konstruieren und nach Gebrauch wieder zu löschen [24]. Für die zustandsselektierende MRCI hingegen ist es durch die reduzierte Anzahl von Konfigurationen im Zustandsvektor in Verbindung mit dem großen (verteilten) Hauptspeicher moderner Parallelrechner möglich, den Residuenbaum für die selektierten Konfigurationen zu bilden. Voraussetzung dafür ist jedoch, dass es möglich ist, den Residuenbaum auf die Knoten des Parallelrechners zu verteilen.

Der Programmablauf gliedert sich, wie bereits in Abschnitt 2.1 beschrieben, im wesentlichen in zwei Teile: Im *Expansionsschritt* wird der Hilbertraum der Ein- und Zweifachanregungen durchsucht. Für jede Testkonfiguration ϕ_t ist die Wechselwirkung

$$q_t = \langle \phi_t | \hat{H} | \Psi_0 \rangle \quad (3.7)$$

mit der Referenzwellenfunktion Ψ_0 zu berechnen. Die Berechnung der Matrixelemente im Expansionsschritt mit dem Residuenbaum ist in Abb. 3.3 schematisch dargestellt: In der Testkonfiguration ϕ_t wird jedes enthaltene Elektronenpaar vernichtet. Dadurch erhält man jeweils ein Residuum R und ein Orbitalpaar o_1, o_2 . Das so erhaltene Residuum R wird im Residuenbaum gesucht; jedes Orbitalpaar o_3, o_4 , das mit R verbunden ist, liefert einen Beitrag

$$V_{o_1 o_2 o_3 o_4} \text{sign}(o_1, o_2, \phi_t) \text{sign}(o_3, o_4, \phi_i) c_i. \quad (3.8)$$

c_i ist hierbei der Koeffizient der Konfiguration ϕ_i , deren Index im entsprechenden Residuen-eintrag gespeichert ist. Die Anzahl nichttrivialer Operationen pro Testkonfiguration ist proportional zu n_e^2 , der gesamte numerische Aufwand im Expansionsschritt skaliert linear mit der Größe des Hilbertraums.

Der im Expansionsschritt benötigte Residuenbaum wird aus den Referenzkonfigurationen konstruiert. Er muss auf allen Knoten in identischer Form vorhanden sein, da in einer zu untersuchenden Testkonfiguration ϕ_t im Prinzip jedes Residuum R vorhanden sein kann. Der Residuenbaum ist aber klein, da er nur aus den wenigen Referenzen konstruiert wurde. Es ist daher kein Problem, ihn auf allen Knoten im Speicher zu halten. Nach dem

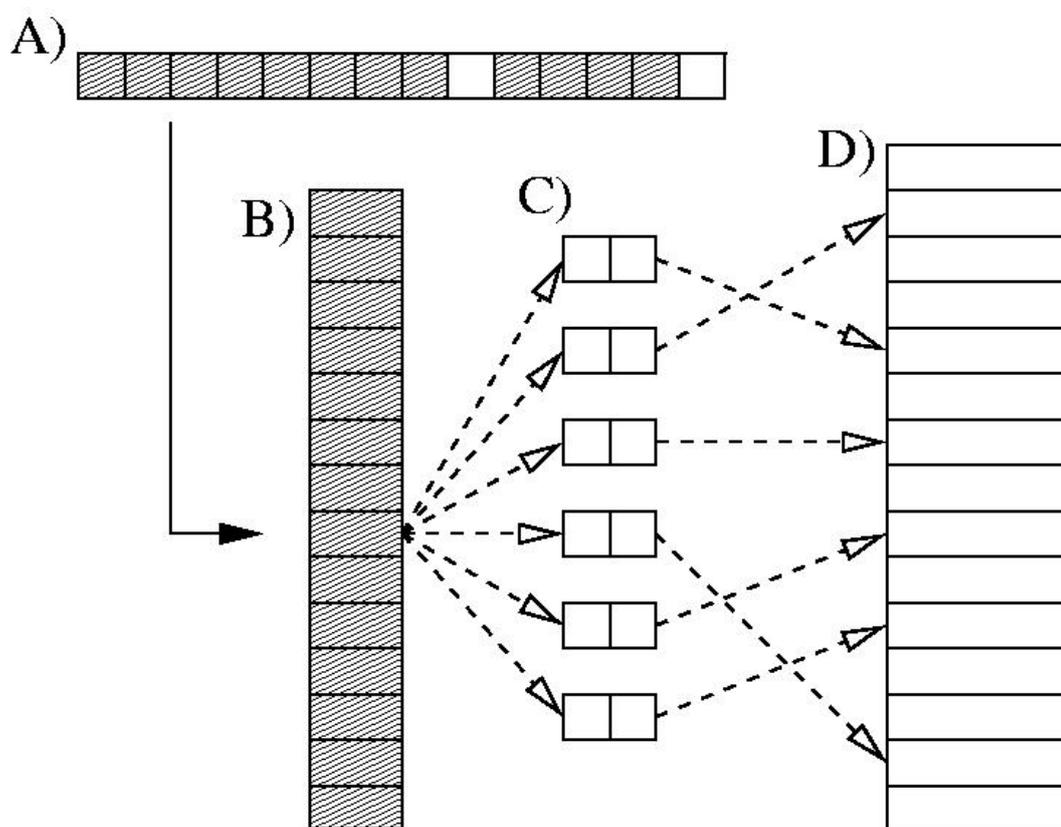


Abbildung 3.3: Schematische Darstellung der Berechnung von Zweiteilchenmatrixelementen im Expansionsschritt unter Benutzung des Residuenbaums. Aus einer gegebenen Testkonfiguration (A) werden alle möglichen Übergangsresiduen gebildet. Jedes Kästchen in der unter (A) dargestellten Konfiguration bezeichnet ein besetztes Orbital. Der dunkel gezeichnete Bereich kennzeichnet das Residuum R . Das Residuum R wird im Residuenbaum gesucht. Jeder zu R gehörende Residuenbeitrag (C) aus dem Residuenbaum kodiert ein Matrixelement der Testkonfiguration mit einer Konfiguration aus dem Zustandsvektor (D). Die Orbitalindizes zur Berechnung des benötigten Integrals sind im Residuenbeitrag (C) gespeichert. Dort ist ebenfalls der Index der ursprünglichen Konfiguration im Zustandsvektor (D) gespeichert, unter dem man den benötigten Koeffizienten findet. Es ist nur eine Suchoperation notwendig, um alle Matrixelemente zu berechnen, die durch ein gegebenes Residuum vermittelt werden. Auch werden nur die Matrixelemente berechnet, die zu tatsächlich vorhandenen Referenzkonfigurationen führen.

Expansionsschritt ist ein neuer Residuenbaum zu konstruieren; dieses Mal aus allen selektierten Konfigurationen. Nun kann jedoch der Residuenbaum auf alle Knoten verteilt werden: Alle zu einem bestimmten Residuum R gehörenden Matrixelemente können lokal ausgewertet werden, sofern auch alle zu R gehörenden Residueneinträge auf einem eindeutigen Knoten gesammelt werden. Der Residuenbaum selbst (Teil B in Abb. 3.3) wird im Iterationsschritt nicht benötigt; es genügt zu wissen, welche Residueneinträge miteinander verbunden sind. Das Residuum selbst kann aus dem Speicher gelöscht werden.

Im Iterationsschritt sind keine Suchoperationen notwendig. Man kann einfach Schleifen über die lokal vorhandenen Segmente von Residueneinträgen bilden. Je zwei Residueneinträge aus einem solchen Segment liefern nun Beiträge zum Vektor

$$\vec{q} = \hat{H}\vec{c}. \quad (3.9)$$

Finden sich beispielsweise in einem Segment das Orbitalpaar o_1 und o_2 , mit einem Zeiger auf die Konfiguration ϕ_i und das Orbitalpaar o_3 und o_4 mit einem Zeiger auf die Konfiguration ϕ_j , so liefern diese Paare die folgenden Beiträge:

$$\begin{aligned} \Delta q_i &= \text{sign}(o_1, o_2, \phi_i) \text{sign}(o_3, o_4, \phi_j) c_j V_{o_1 o_2 o_3 o_4} \\ \Delta q_j &= \text{sign}(o_1, o_2, \phi_i) \text{sign}(o_3, o_4, \phi_j) c_i V_{o_1 o_2 o_3 o_4} \end{aligned} \quad (3.10)$$

Nachdem alle lokalen Beiträge Δq_i errechnet wurden, müssen die Beiträge der einzelnen Knoten addiert werden, um die Davidson-Iteration (s. Abschnitt 2.2) durchzuführen.

Durch das Abspeichern von N zusammengehörigen Residueneinträgen werden N^2 Matrixelemente kodiert. Das bedeutet, dass auch die Arbeit, die die Berechnung der Matrixelemente macht, die durch diese Residueneinträge kodiert werden, proportional zu N^2 ist. Schwankt nun die Zahl der zusammengehörigen Orbitalpaare stark, so kann es vorkommen, dass einige Knoten wesentlich mehr Arbeit leisten müssen als andere. In unserer Implementierung wird das durch ein Ausgleichsverfahren gelöst: Nachdem der Residuenbaum (für den Iterationsschritt) konstruiert wurde, aber vor Beginn des Iterationsschritts, wird die Arbeit, die die einzelnen Knoten zu leisten haben vorab berechnet. Daraufhin werden die Residueneinträge umverteilt, bis alle Knoten gleich viel Arbeit zu leisten haben. Dieses Ausgleichsverfahren wird in Abschnitt 3.3 näher beschrieben.

Der limitierende Faktor für die Anzahl der Zustände im selektierten Subraum ist die Größe des Residuenbaums. Diese wächst mit dem Quadrat der Elektronenzahl (n_e^2) und ist proportional zur Größe des Zustandsvektors. Die Größe des Zustandsvektors wiederum wächst nach Gleichung (2.13) mit dem Quadrat der Größe der Basis und dem Quadrat der Elektronenzahl, so dass die Größe des Residuenbaums insgesamt mit $n_e^4 N^2$ wächst.

Es gibt verschiedene Möglichkeiten, die Anzahl der Residuen zu verringern. So haben wir alle Residuen, die mit nur einem Residueneintrag verbunden sind, aus dem Baum entfernt haben. Derartige Residuen vermitteln nur ein diagonales Matrixelement, das in unserem Programm direkt und ohne Zuhilfenahme des Residuenbaums berechnet wird. Insbesondere bei kleinen Selektionsschwellwerten ist die Anzahl derartiger Residuen jedoch äußerst

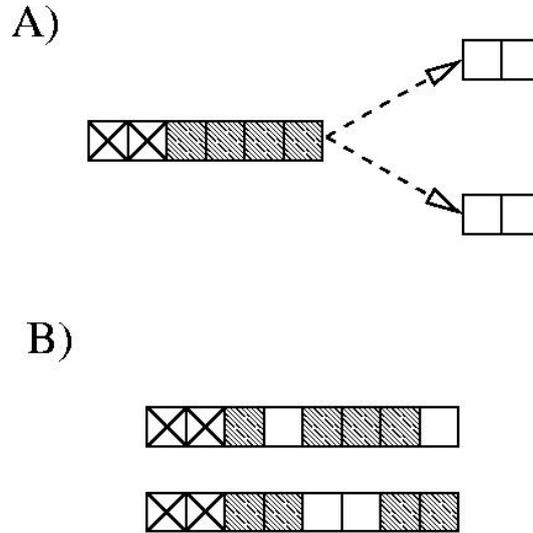


Abbildung 3.4: Sonderbehandlung der XX-Residuen. Anstatt die Residuen mit zwei externen Orbitalen in den Residuenbaum aufzunehmen, wie unter (A) gezeigt, kann man die von den XX-Residuen vermittelten Matrixelemente auch direkt berechnen. Dazu sortiert man die Konfigurationen wie unter (B) gezeigt nach ihrem externen Anteil und bildet die Differenz der internen Anteile der Konfigurationen. Das Matrixelement hat rein interne Indizes.

gering. Eine signifikante Verringerung der Größe des Residuenbaums ließ sich hierdurch nicht erzielen.

Eine deutliche Reduktion des Speicherbedarfs für den Residuenbaum haben wir dadurch erzielt, dass wir alle Residuen, die zwei externe Orbitale enthalten, aus dem Baum entfernt haben. Matrixelemente, die von solchen Residuen vermittelt werden, können berechnet werden, indem alle Konfigurationen, die ein bestimmtes Paar o_x, o_y von externen Orbitalen enthalten, auf einem eindeutigen Knoten gesammelt werden. Zwischen je zwei Konfigurationen ϕ_i und ϕ_j , die das Orbitalpaar o_x, o_y enthalten, ist nun die Differenz der internen Anteile dieser Konfigurationen zu berechnen. Dieser Schritt wird dadurch sehr effektiv, dass die Differenzen der internen Anteile sämtlicher Konfigurationen tabelliert und im Speicher gehalten werden können. Unterscheiden sich die internen Anteile von ϕ_i und ϕ_j nun um zwei Orbitale, also

$$\phi_i \xrightarrow{-o_1, o_2} R \xrightarrow{+o_3, o_4} \phi_j, \quad (3.11)$$

so gibt es nur ein Übergangsresiduum R , das das externe Orbitalpaar o_x, o_y enthält. Von diesem wird ein Matrixelement der Form

$$\begin{aligned} \Delta q_j &= \text{sign}(o_1, o_2, \phi_i) \text{sign}(o_3, o_4, \phi_j) c_i V_{o_1 o_2 o_3 o_4} \\ \Delta q_i &= \text{sign}(o_1, o_2, \phi_i) \text{sign}(o_3, o_4, \phi_j) c_j V_{o_1 o_2 o_3 o_4} \end{aligned} \quad (3.12)$$

Der Wert des Matrixelements lässt sich aus einer Tabelle aller Integrale mit rein internen Indizes berechnen, die auf allen Knoten bequem im Hauptspeicher gehalten werden kann.

Beträgt die Differenz von ϕ_i und ϕ_j nur ein Orbital, so gibt es $n_e - 3$ mögliche Übergangsr residuen R , die o_x und o_y enthalten:

$$\phi_i \xrightarrow{-o_1, o} R \xrightarrow{+o_2, o} \phi_j. \quad (3.13)$$

Das Orbital o muss hierbei die Bedingungen $o \in \phi_i$ und $o \notin \{o_1, o_2, o_x, o_y\}$ erfüllen. Die Matrixelemente, die von einem Übergang der Form (3.13) vermittelt werden, sind von der Form:

$$\begin{aligned} \Delta q_j &= \text{sign}(o_1, o, \phi_i) \text{sign}(o_2, o, \phi_j) c_i V_{o_1 o o_2 o} \\ \Delta q_i &= \text{sign}(o_1, o, \phi_i) \text{sign}(o_2, o, \phi_j) c_j V_{o_1 o o_2 o}. \end{aligned} \quad (3.14)$$

Einen weiteren speicherbestimmenden Faktor stellt der Koeffizientenvektor \vec{c} und das Feld \vec{q} aus Gleichung (3.9) dar. Die für die Davidson-Iteration notwendigen Kopien dieser Vektoren konnten wir auf die Knoten verteilen, so dass nur jeweils eine Kopie von \vec{c} und \vec{q} unverteilt auf allen Knoten im Hauptspeicher gehalten werden muss. Geht man von einer 16 Bit Darstellung der Koeffizienten aus, erfordert das bei 50 Millionen selektierten Konfigurationen insgesamt 400 MB des Hauptspeichers. Es handelt sich hierbei um die größte nicht verteilte Datenstruktur des Programms.

Es ist möglich, auch die Vektoren \vec{c} und \vec{q} zu verteilen, indem man innerhalb eines Residuenintrags nicht nur den Index der ursprünglichen Konfiguration ϕ_i innerhalb des Zustandsvektors abspeichert, sondern auch Speicherplatz für die entsprechende Komponenten c_i des Vektors \vec{c} und q_i des Feldes \vec{q} vorsieht und somit den Residuenbaum lokal vergrößert. Dieses Verfahren ist in der aktuellen Version des Programms jedoch nicht implementiert.

Ebenfalls speicherbestimmend sind die Zweiteilchenintegrale. Deren Anzahl wächst mit der Größe der Basis in vierter Potenz; die Integrale können daher bei größeren Molekülen und größeren Basissätzen nicht mehr komplett im Speicher gehalten werden. Beispielsweise beanspruchen die Integrale bei einer *triple zeta*-Rechnung an Benzol ca. 2,1 GB. Der Speicherbedarf für die Integrale konnte drastisch reduziert werden, indem die Integrale auf die einzelnen Knoten verteilt wurden. Mit den Details des integralgetriebenen Algorithmus befasst sich der nächste Abschnitt.

3.2 Integralgetriebener Algorithmus

Neben dem Residuenbaum, dem Koeffizientenvektor \vec{c} und dem Feld \vec{q} sind die Zweiteilchenintegrale ein weiterer speicherbestimmender Faktor. Diese sind — setzt man rein reelle Orbitale o_i voraus — in der so genannten *Physikernotation* von der Form

$$\langle o_1 o_2 | o_3 o_4 \rangle = \int d^3 \vec{r} d^3 \vec{r}' o_1(\vec{r}) o_2(\vec{r}') \frac{1}{|\vec{r} - \vec{r}'|} o_3(\vec{r}) o_4(\vec{r}') \quad (3.15)$$

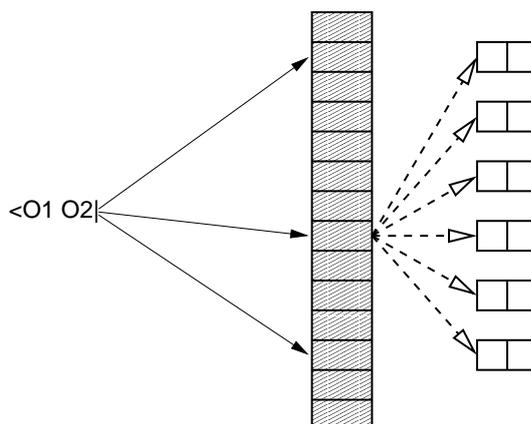


Abbildung 3.5: Integralgetriebener Algorithmus. In einer inversen Residuentabelle werden zu einem *pip* $\langle o_1 o_2 |$ die Indizes aller Residuen abgespeichert, in deren Residueneinträgen das Orbitalpaar $\langle o_1 o_2 |$ vorkommt. Nachdem alle Matrixelemente berechnet wurden, die sich mit dem Integralen $\langle o_1 o_2 | \dots \rangle$ berechnen lassen, werden die Integrale ringförmig getauscht.

Das Orbitalpaar o_1, o_2 wird im folgenden als *pip* (*primary index pair*), das Orbitalpaar o_3, o_4 als *sip* (*secondary index pair*) bezeichnet. Das Abspeichern der Integrale erfolgt innerhalb unseres Programms so, dass zu einem *pip* jeweils ein Feld von Integraleinträgen gespeichert wird. In jedem Integraleintrag wird das *sip* und der Wert des Integrals gespeichert. Integraleinträge werden nur gespeichert, wenn ihr Wert größer als ein gegebener Schwellwert ist (im Programm 10^{-13}).

Die Aufteilung der Integraldatei auf die einzelnen Knoten erfolgt nun nach den *pips*: Alle Integrale $\langle o_1 o_2 | \dots \rangle$, die zum selben *pip* $\langle o_1 o_2 |$ gehören, werden auf demselben Knoten abgespeichert. In einer *inversen Residuentabelle* (Abb. 3.5) werden für jedes *pip* $\langle o_1 o_2 |$ die Indizes der Residuen abgespeichert, in deren Residueneinträgen das Orbitalpaar o_1, o_2 vorkommt. Im Iterationsschritt werden dann zunächst nur die Matrixelemente berechnet, die sich mit den lokal vorhandenen Integralen auswerten lassen. Danach werden die Integrale ringförmig getauscht.

Dadurch, dass für jeden Residueneintrag der Index des zugehörigen Residuums in der inversen Residuentabelle abgespeichert wird, ist die inverse Residuentabelle genauso groß wie die Residuentabelle selbst. Der Speicherbedarf für den Residuenbaum wird durch den integralgetriebenen Algorithmus daher verdoppelt.

Die Anzahl der Ringtäusche entspricht der Anzahl der Knoten $\#nodes$. Das Datenvolumen, das in einem einzelnen Ringtausch verschickt werden muss, ist umgekehrt proportional zur Anzahl der Knoten. Die Zeit für den gesamten Ringtauschzyklus ist daher, wenn man Latenzzeiten vernachlässigt, die sich beim Etablieren der Kommunikationsprozesse

	Physikernotation	Chemikernotation
1	$\langle o_1 o_2 o_3 o_4 \rangle$	$(o_1 o_2 o_3 o_4)$
2	$\langle o_2 o_1 o_4 o_3 \rangle$	$(o_2 o_1 o_3 o_4)$
3	$\langle o_3 o_4 o_1 o_2 \rangle$	$(o_1 o_2 o_4 o_3)$
4	$\langle o_4 o_3 o_2 o_1 \rangle$	$(o_2 o_1 o_4 o_3)$
5	$\langle o_1 o_4 o_3 o_2 \rangle$	$(o_3 o_4 o_1 o_2)$
6	$\langle o_4 o_1 o_2 o_3 \rangle$	$(o_4 o_3 o_1 o_2)$
7	$\langle o_3 o_2 o_1 o_4 \rangle$	$(o_3 o_4 o_2 o_1)$
8	$\langle o_2 o_3 o_4 o_1 \rangle$	$(o_4 o_3 o_2 o_1)$

Tabelle 3.1: Symmetrien der Integrale in Physiker- und Chemikernotation. Jeweils acht verschiedene Integrale haben denselben Wert. Die Symmetrien der Integrale in der Chemikernotation sind einfacher zu implementieren, da hier niemals Orbitale zwischen *pip* und *sip* ausgetauscht werden.

einstellen, gegeben durch

$$t \sim \#nodes \frac{\#integs}{\#nodes v} = \frac{\#integs}{v}, \quad (3.16)$$

wobei v die Kommunikationsgeschwindigkeit, gemessen in MB/s, ist. Diese Zeit wird auf jedem Knoten verbraucht, der totale Verbrauch an CPU-Zeit lässt sich also berechnen durch

$$T \sim \#nodes \frac{\#integs}{v}. \quad (3.17)$$

Es handelt sich also hier um einen nicht skalierbaren Schritt, man verliert CPU-Zeit, wenn man zu größeren Knotenzahlen übergeht. Man gewinnt jedoch den großen (verteilten) Hauptspeicher des Parallelcomputers. Auf einem Einzelrechner könnte die Integraldatei keinesfalls komplett im Speicher gehalten werden, so dass man auch hier durch die erforderlichen Festplattenzugriffe CPU-Zeit verlieren würde.

Das beschriebene Verfahren zur Verteilung der Integrale funktioniert nur im Iterationsschritt; der Expansionsschritt würde durch eine Verteilung der Integrale sehr ineffektiv werden. Die hier benötigten Integrale sind allerdings so aufgebaut, dass in Gleichung (3.15) o_1 und o_2 *interne Orbitale* sind. o_3 und o_4 können sowohl intern als auch extern sein. Die Anzahl der im Expansionsschritt benötigten Integrale skaliert daher mit $n_i^2 N^2$, wobei n_i die Anzahl der internen Orbitale ist. Da diese sehr viel kleiner ist als die Anzahl der externen Orbitale, können die für den Expansionsschritt benötigten Integrale auf allen Knoten im Hauptspeicher gehalten werden.

Weiterhin werden im Expansionsschritt diagonale Integrale der Form $\langle o_3 o_4 | o_3 o_4 \rangle$ und $\langle o_3 o_4 | o_4 o_3 \rangle$ benötigt. Die Anzahl dieser Integrale skaliert mit N^2 und kann daher auch auf allen Knoten im Hauptspeicher gehalten werden.

Der Algorithmus erfordert, dass die Integrale in der *Physikernotation* (3.15) abgespeichert werden. Ein Orbitalpaar im Residuenbaum entspricht hier genau einem *pip* bzw. *sip*. Speichersparender ist die sogenannte *Chemikernotation*

$$(o_1 o_2 | o_3 o_4) = \int d^3 \vec{r} d^3 \vec{r}' o_1(\vec{r}) o_3(\vec{r}') \frac{1}{|\vec{r} - \vec{r}'|} o_2(\vec{r}) o_4(\vec{r}'). \quad (3.18)$$

Die Symmetrien der Integrale in Physiker- und Chemikernotation sind in Tabelle 3.1 aufgelistet. Natürlich gibt es in Physikernotation genauso viele Symmetrien wie in der Chemikernotation; sie sind jedoch schwieriger zu implementieren. Gar nicht oder nur sehr umständlich sind die Symmetrien wie unter (5) in Tabelle 3.1 zu implementieren, in denen die Orbitale des ursprünglichen Integrals (1) zwischen *pip* und *sip* ausgetauscht wurden. Zusätzliche Schwierigkeiten bereitet, dass die Reihenfolge der Orbitale in *pip* und *sip* in der Physikernotation nicht mehr unabhängig voneinander vertauscht werden kann.

Daher werden, anstatt die Integrale direkt in der Physikernotation abzuspeichern, in unserem Programm zwei neue Integraltabellen definiert:

$$\begin{aligned} I_e(o_1, o_2, o_3, o_4) &= \frac{1}{2} (\langle o_1 o_2 | o_3 o_4 \rangle + \langle o_1 o_2 | o_4 o_3 \rangle) \\ I_o(o_1, o_2, o_3, o_4) &= \frac{1}{2} (\langle o_1 o_2 | o_3 o_4 \rangle - \langle o_1 o_2 | o_4 o_3 \rangle) \end{aligned} \quad (3.19)$$

I_e ist symmetrisch gegenüber Vertauschung der Orbitale in *pip* und *sip*, I_o ändert sein Vorzeichen, nicht aber seinen Wert.

Der eigentliche Wert des Matrixelements hängt auch vom Spin der Orbitale ab. Haben o_1 und o_2 den gleichen Spin (*tps*, *two particle same*), so haben natürlich auch o_3 und o_4 den gleichen Spin wie o_1 und o_2 . Neben dem reinen Coulomb-Integral (3.15) ist noch ein Austauschterm zu berücksichtigen:

$$V_{1234}^{\text{tps}} = I_o(o_1, o_2, o_3, o_4) = \langle o_1 o_2 | o_3 o_4 \rangle - \langle o_1 o_2 | o_4 o_3 \rangle \quad (3.20)$$

Haben o_1 und o_2 unterschiedlichen Spin (*tpd*, *two particle different*), so muss o_3 den gleichen Spin haben wie o_1 und o_4 den gleichen Spin wie o_2 . Der Austauschterm entfällt hier:

$$V_{1234}^{\text{tpd}} = I_o(o_1, o_2, o_3, o_4) + I_e(o_1, o_2, o_3, o_4) = \langle o_1 o_2 | o_3 o_4 \rangle \quad (3.21)$$

Ein Nachteil des integralgetriebenen Algorithmus ist, dass sich das Loadbalancing im Iterationsschritt verschlechtert. Vor jedem Ringtausch der Integrale muss nun jeder Knoten auf den langsamsten Knoten warten, bis dieser die Matrixelemente berechnet hat, die sich mit den lokal vorhandenen Integralen berechnen lassen. Der integralgetriebene Algorithmus wurde daher so implementiert, dass er sich über die Eingabedatei an- und ausschalten lässt.

3.3 Parallele Implementierung

Die wichtigste Forderung bei der Konstruktion eines massiv parallelen Algorithmus ist die Skalierbarkeit: Man fordert, dass die totale CPU-Zeit, also die Summe der CPU-Zeiten der einzelnen Knoten nahezu konstant bleibt. In einem solchen Fall ist die Arbeit perfekt auf die N Knoten des Parallelrechners verteilt; die Laufzeit des Programms skaliert mit $1/N$. Verbleibt hingegen auch nur 1% der Arbeit in nicht skalierbarer Form, so resultiert daraus ein Überhang von 100% der CPU-Zeit, wenn der Prozess auf 100 Knoten gestartet wird.

In unserem Algorithmus für eine zustandsselektierende MRCI sind Kommunikations- und Berechnungsphasen streng getrennt. Bereits vor den zeitintensiven Berechnungsphasen wird die Arbeit unter den Knoten aufgeteilt (statisches Loadbalancing). Dahinter steckt die Idee, nur die besonders effektiv implementierten *globalen* Kommunikationsroutinen der zugrundeliegenden MPI (*message passing interface*) Bibliothek zu nutzen. Es wurden ausschließlich Kommunikationsroutinen aus der MPI-Bibliothek benutzt, um das Programm so portabel wie möglich zu halten.

Tabelle 3.2 zeigt die wichtigsten Schritte unseres MRCI Algorithmus. Das Programm beginnt mit einer (kurzen) Initialisierungsphase (I), in der die Parameter der aktuellen Rechnung (Symmetrie der Wellenfunktion, Partitionierung der Orbitale, Schwellwert der Selektion, ...) eingelesen werden und der Raum der Referenzkonfigurationen gebildet und diagonalisiert wird. Die Initialisierungsphase läuft im wesentlichen auf allen Knoten identisch ab.

Nach der Initialisierung werden die Referenzkonfigurationen gleichmäßig auf alle Knoten verteilt (D1). Jeder Knoten bildet dann alle Übergangsresiduen, die sich aus den Konfigurationen, die er in Schritt D1 erhalten hat, bilden lassen. Der Rechenzeitaufwand für diesen Schritt ist proportional zu $n_e^2 N_{\text{Conf}}/N_k$, er skaliert daher gut mit der Anzahl der Knoten N_k . Im nächsten Schritt (G1) müssen die Residuen (mit den zugehörigen Orbitalpaaren) so auf die Knoten verteilt werden, dass Orbitalpaare, die zum selben Residuum gehören, auf demselben Knoten landen. Da diese Residuen auf unterschiedlichen Knoten erzeugt werden, handelt es sich hierbei um eine nicht triviale Operation. Durch einen *hashing* Mechanismus wird den Residuen zunächst ein eindeutiger Knoten zugewiesen. Dann werden die Residuen auf dem entsprechenden Knoten gesammelt. Das *hashing* Verfahren stellt sicher, dass die Residuen gleichmäßig auf die Knoten verteilt werden.

Auf jedem Knoten wird nun eine Sektion des Residuenbaums gebildet (R2). Da im Expansionsschritt der komplette (interne) Residuenbaum auf allen Knoten vorhanden sein muss, müssen nun die einzelnen Sektionen vereinigt werden. Das geschieht in Schritt G2 so, dass je zwei Knoten ihre Residuenbäume vereinigen. Für diese Operation sind $\log(N_k)$ Kommunikationsschritte erforderlich.

Im Expansionsschritt (E) kann nun jeder Knoten einen bestimmten Abschnitt des Hilbertraums absuchen, um die perturbativen Energiekorrekturen zu bestimmen und die

	Phase		Typ
I	Initialisierung	seriell	Computation
D1	Verteilen des Anfangszustands	parallel	Scatter
Konstruktion des internen Residuenbaums			
R1	Bilden der Residuen	parallel	Computation
G1	Sammeln der Residuen	parallel	All-to-All
R2	Sektion des internen Residuenbaums bilden	parallel	Computation
G2	Zusammenfügen des internen Residuenbaums	parallel	Point-to-Point
Expansion			
E	Expansion	parallel	Computation
D2	Verteilen der selektierten Konfigurationen	parallel	All-to-All
Konstruktion des vollen Residuenbaums			
R3	Bilden der Residuen	parallel	Computation
G3	Sammeln der Residuen	parallel	All-to-All
R4	Sektion des vollen Residuenbaums bilden	parallel	Computation
Q	Ausgleich der zu leistenden Arbeit	Point-to-Point	Computation
Iteration			
M	Berechnung der Matrixelemente	parallel	Computation
A	Ringtausch der Integrale	parallel	Point-to-Point
G4	Aufsummieren des Vielkörperfeldes	parallel	Allsum
X	Davidson Iteration	seriell	Computation

Tabelle 3.2: Unterschiedliche Phasen in der parallelen Implementierung der zustandsselektierenden MRCI Prozedur. Das Programm teilt sich in drei Phasen, die mit der Initialisierung des Programms, der Expansion und der Iteration des Zustandsvektors verbunden sind. Die Details dieser Phasen werden im Text diskutiert. Die dritte Spalte der Tabelle gibt an, ob auf allen Knoten des Rechners das gleiche abläuft (*seriell*), oder ob die Knoten unterschiedliche Aufgaben bearbeiten (*parallel*). Fast die ganze CPU-Zeit fließt in die Schritte E und M, die nächst zeitintensiven Schritte sind D, R2 und G2. Die letzte Spalte gibt für die Kommunikationsschritte den Typ der verwendeten MPI-Routine an.

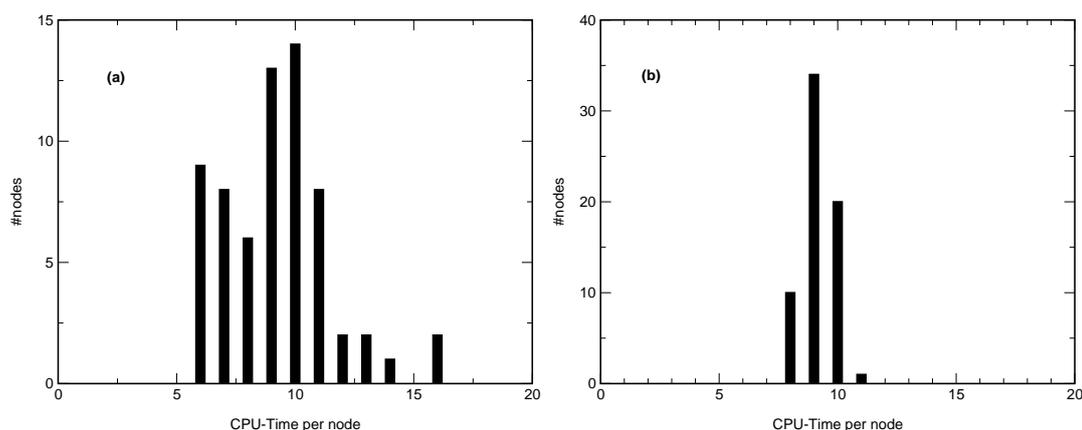


Abbildung 3.6: Histogramm der CPU-Zeit im Iterationsschritt für eine benchmark Rechnung an O₂ wie im Text beschrieben. (a) Ohne Ausgleich der Arbeit (b) mit Ausgleich der Arbeit nach dem Aufbau des vollen Residuenbaums zwischen den Knoten. Die Reduktion der Breite der Verteilung verbessert die Skalierbarkeit des Algorithmus.

Konfigurationen des variationellen Subraums zu selektieren. Neben der noch folgenden Iteration erfordert der Expansionsschritt den größten Aufwand an CPU-Zeit.

Die nächsten vier Schritte bereiten die Iteration des variationellen Subraums vor. Da sich die Anzahl der selektierten Konfigurationen auf den einzelnen Knoten stark unterscheiden kann, erfolgt zunächst eine Umverteilung (D2). In diesem Schritt wird weiterhin sichergestellt, dass sich Konfigurationen mit demselben externen Orbitalpaar auf einem Knoten befinden, damit die durch XX-Residuen vermittelten Matrixelemente, wie in Abschnitt 3.1 beschrieben, berechnet werden können.

Die nun folgenden Schritte R3, G3 und R4 sind völlig analog zu den entsprechenden Schritten R1, G2 und R2 beim Aufbau des internen Residuenbaums: Es werden auf einem Knoten alle Residuen (und Orbitalpaare) gebildet, die sich aus den vorhandenen Konfigurationen bilden lassen (R3). Dann werden die Residuen so auf die Knoten verteilt, dass sich gleiche Residuen auf dem gleichen Knoten befinden (G3) und schließlich wird auf jedem Knoten eine Sektion des Residuenbaums gebildet (R4). Es ist jedoch nicht nötig (und speichertechnisch in der Regel auch gar nicht möglich), den Residuenbaum zu vereinigen. Jeder Knoten operiert während der folgenden Iteration nur auf seiner Sektion des Residuenbaums.

Der Rechenaufwand für die Berechnung der Matrixelemente ist proportional zum Erwartungswert des Quadrats der Anzahl der Orbitalpaare über die Residuen. Das *hashing* Verfahren, das wir verwenden, um die Residuen auf die Knoten zu verteilen, stellt sicher, dass die Anzahl der Residuen auf den unterschiedlichen Knoten ungefähr gleich ist. Dennoch kann der Rechenaufwand auf den Knoten signifikant variieren (3.6a), ein Effekt, der sich mit einer wachsenden Anzahl von Knoten verschlimmert. Ein solches Ungleichgewicht in der Verteilung der Arbeit führt zu einem Verlust der Skalierbarkeit des Algorithmus.

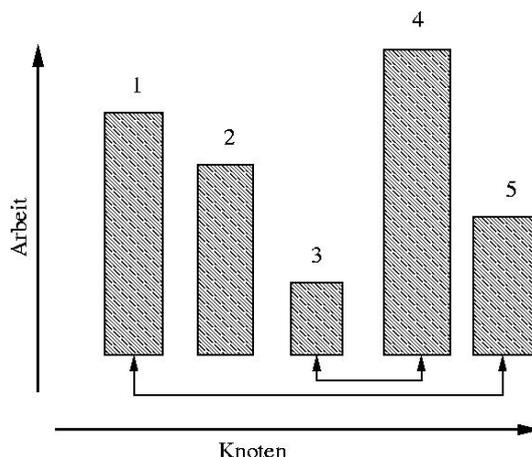


Abbildung 3.7: Iterativer Ausgleich der Arbeit nach dem Aufbau des vollen Residuenbaums (Schritt Q). Nach der Konstruktion wird zunächst die Arbeit berechnet, die jeder Knoten zu leisten hat. Der Knoten mit der meisten Arbeit gibt dann Pakete von Orbitalpaaren an den Knoten mit der wenigsten Arbeit ab, usw.. Nach wenigen Iterationen hat sich die zu leistende Arbeit innerhalb der Grenzen des Verfahrens ausgeglichen.

Es ist daher notwendig, eine Umverteilung des Arbeitsaufwandes vorzunehmen (Q), um eine bessere Performance zu erreichen.

Um das zu erreichen verwenden wir ein iteratives Ausgleichsverfahren, das in Abb. 3.7 veranschaulicht ist: Für jeden Knoten wird zunächst die zu leistende Arbeit berechnet. Der Knoten, der die meiste Arbeit zu leisten hat, gleicht sich dann mit dem Knoten ab, der die wenigste Arbeit zu leisten hat, usw.. Die Arbeit wird ausgeglichen, indem einige Pakete von Orbitalpaaren von einem Knoten zu einem anderen gesendet werden. Nach dem Ausgleich wird die zu leistende Arbeit neu berechnet, und der Ausgleich beginnt von vorne. Dadurch, dass immer nur Pakete von Orbitalpaaren ausgetauscht werden können (nämlich alle Orbitalpaare, die zu einem Residuum gehören), sind dem Verfahren Grenzen gesetzt. Diese werden in der Regel nach 2–3 Iterationen erreicht.

Durch das Ausgleichsverfahren ließ sich in der Praxis die Abweichung der Arbeit von ihrem Mittelwert von über 50% auf unter 12% reduzieren. Abb. 3.6 zeigt Histogramme der Verteilung der CPU-Zeit mit und ohne den Ausgleich. Die Zeit, die das Ausgleichsverfahren selbst benötigt, beträgt wenige Sekunden.

Nach der Umverteilung des Residuenbaums kann das Programm mit den Iterationsschritten fortfahren, um den Grundzustand im variationellen Subraum zu bestimmen. Die vier Phasen, die den Iterationsschritt bilden werden mehrfach wiederholt. Sie benötigen jedoch keine neuen logischen Informationen. In Schritt M, auf den fast die ganze Rechenzeit im Iterationsschritt entfällt, wird auf jedem Knoten das Feld $\vec{q} = \hat{H}\vec{c}$ berechnet wie in Abschnitt 3.1 beschrieben. Allerdings werden auf einem Knoten zunächst nur die Matrixelemente berechnet, die sich mit den lokal vorhandenen Integralen berechnen lassen.

Anschließend werden die Integrale zwischen den Knoten ringförmig getauscht (A), wonach Schritt M von vorne mit neuen Integralen beginnt. Nach N_k Wiederholungen sind auf den einzelnen Knoten alle Matrixelemente berechnet worden, die durch die entsprechende lokale Sektion des Residuenbaums kodiert wurden. Es ist nun erforderlich, die lokalen Beiträge zum Feld \vec{q} aufzusummieren (G4). Aus den Vektoren \vec{c} und \vec{q} , die nun auf allen Knoten identisch vorliegen, wird nun mit dem Verfahren von Davidson (Abschnitt 2.2) ein neuer Startvektor \vec{c} gebildet. Die Davidson-Iteration selbst — hier muss lediglich eine Matrix der Größenordnung 10×10 diagonalisiert werden — dauert nur den Bruchteil einer Sekunde. Nach 8–10 Iterationen ist der Vektor \vec{c} in der Regel konvergiert.

3.4 Programmierung in C++

Nachdem nun in den vorangegangenen Abschnitten die grundlegenden Konzepte des parallelen MRD-CI Algorithmus abstrakt diskutiert worden sind, soll hier noch ein Einblick in die Implementierung gegeben werden. Es geht dabei weniger um alle technischen Details, deren Diskussion den Rahmen dieser Arbeit übersteigen würde, als um einen Eindruck vom Gebrauch objektorientierter Techniken in einem wissenschaftlichen Kontext.

Objektorientierte Techniken erfahren eine starke Verwendung in industriellen Programmen, während es im wissenschaftlichen Bereich und insbesondere in der Physik oder Chemie nur wenige Beispiele gibt. Unser komplettes Quantenchemie Programmpaket (mit mehr als 100.000 Zeilen Programmcode, davon 19.000 Zeilen für die parallele MRD-CI) wurde in C++ geschrieben. Objektorientierte Techniken wie Datenkapselung, Vererbung, Templates und überladbare Operatoren haben sich hierbei als äußerst hilfreich erwiesen.

Im Gegensatz zu prozeduralen Programmiersprachen konzentriert man sich in der objektorientierten Programmierung auf das Design von Klassen, in denen spezielle Ideen oder mathematische Konstrukte des Problems gekapselt werden. Kapselung bedeutet in diesem Fall, dass der Benutzer der Klasse auf ihre Daten nur über eine wohldefinierte Schnittstelle zugreifen kann, sich jedoch keine Gedanken über die konkrete Organisation der Daten oder die genaue Funktionsweise der Operationen machen muss. Dadurch, dass Daten und Funktionen zu kleinen Einheiten, den Objekten, zusammengefasst werden, ist es wesentlich einfacher, auch radikale Änderungen des Codes — von denen das vorliegende Programm im Laufe der Zeit einige erfahren hat — durchzuführen. Unserer Erfahrung nach verringert sich die Zeit, die man zur Programmentwicklung und zur Fehlersuche aufwenden muss, um mindestens einen Faktor zwei.

Die Hierarchie der Klassen ergibt sich meistens völlig natürlich aus dem zugrundeliegenden Problem. Der Aufbau einer einfachen Klasse soll im folgenden am Beispiel der Klasse `Orbital` gezeigt werden. Ein `Orbital` ist gekennzeichnet durch eine Symmetrie (s. Tab. 2.4), einen Typ (*inaktiv, aktiv, extern, . . .*) und einen laufenden Index. Diese Daten werden in der Klasse `Orbital` *privat* gespeichert; der Benutzer kann sie nicht direkt verändern. Initialisiert werden kann die Klasse entweder durch die Übergabe von Typ,

Symmetrie und Index oder durch Übergabe eines Objekts selben Typs. Auch mit nur rudimentären C++ Kenntnissen ist man in der Lage, die folgende Klassendefinition zu verstehen:

```
class Orbital
{
private:
    Symmetry    v_s;        // orbital symmetry
    int         v_i;        // orbital index
    OrbitalType v_type;    // orbital type

    friend class OrbitalIter;
    friend class OrbitalSymIter;

public:
    Orbital(OrbitalType intype, Symmetry sym, int index) // Constructor
        : v_s(sym), v_i(index)    { v_type = intype; }
    Orbital(const Orbital& o) // Constructor
        : v_s(o.v_s), v_i(o.v_i), v_type(o.v_type){}
    ~Orbital(); // Destructor
    const Orbital& operator=(const Orbital& o);

    Symmetry symmetry() const { return v_s; }
    int index() const { return v_i; }
    OrbitalType type() const { return v_type; }

    static int number_of_orbitals();
    static int number_of_orbitals(const OrbitalType& type);
    static int number_of_orbitals(const Symmetry& s );
    static int number_of_orbitals(const OrbitalType& type, const Symmetry& s );

    int compare(const Orbital& a) const {return number() - a.number();};

    int in_range();
    const Orbital& operator++();

    ...
};
```

Das Beispiel zeigt, dass sauber geschriebener C++ Code einfach zu dokumentieren ist und die Klassendefinition — frei von allen Details der Implementierung — die gesamte Funktionalität der Klasse beschreibt.

Die Funktion `compare()` erlaubt es, Vergleichsoperatoren wie z.B.

```

inline int operator< (const Orbital& o1,const Orbital& o2)
{
    return o1.compare(o2) < 0;
}

```

zu schreiben, die es erlauben, Orbitale genau wie Integerzahlen zu vergleichen:

```

if (o1<o2) ...

```

Analog dazu erlauben die beiden letzten Funktionen es, Schleifen über Orbitale genauso zu schreiben, wie simple Schleifen über Integerzahlen:

```

for (Orbital o(type,sym,index);o.in_range();o++)
{
    ...
}

```

Auf diese Weise kann man sich auf den Inhalt der Schleife konzentrieren, anstatt sich in verschachtelten Schleifen um die Inkrementierung zu kümmern. Auch muss man die Inkrementierung nur einmal implementieren, so dass der Code unter der Voraussetzung, dass man die Schleife mehrmals verwendet, insgesamt kürzer wird.

Mehrere Orbitale bilden eine Konfiguration, die durch die Klasse `Residue` implementiert wird. Der Spin der Orbitale wird berücksichtigt, in dem die Konfiguration in einen *up*- und einen *down*-Teil aufgeteilt wird. *up*- und *down*-Teil der Konfigurationen werden durch die Klasse `SpinResidue` implementiert:

```

class SpinResidue
{
    ...

    static SArray<FullSpinResidue> inactive;
    static SArray<FullSpinResidue> active;
    static SArray<FullSpinResidue> external;
    static SArray<INT_4> del_i;
    static SArray<INT_4> del_a;
    static SArray<INT_4> del_e;
    static SArray<INT_4> add_i;
    static SArray<INT_4> add_a;
    ...
}

```

```

    int idi;
    int ida;
    int ide;

public:
    // construction and destruction
    SpinResidue(){};
    SpinResidue(int idi, int ida, int ide);
    SpinResidue(const SpinResidue& src);
    SpinResidue& operator=(const SpinResidue& src);
    SpinResidue(const FullSpinResidue& src);
    ~SpinResidue(){};

    // information
    int size() const; // read only access
    int number_of_external_orbitals() const; // read only access
    int number_of_internal_orbitals() const;
    ...

    // annihilation and creation of orbitals
    Orbital remove(int pos);
    int add(Orbital o);

    // comparison
    int compare(const SpinResidue& conf) const;

    // input and output
    void write(ostream& os);
    void read (istream& is);
};

```

Hier zeigt sich der volle Vorteil von C++: Anstatt jede Konfiguration als ein Feld von Orbitalen zu implementieren, was den Code erheblich speicherintensiver und zudem langsamer machen würde, werden intern nur die drei Indizes *idi*, *ida* und *ide* gespeichert, die für eine bestimmte Konfiguration von inaktiven, aktiven und externen Orbitalen stehen. Der genaue Aufbau dieser Konfigurationen ist in statischen Tabellen, d.h. Tabellen, die in jeder Instanz der Klasse identisch vorliegen, gespeichert. Der Benutzer der Klasse SpinResidue kommt mit diesen Tabellen jedoch überhaupt nicht in Berührung. Er kann durch die Funktionen `add()` und `remove()` Orbitale erzeugen und vernichten ohne Einblick in die konkrete Programmierung dieser Funktionen zu haben.

Als hilfreich hat sich auch das Konzept der *templates* erwiesen. Hierbei handelt es sich um Klassen, die wiederum Funktionen von anderen Klassen sind. *templates* finden keine Entsprechung in den prozeduralen Programmiersprachen und eröffnen eine neue Welt des

Programmierens. Im folgenden soll das Konzept der *templates* am Beispiel eines Feldes (*Array*) von Objekten illustriert werden, wie es ständig in wissenschaftlichen Programmen vorkommt. Ein komplexes benutzerdefiniertes Objekt könnte hier beispielsweise die oben erwähnte Klasse `Residue` sein. Ein solches Feld von Konfigurationen würde dann, zusammen mit den entsprechenden Koeffizienten der Konfigurationen, einen Zustandsvektor bilden.

```
template <class ArbitraryType> class Array
{
    private:
        ArbitraryType *A;
    public:
        ArbitraryType& operator[](const int& n) { return A[n]; }
}
```

Ein Zustandsvektor ließe sich nun beispielsweise folgendermaßen implementieren:

```
class State
{
    private:
        Array<Residue> config;
        Array<double> coef;
    public:
        ...
}
```

Die beiden Felder `config` und `coef` benutzen dieselbe template-Klasse `Array`. Diese Wiederbenutzbarkeit ist das zentrale Anliegen von template-Klassen und -Funktionen.

Am obigen Beispiel lässt sich jedoch auch der Mechanismus der Vererbung illustrieren. Tatsächlich ist nämlich der Zustandsvektor von der Klasse `Array` abgeleitet:

```
class State : public Array<double>
{
    //Referenz to instance of class StateInfo:
    StateInfo& info;

    public:
        State(StateInfo& inf);
        double normalize();
        double product(State& s2);
        ...
}
```

In diesem Zusammenhang ist `Array<double>` keine Eigenschaft der Klasse `State`, sondern dient als Basisklasse: Seine Eigenschaften und Methoden dienen als Anfangskomponenten der Klasse `State`.

Das Feld von Konfigurationen ist in der Klasse `StateInfo` versteckt, wobei es sich wiederum um eine von `Array<Residue>` abgeleitete Klasse handelt. Eine der Eigenschaften von `State` ist eine Referenz auf eine Instanz dieser Klasse. Der Vorteil dieser etwas komplizierteren Konstruktion liegt auf der Hand: Bei der Berechnung der Grundzustandsenergie mit dem in Abschnitt 2.2 beschriebenen Davidson-Verfahren können viele Instanzen der Klasse `State` vorkommen. Um Speicherplatz zu sparen ist es daher wünschenswert, das Feld der Konfigurationen nur ein einziges Mal im Speicher zu halten. Das wird durch die obige Struktur erreicht: Dem Konstruktor der Klasse `State` wird eine Instanz der Klasse `StateInfo` übergeben, auf die ein Verweis in der Eigenschaft `info` abgespeichert wird.

Kapitel 4

Benchmarkrechnungen

4.1 Leistung des nicht integralgetriebenen MRD-CI Algorithmus

Um die Skalierbarkeit der Implementierung zu demonstrieren, haben wir Benchmarkrechnungen für typische Anwendungen des Programms durchgeführt. Das erste Beispiel behandelt die Wichtigkeit von drei- und vierfach angeregten Konfigurationen für die Potentialenergieoberflächen (PES, *potential energy surface*) des Sauerstoffmoleküls und seinem Anion. Frühere Rechnungen zeigten, dass die Berechnung der Elektronenaffinität von O_2 auch mit Mitteln der modernen Quantenchemie schwierig bleibt. Auf dem Niveau einer CAS-SCF Rechnung, wird die Elektronenaffinität mit falschem Vorzeichen vorausgesagt, unabhängig vom verwendeten Basissatz. Eine ausführliche Studie zeigte, dass stark unterschiedliche differentielle Korrelationseffekte des Sauerstoffs und seines Anions vermutlich die Quelle dieser Diskrepanz darstellen. In MRCI-SD Rechnungen erhält man nicht einmal unter Verwendung von *quadruple zeta* Basissätzen das richtige Vorzeichen der Elektronenaffinität. Das richtige Vorzeichen und annähernd den richtigen Wert erhält man, wenn man die Davidson-Korrektur oder andere Extensivitätskorrekturen auf die MRCI-SD Ergebnisse anwendet, um den Effekt der höheren Anwendungen abzuschätzen.

Das Sauerstoffmolekül berührt daher trotz seiner Einfachheit eines der zentralen Paradigmen der Quantenchemie, nämlich dass die explizite Behandlung der Ein- und Zweifachanregungen eines chemisch motivierten Satzes von Referenzen ausreicht, die dynamischen Korrelationseffekte quantitativ richtig zu behandeln. Diese Beobachtung, als auch der Wunsch, Extensivitätskorrekturen der MR-SDCI explizit zu testen, war eine der Motivationen zur Entwicklung des vorliegenden Programmcodes. Da in einer CAS + SDTQ Rechnung der Hilbertraum von O_2 die Dimension 32×10^9 annimmt, kann das Problem mit keiner der bisher verfügbaren MR-SDCI oder MRD-CI Implementierungen behandelt werden; es erweist sich jedoch als geeigneter Test für unsere parallele Implementierung. Die Rechnungen wurden durchgeführt in einer (sp)-augmentierten cc-pVTZ Basis in D_{2h}

Symmetrie und bei den experimentell gefundenen Bindungsabständen. Die etwa 15 MB große Integraldatei konnte auf allen Knoten im Hauptspeicher gehalten werden, so dass die Rechnungen im nicht integralgetriebenen Modus des Programms durchgeführt wurden. In Hilberträumen einer Größe von bis zu 5×10^9 haben wir, je nach Größe des von 10^{-3} bis 10^{-6} reichenden Schwellwerts für die Koeffizienten, bis zu 3×10^7 Konfigurationen selektiert. Die Ergebnisse dieser Rechnungen, und vergleichbarer Rechnungen an NO und BO, sind in Abschnitt 5.1 zusammengefasst.

In einem zweiten Beispiel, das eher den klassischen Anwendungen einer zustandsselektierenden MRCI entspricht, haben wir die Grundzustandsenergie von Benzol in einer cc-pVDZ BASIS berechnet. Die Größe der Integraldatei war mit 55 MB immer noch klein genug, um sie auf allen Knoten unverteilt im Hauptspeicher zu halten. Die Referenzräume umfassten sechs und zwölf aktive Orbitale. Zwar ist der kleine aktive Raum ausreichend, um die Chemie von Benzol adäquat zu beschreiben. Es wurden dennoch zwölf aktive Orbitale verwendet, um das Programm für sehr große Hilberträume zu testen. Die Rechnungen wurden durchgeführt in einer C_{6v} Geometrie in D_{2h} Symmetrie. Der Hilbertraum umfasste bis zu 3×10^9 Konfigurationen, von denen bis zu 2×10^7 für die variationelle Bestimmung der Grundzustandsenergie selektiert wurden.

Wie bereits erwähnt, ist die wichtigste Forderung, die an ein paralleles Programm gestellt werden muss, seine Skalierbarkeit mit der Anzahl der Prozessoren, die für eine gegebene Rechnung verwendet wurden. Um die Skalierbarkeit des Codes zu demonstrieren, haben wir zwei typische Läufe ausgewählt mit 10^9 Konfigurationen (1.8×10^6 selektiert) für O_2 und 1.3×10^9 Konfigurationen (1.6×10^6 selektiert) für Benzol. Die Rechnungen wurden durchgeführt auf der 256-Knoten IBM SP/2 des Rechenzentrums Karlsruhe.

Abb. 4.1 zeigt den gesamten Rechenzeitaufwand der oben erwähnten Rechnungen auf der IBM-SP2 als eine Funktion der Anzahl der Knoten. In diesen Plots sind die Iterationsschritte (M, A, G4, X), die logischen Schritte zum Aufbau des Residuenbaums (R1, G1, R2, G2, R3, G3, R4, Q) und die Expansionsschritte (E, D2) zu jeweils einer Kategorie zusammengefasst. Diese Unterteilung ist dadurch motiviert, dass die relative Wichtigkeit dieser drei Schritte mit dem Typ der durchgeführten Berechnung variiert. Durch ein unterschiedliches Verhalten dieser Schritte ergeben sich signifikante Änderungen in der Leistungsfähigkeit des Programms bei verschiedenen Rechnungen. Eine Änderung des Schwellwerts für die Koeffizienten, führt beispielsweise dazu, dass das relative Gewicht der Iterationsschritte auf Kosten der Expansionsschritte zunimmt. Erhöht man die Anzahl der Zustände, deren Energien zu berechnen sind (in den Beispielrechnungen wurde nur die Grundzustandsenergie berechnet), so reduziert das die Wichtigkeit der logischen Schritte. Der Aufwand für die Expansion würde sich in einem solchen Falle nicht deutlich erhöhen. Die Zeit für die Iterationsschritte erhöht sich um ungefähr einen Faktor N , wenn N die Anzahl der Zustände ist.

Für Benzol finden wir eine fast perfekte Skalierung von 48 bis 128 Knoten auf der IBM-SP2. Der CPU-Aufwand für die Expansionsschritte ist konstant innerhalb von 0.4%, wenn man von der kleinsten auf die größte Knotenzahl übergeht. Im Gegensatz dazu wächst

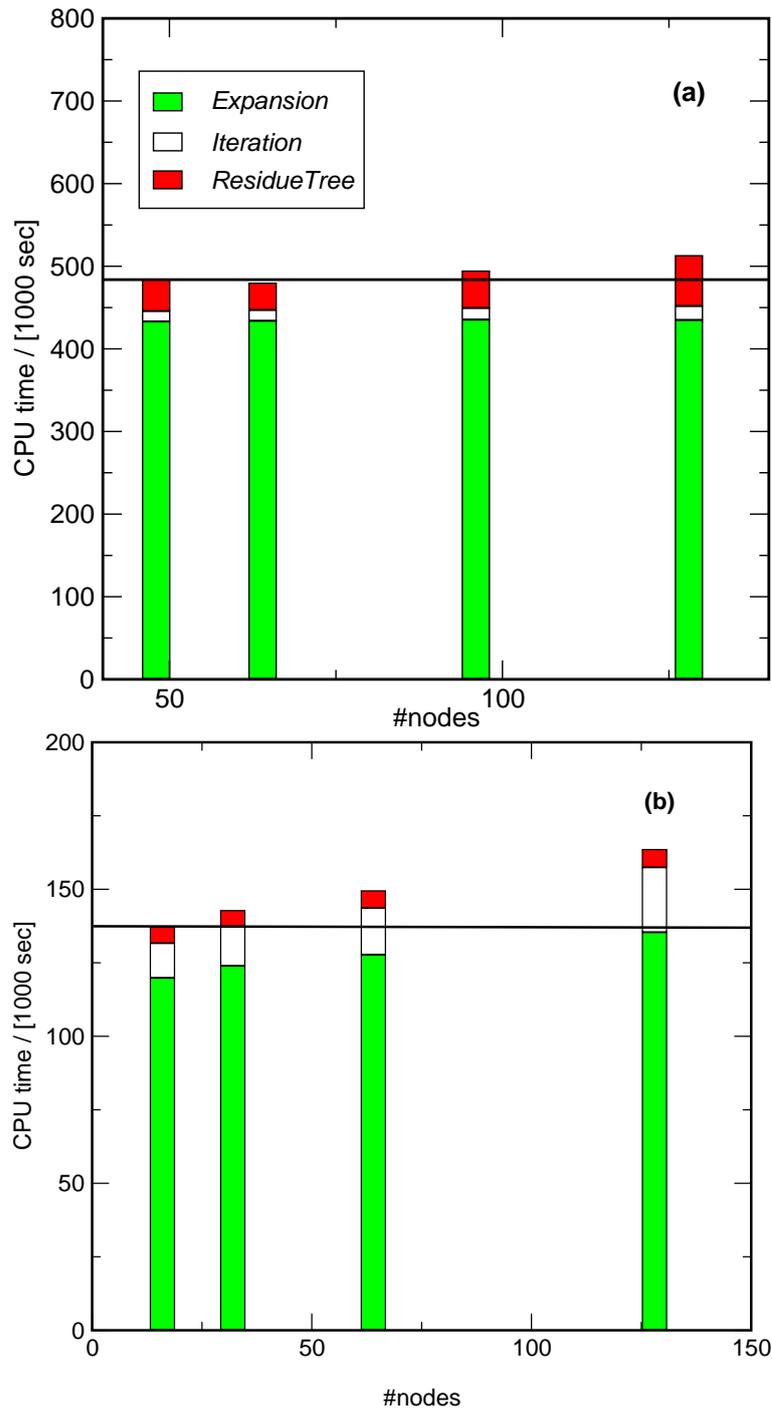


Abbildung 4.1: Totale CPU Zeit für die voll konvergierten Berechnungen der Grundzustandsenergien von Benzol (a) und Sauerstoff (b), die im Text beschrieben sind, als Funktion der Zahl der Knoten auf der IBM-SP2. Eine gerade Linie würde ein perfektes Skalieren des Rechenzeitaufwands mit der Zahl der Knoten bedeuten. Die markierten Bereiche innerhalb der Balken fassen von oben nach unten die Iterationsschritte (M, A, G4, X), die logischen Schritte zum Aufbau des Residuenbaums (R1, G1, R2, G2, R3, G3, R4, Q) und die Expansionsschritte (E, D2) zusammen.

der Aufwand für die logischen Schritte, in denen auch einiger Kommunikationsaufwand zwischen den Knoten beinhaltet ist, ein wenig an. Das ist zu erwarten, da die Kommunikation mit der Anzahl der Knoten anwächst und insgesamt 3.7 und 9.1 GB an Daten für den internen bzw. vollen Residuenbaum über den Parallelrechner verteilt werden müssen. Insgesamt erhält man einen Beschleunigungsfaktor von 1.86, wenn man von 64 Knoten zu 128 Knoten übergeht.

Bei O_2 lässt sich ein stärkerer Anstieg des Rechenzeitaufwands beobachten, bei Knotenzahlen von 32 bis 128. Insbesondere die letzte Verdopplung der Anzahl der Knoten von 64 auf 128 lässt sich ein stärkeres Anwachsen der CPU-Zeit feststellen. Wie bei Benzol ist der CPU-Zeit Bedarf des Expansionsschritts relativ stabil. Er erhöht sich um nur 6,7% (6,3%) wenn man von 32 zu 64 (64 zu 128) Knoten übergeht. Dieses ist das Resultat des nahezu perfekten *load balancing*, das in Abb. 4.2 dargestellt ist. Gezeigt wird hier ein Histogramm der CPU-Zeit *pro Knoten* für den Expansionsschritt bei einem Lauf auf 64 Knoten.

Durch die relativ breite Verteilung des Rechenzeitaufwands pro Residuum (Abb. 4.3) wird es immer schwieriger, den Load im Iterationsschritt zwischen den Knoten auszugleichen. Daraus resultiert, verglichen mit dem Expansionsschritt, eine deutlich breitere Verteilung des Loads (Abb. 4.2). Die Performance des Iterationsschritts wird dadurch verringert, da alle Prozessoren auf den langsamsten Knoten warten müssen. Da für die Berechnung der Matrixelemente große Datenmengen erforderlich sind, ist es schwierig, die vorliegende Implementierung dahingehend zu verbessern, den Load dynamisch zu balancieren, während der Iterationsschritt bereits in Gang ist. Ein solches dynamisches *load balancing* wäre aber ohnehin höchstens marginal besser als das von uns praktizierte statische *load balancing*: Wie Abb. 4.3 illustriert, hat jedes Ausgleichsverfahren Grenzen. So bedeuten die wenigen Residuen mit 1600 Orbitalpaaren mehr Arbeit ($1600^2 = 2,5$ Mio.) als alle Residuen mit weniger als 100 Orbitalpaaren zusammen. Ab einer bestimmten Anzahl von Knoten ist daher kein Ausgleich der Arbeit mehr möglich.

Die Daten für die Testläufe sind in Tabelle 4.1 zusammengefasst. Auf der IBM-SP2 finden wir nahezu perfekte Beschleunigungsfaktoren für Benzol. Die mit einer Verdopplung der Knotenzahl verbundene Beschleunigung ist im Fall von Sauerstoff etwas schlechter, erlaubt aber dennoch, die Rechnung auf großen Knotenzahlen durchzuführen. Die Cray-T3E bestand zum Zeitpunkt der Durchführung der Testrechnungen aus zwei physikalisch unterschiedlichen Maschinen. Das kleinere Cluster erlaubte Läufe von 16 bis zu 64 Knoten; das größere Läufe zwischen 65 und 256 Knoten. Die Interpretation der Skalierungsdaten muss diesen Bruch berücksichtigen. Auf 128 Knoten finden wir für die Benzol-Rechnung, durchgeführt im realistischeren aktiven Raum aus 6 Orbitalen, eine Programmlaufzeit von weniger als drei Minuten. Die Residuentabelle ist hier so dünn auf die Knoten verteilt, dass ein Ausgleich der Arbeit unmöglich ist. Das erklärt den etwas großen Verlust von 20% an Effizienz beim Schritt von 65 auf 128 Knoten.

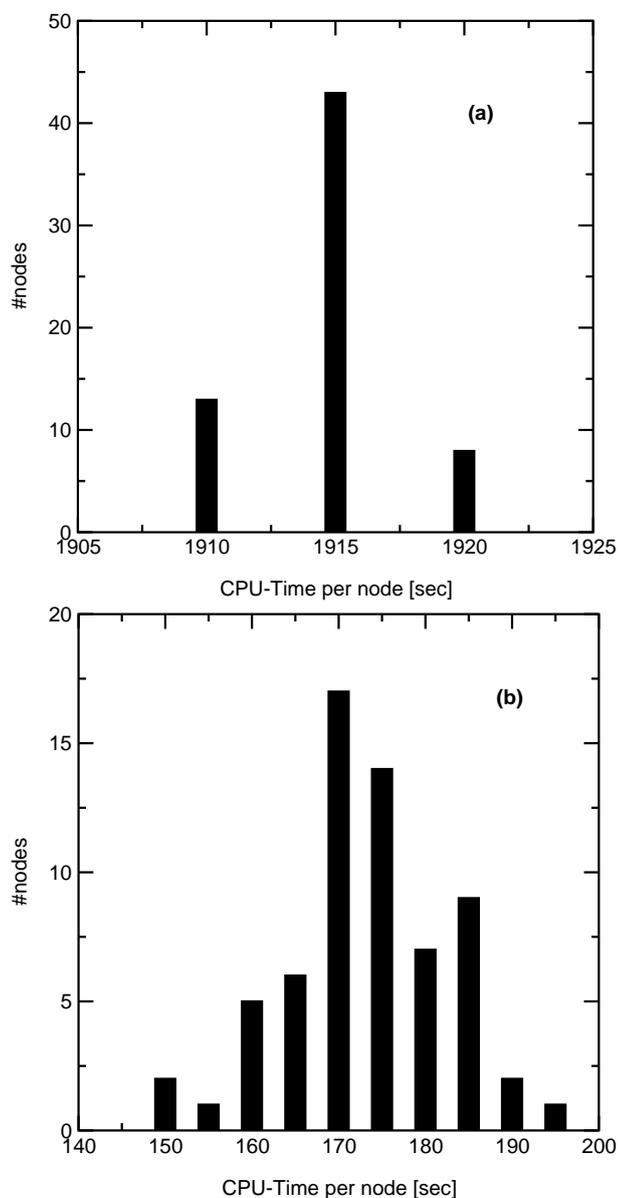


Abbildung 4.2: Histogramm der CPU-Zeit Verteilung für (a) die Expansionsschritte und (b) die Iterationsschritte für die im Text beschriebene O_2 Benchmarkrechnung auf 64 Knoten der IBM SP/2. Der Rechenzeitaufwand im Expansionsschritt ist fast perfekt verteilt; der des Iterationsschritts variiert mit einer Standardabweichung von ungefähr 4%, die zu einem Verlust an Effizienz führt, da die schnellen Knoten nun auf die langsamen Knoten warten müssen. Da die Zahl der Residuen pro Knoten mit steigender Knotenzahl abnimmt, wird es bei großer Knotenzahl schwieriger, Schwankungen des Arbeitsaufwandes auszugleichen.

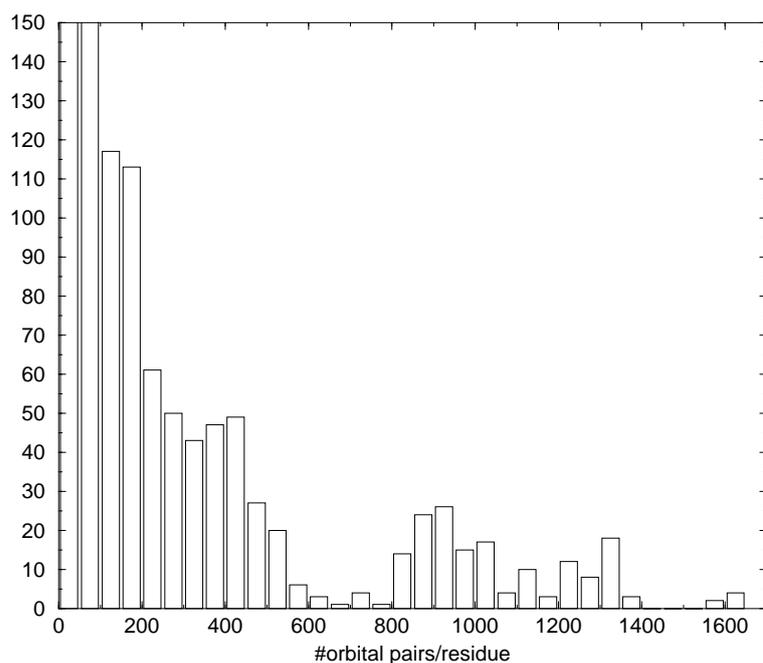


Abbildung 4.3: Histogramm der Verteilung der Anzahl der Orbitalpaare pro Residuum im Iterationsschritt für die im Text beschriebene O_2 Benchmark Rechnung. Der CPU-Zeitbedarf pro Residuum ist proportional zum Quadrat der Zahl der Orbitalpaare. Diese Abhängigkeit macht unseren Algorithmus effizient, da die Konstruktion des Residuenbaums viel weniger kostet als die Berechnung der Matrixelemente. Andererseits erschwert eine breite Verteilung einen Ausgleich des Loads im Iterationsschritt. Da die Zahl der Residuen pro Knoten mit der Zahl der Knoten abnimmt, sind Schwankungen im Rechenzeitaufwand für große Knotenzahlen immer schwieriger zu balancieren.

	IBM-SP2			
	O ₂		C ₆ H ₆	
number of nodes	time (s/node)	loss	time (s/node)	loss
16	8308			
32	4374	5%		
48			10122	
64	2480	13%	7510	0%
96			5160	3%
128	1410	14%	4012	4%
	CRAY-T3E			
	O ₂		C ₆ H ₆	
16	5334			
32	2887	8%	319	
64	1539	10%	172	8%
65	2062		244	
128	1107	6%	147	20%
256	620	12%		

Tabelle 4.1: Totale CPU-Zeiten für die im Text beschriebenen Benchmarkrechnungen auf der IBM-SP2 und der Cray T3E als eine Funktion der Anzahl der Knoten. Der Verlust ist definiert als das Verhältnis der CPU-Zeiten pro Knoten geteilt durch den perfekten Beschleunigungsfaktor, der durch das Verhältnis der Knotenzahlen gegeben ist. Die den Verlust betreffenden Zahlenwerte beziehen sich immer auf aufeinanderfolgende Einträge. Die Rechnung für Benzol auf der IBM-SP2 wurde unter Verwendung von 12 aktiven Orbitalen durchgeführt; diejenige auf der Cray-T3E benutzte einen realistischen aktiven Raum von sechs aktiven Orbitalen. Eine vernünftige obere Grenze der Knotenzahl für die letzter Rechnung liegt bei 64 Knoten; die Rechnung erfordert dann weniger als 3 Minuten.

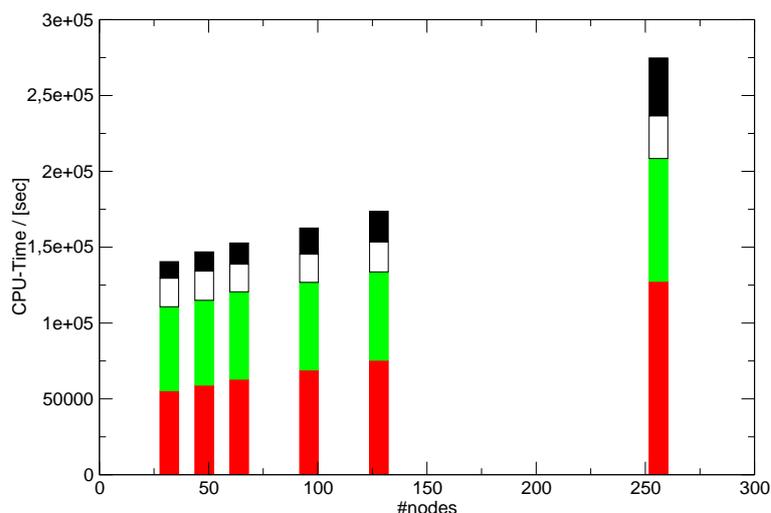


Abbildung 4.4: Skalierungsplot der totalen CPU-Zeit gegen die Zahl der Knoten einer MRD-CI Berechnung der Grundzustandsenergie von Dinitrosoethylen, in der 3,7 Millionen von 20 Millionen Konfigurationen selektiert wurden. Die einzelnen Beiträge sind (von oben nach unten): Vorbereitung (Berechnung der Integrale, SCF, Vierindextransformation), Logik (Konstruktion und Verteilung der Residuentabelle), Expansion (Konfigurationsselektion inklusive der perturbativen NO-Generierung) und Iteration.

4.2 Leistung des integralgetriebenen Algorithmus und des Gesamtpakets

Die Beispiele des letzten Abschnitts haben sich mit der Skalierbarkeit des Programms im nicht integralgetriebenen Modus beschäftigt und auch nur den MRCI-Schritt behandelt. Es soll daher nun noch die Leistungsfähigkeit des integralgetriebenen Modus und des Gesamtpakets demonstriert werden. Wir haben dazu Rechnungen zur Berechnung der Grundzustandsenergien des einfachsten Mitglieds der Endiin-Familie und Dinitrosoethylen durchgeführt. Die Endiin-Rechnung wurde auf der IBM SP3/SMP des Rechenzentrums Karlsruhe durchgeführt. Der Referenzraum umfasste 10 aktive Orbitale, der Hilbertraum umfasste 26 Millionen Konfigurationen, von denen 4,2 Millionen selektiert wurden. Es wurde eine cc-pVDZ Basis von Dunning verwendet. Die Dinitrosoethylen-Rechnung wurde auf der T3E-1200 des Forschungszentrums Jülich durchgeführt. Der Referenzraum umfasste hier 9 aktive Orbitale, der Hilbertraum 20 Millionen Konfigurationen und selektiert wurden 3,7 Millionen Konfigurationen. Für die Rechnung wurde eine (sp)-augmentierte cc-pVDZ Basis von Dunning verwendet.

In den Abbildungen 4.4 und 4.5 sind die totalen CPU-Zeiten der beiden Skalierungsläufe gegen die Anzahl der Knoten aufgetragen. Die CPU-Zeiten sind dabei unterteilt nach wichtigen (nicht notwendig zusammenhängenden) Teilen des Programmablaufs: (a) Berechnung der Integrale, Hartree-Fock, Vierindextransformation, (b) Konstruktion und

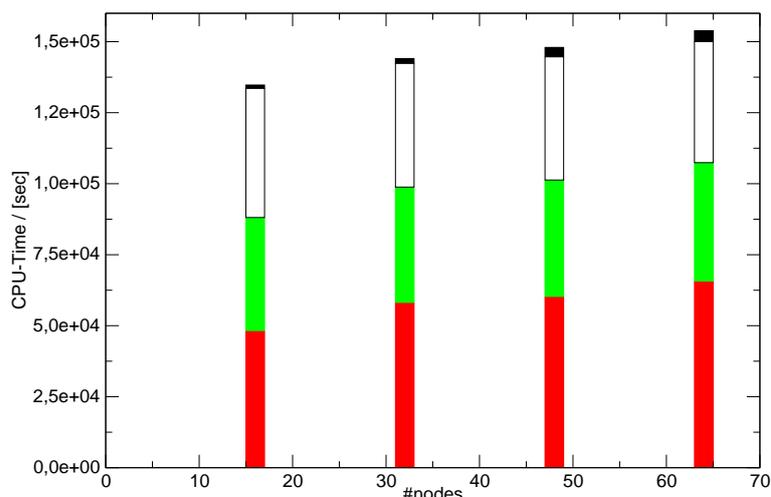


Abbildung 4.5: Skalierungsplot der totalen CPU-Zeit gegen die Zahl der Knoten einer MRD-CI Berechnung der Grundzustandsenergie von Endiin, in der 4,2 Millionen von 26 Millionen Konfigurationen selektiert wurden. Die einzelnen Beiträge entsprechen denen in Abb. 4.4.

Verteilung des Residuenbaums, (c) Expansion inklusive der perturbativen, 5 Iterationen umfassenden, Berechnung der natürlichen Orbitale (NO) und (d) Iteration.

Abb. 4.4 zeigt den Skalierungsplot der auf der T3E-1200 in Jülich durchgeführten Dinitrosoethylen-Rechnung von 32 bis 256 Knoten. Der Plot zeigt ein fast perfektes Skalierungsverhalten von 32 bis 128 Knoten. Erst bei der letzten Verdopplung der Knotenzahl tritt ein signifikanter Verlust an Effizienz auf. Die Vervierfachung der Knotenzahl von 32 auf 128 Knoten führt zu einem Effizienzverlust von 24% für die gesamte Rechnung und von 18% für den MRCI-Schritt allein. Hier wird die Tatsache widerspiegelt, dass wir unsere Bemühungen zur Parallelisierung des Programmpakets auf die MRCI konzentriert haben. Innerhalb des MRCI-Schritts skaliert die Iteration am schlechtesten mit einem Effizienzverlust von 35%. Der Grund hierfür liegt darin, dass es mit abnehmender Anzahl von lokal auf einem Knoten auszuwertenden Matrixelementen immer schwieriger wird, den Load der einzelnen Knoten auszugleichen. Das Loadbalancing erschwert sich, wie zu erwarten, auch dadurch, dass das Programm für diese Rechnung im integralgetriebenen Modus betrieben wurde. Die Größe dieser Rechnung wurde so gewählt, dass auch ein Lauf auf 32 Knoten möglich ist. Vor diesem Hintergrund ist es nicht überraschend, dass die Rechnung auf 256 Knoten nur mit einem Effizienzverlust von 80% durchzuführen ist. Wenn man die Anzahl der selektierten Konfigurationen erhöht, reduziert sich der Verlust an Performance zwischen 128 und 256 Knoten. Das Programm läuft dann aber nicht mehr auf 32 Knoten.

Abb. 4.5 zeigt den Skalierungsplot der Endiin-Rechnung, die auf der IBM SP3/SMP in Karlsruhe durchgeführt wurde. Hier führt eine Vervierfachung der Knotenzahl von 16 auf 64 Knoten lediglich zu einem Verlust von 12% der CPU-Zeit. Am schlechtesten skaliert

wiederum der Iterationsschritt mit einem Effizienzverlust von 34%. Die bessere Performance des Programms auf der IBM SP3/SMP ist vor allem ihrem größeren Hauptspeicher zu verdanken: Mit 1 GB pro Knoten verfügt die IBM SP3/SMP über doppelt soviel Hauptspeicher wie die T3E-1200 (512 MB). Insgesamt zeigen beide Läufe, dass das Programm über eine ausreichende Skalierbarkeit auf den beiden wichtigsten heutzutage verwendeten massiv parallelen Rechnerarchitekturen verfügt.

4.3 Performance-Daten

Nachdem die Skalierbarkeit unseres Verfahrens verifiziert worden ist, soll ein nun ein Eindruck von der Performance des Programms anhand der CPU-Zeiten einiger Rechnungen an wohlbekannten Molekülen gegeben werden. Bei der Interpretation dieser Daten ist jedoch Vorsicht geboten: Die Hardwarelandschaft ändert sich rapide mit der Zeit; veränderte Prozessorgeschwindigkeiten, -Architekturen und Caching-Mechanismen machen einen Vergleich verschiedener Testrechnungen auf verschiedenen Maschinen äußerst schwierig. Dennoch können die Zeitmessungen der Performance dem Anwender einen Eindruck von der Größenordnung der für die Berechnung der Grundzustandsenergien einzelner Moleküle erforderlichen Zeit geben. Die Benchmarkrechnungen wurden auf den unterschiedlichsten Rechnerarchitekturen durchgeführt: Auf der IBM-SP2 (120 MHz RISC2 Prozessoren, 512 MB RAM pro Knoten) und der IBM-SP3/SMP (375 MHz Power3-II Proz., 1 GB RAM) des SCC Karlsruhe, der T3E-600 (300 MHz DEC alpha Proz., 128 MB RAM) und der T3E-1200 (600 MHz DEC alpha Proz., 512 MB RAM) des NIC Jülich und auf dem Linux-PC Cluster des Instituts für Nanotechnologie in Karlsruhe. Keiner der Prozessoren befindet sich am oberen Ende des Leistungsspektrums heutiger Prozessoren. In Tabelle 4.2 sind die CPU-Zeiten für verschiedene Benchmarkrechnungen zusammengefasst. Zu beachten ist, dass hier die Laufzeiten des Programms und nicht die totalen CPU-Zeiten tabelliert sind.

Besondere Aufmerksamkeit soll im folgenden der Rechnung an NO_2 in der cc-pVTZ Basis geschenkt werden, da diese denselben Grundzustand behandelt und dieselbe Anzahl an Basisfunktionen aufweist, wie eine Benchmark Rechnung einiger anderer MRD-CI Implementierungen [25]. Wir haben den Schwellwert so eingestellt, dass ein Hilbertraum gleicher Größe selektiert wurde. Der Vergleich der totalen CPU-Zeit unserer Implementierung von 8100 s mit dem CPU-Zeitbedarf von TDCI-SAF (24000s) und TDCI-PDD (21404s) fällt für unser Programm vorteilhaft aus. In ähnlicher Weise haben wir den Schwellwert in der Benzol-Rechnung eingestellt, um ungefähr die gleiche Anzahl an Determinanten wie in der C_6 Benchmark-Rechnung des DIESEL-CI Programms [26]. Die an unserem Programm gemessene Zeit von 2256 s pro Iteration ist vergleichbar mit 2166 s pro Iteration bei DIESEL-CI. Da die Rechnungen auf verschiedenen Rechnerarchitekturen durchgeführt wurden, lässt sich daraus schwerlich eine Reihenfolge der Effizienz ableiten. Dennoch haben wir die Vergleiche hier präsentiert, um zu zeigen, dass sich der Zeitbedarf unserer

	Basis	N_{det}	N_{sel}	Nodes	CPU (s)	Arch.
O_2^-	aug-cc-pVDZ	1×10^5	5.1×10^4	8	24	a
	aug-cc-pVTZ	1×10^6	3.5×10^5	65	61	a
	aug-cc-pVQZ	3×10^7	7.5×10^5	65	226	a
NO_2	cc-pVDZ	3×10^5	9×10^4	8	80	a
	cc-pVTZ	2×10^6	4×10^5	64	127	a
	cc-pVQZ	9×10^6	5×10^5	64	1225 ^(*)	a
C_6H_6	cc-pVDZ	4×10^7	2×10^5	64	172	a
BO (TQ)	aug-cc-pVTZ	3×10^7	2.2×10^6	64	1440	b
C_6H_4	cc-pVDZ	2.6×10^7	4.2×10^6	64	2361	b
$\text{C}_2\text{H}_2\text{O}_2\text{N}_2$	aug-cc-pVDZ	1.9×10^7	2×10^6	128	1390	c
NO^- (TQ)	aug-cc-pVTZ	2.2×10^8	7.4×10^6	8	25920	d
VF_2	roos-dzp	2.3×10^7	1.1×10^6	64	6568 ^(†)	e

Tabelle 4.2: Performance Daten für verschiedene MRD-CI SD Rechnungen. N_{det} bezeichnet die Größe des Hilbertraums, N_{sel} die Größe des selektierten Subraums. Die nächste Spalte gibt die Anzahl der benutzten Knoten und die vorletzte Spalte die Gesamtlaufzeit des Programms bis zur Konvergenz an. Die Abkürzungen für die Rechnerarchitekturen in der letzten Spalte bedeuten: (a) T3E-600, (b) IBM-SP3/SMP, (c) T3E-1200, (d) Linux-Cluster, (e) IBM-SP2.

(*) Die Rechnung in der cc-pVQZ Basis erfordert die doppelte Anzahl von Iterationen verglichen mit der Rechnung in der cc-pVTZ Basis. Der unverhältnismäßige Anstieg der CPU-Zeit resultiert hieraus.

† In der VF_2 Rechnung wurden neben dem Grundzustand drei weitere angeregte Zustände berechnet.

Implementierung mit der moderner Einprozessorimplementierungen der zustandsselektierenden MRCI messen kann und keine CPU-Zeit durch die Verwendung einer parallelen Rechnerarchitektur verschwendet wird.

Kapitel 5

Anwendungen

5.1 Elektronenaffinität von BO, NO und O₂

Die Berechnung genauer Dissoziationsenergien, Elektronenaffinitäten und Ionisierungsenergien ist wichtig für die Interpretation vieler chemischer Phänomene. Bei der Berechnung von Dissoziationsenergien wurden in letzter Zeit große Fortschritte erzielt [27, 28], hingegen ist die Berechnung der Elektronenaffinität (EA) von schwach gebundenen Anionen auch für die Methoden der heutigen theoretischen Chemie eine Herausforderung. Insbesondere hat sich die quantitativ korrekte Behandlung von Oxiden als problematisch erwiesen. Nur mit hochgenauen Korrelationsmethoden lässt sich hier eine Übereinstimmung der theoretischen Resultate mit dem Experiment erzielen.

Die erste Messung der Elektronenaffinität von BO stammt aus dem Jahr 1970: Jensen *et al.* [29] erhielten hier 2,5 eV als untere Grenze der Elektronenaffinität. Im folgenden Jahr haben Srivastava, Uy und Farber [30] $3,12 \pm 0,1$ eV gemessen. Zahlreiche theoretische Studien mit Ergebnissen zwischen 2,17 eV und 2,79 eV ließen jedoch Zweifel an der Richtigkeit des experimentellen Messwerts aufkommen. Am verlässlichsten erscheint hier eine CCSD(T)-Rechnung (*Cuppled-Cluster* Methode) von Rienstra *et al.* [31] in einer augmentierten cc-pVQZ Basis. Der errechnete Wert von 2,52 eV ist mittlerweile auch experimentell bestätigt worden [32]. Unter den veröffentlichten theoretischen Arbeiten waren auch MRCI und MR-ACPF Rechnungen [33], welche aber den experimentellen Wert um mehr als 0,3 eV verfehlten, auch in einer doppelt-augmentierten cc-pV5Z Basis.

Stickstoffmonoxid (NO) ist ein schwierig zu behandelndes System, da das Anion hier nur sehr schwach (0,026 eV [34]) gebunden ist. Klassisch liegt die Potentialenergieoberfläche von NO sogar unter der des NO⁻-Anions. Erst die ersten Schwingungsniveaus weisen die umgekehrte Reihenfolge auf. NO⁻ wird also nur durch die Nullpunktsbewegung der Elektronen stabilisiert. Von akkuraten CCSD(T)-Rechnungen wird dieses Resultat quantitativ fast richtig wiedergegeben [31, 33].

Auch die Berechnung der Elektronenaffinität des Sauerstoffmoleküls (0,42 eV [34]) hat sich als Herausforderung für die theoretische Chemie erwiesen [35]: Auf dem Niveau einer CAS-SCF Rechnung wird die adiabatische Elektronenaffinität mit falschem Vorzeichen vorausgesagt, auch im Basissatzlimit. Eine ausführliche Studie [35] hat gezeigt, dass stark unterschiedliche dynamische Korrelationseffekte in O_2 und O_2^- für diese Diskrepanz verantwortlich sind. Dennoch sind MRCI-SD Rechnungen auch unter Verwendung von augmentierten cc-pVQZ Basissätzen nicht in der Lage, das richtige Vorzeichen zu reproduzieren. Man erhält eine semiquantitative Übereinstimmung zwischen Experiment und Theorie, wenn man die auf den Multireferenzfall generalisierte Davidson-Korrektur auf die MRCI-Resultate anwendet, um den Effekt der höheren Anregungen abzuschätzen [14].

Die bisherigen Resultate zeigen, dass ein Zusammenspiel von starken dynamischen als auch von statischen Korrelationseffekten eine Rolle beim Übergang von O_2 zu O_2^- spielen. In CCSD(T)-Rechnungen, bei denen durch das perturbative Hinzufügen von Dreifachanregungen statische und dynamische Korrelationen ausgewogen behandelt werden, erhält man quantitativ korrekte Resultate. Reine CCSD- und MRCI-Rechnungen versagen bei der Voraussage der Elektronenaffinität von Sauerstoff. Hierfür sind einerseits der Multireferenzcharakter der Wellenfunktion [35], andererseits große Extensivitätskorrekturen [33] verantwortlich gemacht worden. Dadurch motiviert, haben wir eine systematische Untersuchung durchgeführt, in der wir über die Standard-MRCI hinausgehend selektiv auch höhere Anregungen der Referenzkonfigurationen berücksichtigt haben. Für NO und BO haben wir eine exzellente quantitative Übereinstimmung zwischen unserer Rechnung und den besten verfügbaren experimentellen Werten gefunden. Für Sauerstoff haben wir die Elektronenaffinität zu 0,39 eV berechnet, was immer noch in guter Übereinstimmung mit dem experimentellen Ergebnis ist.

Da die Anzahl der mehr als doppelt angeregten Konfigurationen sehr schnell mit der Systemgröße wächst, sind sowohl Full-CI, als auch MRCI-SDTQ Rechnungen praktisch nicht durchführbar. Die zustandsselektierende Variante bietet daher den vielleicht einzig gangbaren Kompromiss zwischen Rechenaufwand und Genauigkeit, um den Effekt der höheren Anregungen in CI-Rechnungen zu untersuchen.

Die Konfigurationen eines Full-CI Hilbertraums sind stark unterschiedlich gewichtet. Das Gewicht einer speziellen Konfiguration aus dem Hilbertraum sinkt mit ihrem Anregungsniveau, also der Anzahl der externen Orbitale. Aber auch innerhalb der Konfigurationen eines Anregungsniveaus gibt es eine Hierarchie: So entspricht es der chemischen Intuition, dass Vierfachanregungen, an denen nur Orbitale des 3s- oder 3p-Typs beteiligt sind, wichtiger sind als solche, bei denen ausschließlich d- oder f-Orbitale beteiligt sind. In großen Basissätzen kann es jedoch passieren, dass die wichtigsten Drei- und Vierfachanregungen genauso wichtig werden, wie Ein- und Zweifachanregungen in Orbitale mit sehr großer Haupt- oder Drehimpulsquantenzahl (Abb. 5.1). Orbitale mit großer Haupt- oder Drehimpulsquantenzahl werden benötigt, um eine gute Auflösung der Energie auf MRCI-SD Niveau zu erhalten; Drei- und Vierfachanregungen in solche Orbitale hingegen tragen kaum zur Gesamtwellenfunktion bei.

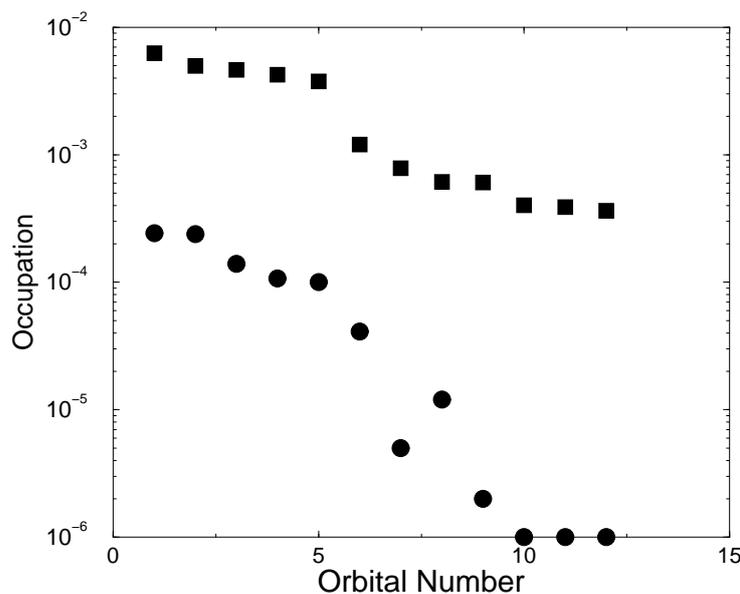


Abbildung 5.1: Beiträge der einzelnen Orbitale (von a_g -Symmetrie) zu den diagonalen Matrixelementen der Einteilchen-Dichtematrix, die von Ein- und Zweifachanregungen (Quadrate) oder Drei- und Vierfachanregungen (Kreise) ausgehen. Die Daten wurden in einer CI-SDTQ Rechnung in einer cc-pVDZ und in der experimentellen Geometrie berechnet.

Alle Rechnungen wurden in einer augmentierten cc-pVTZ von Dunning *et al.* [17, 36] durchgeführt. Frühere Untersuchungen [37, 33] haben gezeigt, dass Basissätze auf *triple zeta*-Niveau ausreichen, um quantitativ korrekte Resultate für die hier untersuchten Oxide zu liefern. Für Sauerstoff haben wir zusätzlich Rechnungen in einer cc-pVQZ Basis durchgeführt, um die Konvergenz zu testen. In allen Rechnungen wurden approximative natürliche Orbitale verwendet, die mit der BSR-Methode generiert wurden. Die MRD-CI Rechnungen wurden für NO und BO in C_{2h} -Symmetrie, für Sauerstoff in D_{2h} -Symmetrie durchgeführt. Die 1s Elektronen der beteiligten Atome wurden eingefroren und in allen Rechnungen doppelt besetzt gehalten. Für die SD-Rechnungen wurde ein Raum aus 8 aktiven Orbitalen verwendet, entsprechend den 2s- und 2p-Orbitalen der beteiligten Atome. Die Rechnungen wurden für verschiedene Selektionsschwellwerte λ durchgeführt und für $\lambda \rightarrow 0$ extrapoliert. Für die SD-Rechnungen wurde keine Referenzselektion durchgeführt. Die Anzahl der Referenzen betrug 392 für NO, 120 für NO^- , 1000 für BO und 792 für BO^- . Für das symmetrischere Sauerstoffmolekül kam man mit deutlich weniger Referenzen aus: 96 für O_2 und 28 für O_2^- . Die Größe der Hilberträume betrug 2,8 Mio bzw. 1,2 Mio für Sauerstoff und schwankte zwischen 10 und 13 Mio für NO bzw. BO. Selektiert wurden zwischen 0,1 Mio und 10 Mio Konfigurationen.

Eine Interpolation, wie wir sie in Abschnitt 2.1 vorgeschlagen haben, hat sich in diesem Fall aufgrund der von der Rechnung geforderten extrem hohen Genauigkeit als schwierig erwiesen. In Abb. 5.2 sind die variationelle MRCI-Energie E_V , die totale MRCI-Energie

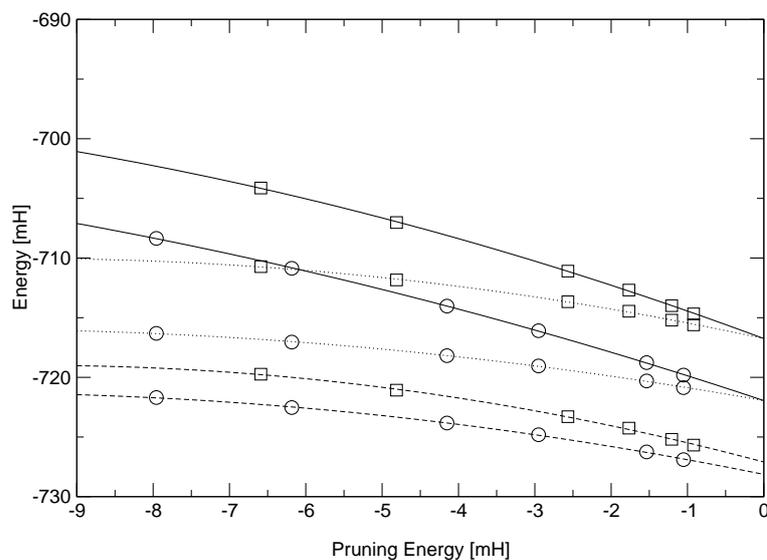


Abbildung 5.2: Extrapolation der MRCI-Energie gegen die perturbative Energiekorrektur für eine MRCI(TQ)-Berechnung von NO und NO^- . Aufgetragen sind die variationelle MRCI-Energie (durchgezogen), die totale MRCI-Energie (gepunktet) und die Davidson-korrigierte MRCI-Energie (gestrichelt) von NO (Kreise) und NO^- (Quadrate) gegen die perturbative Energiekorrektur.

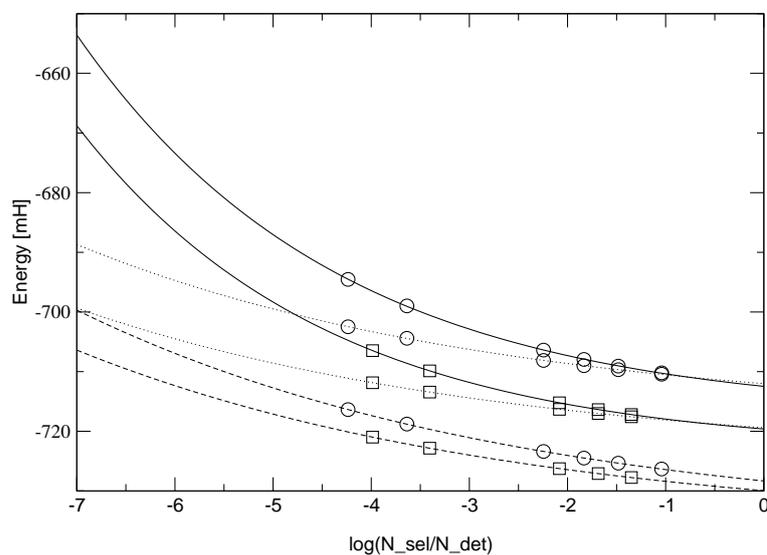


Abbildung 5.3: Extrapolation der MRCI-Energie gegen das Verhältnis des selektierten Subraums zur Größe des Hilbertraums für eine MRCI(TQ)-Berechnung der Grundzustandsenergien von NO und NO^- . Aufgetragen sind die variationelle MRCI-Energie (durchgezogen), die totale MRCI-Energie (gepunktet) und die Davidson-korrigierte MRCI-Energie (gestrichelt) von NO (Kreise) und NO^- (Quadrate) gegen $\log(N_{\text{sel}}/N_{\text{det}})$.

$E_T = E_V + E_P$ und die Davidson-korrigierte MRCI-Energie E_D für eine Rechnung an NO und NO^- gegen die perturbative Energiekorrektur E_P aufgetragen. Die Energien E_V , E_T und E_D hängen nicht mehr linear von E_P ab. Dass E_T mit kleiner werdendem E_P fällt liegt daran, dass hier Wechselwirkungen zwischen den angeregten Konfigurationen an Bedeutung gewinnen, die in der Störungstheorie nicht mit berücksichtigt werden können.

Man kann diesen Effekt jedoch berücksichtigen, in dem man nicht gegen die perturbative Energiekorrektur, sondern gegen den Anteil des selektierten Subraums am gesamten Hilbertraum extrapoliert. In Abb. 5.3 sind die berechneten MRCI-Energien gegen

$$\eta = \log \left(\frac{N_{\text{sel}}}{N_{\text{det}}} \right) \quad (5.1)$$

aufgetragen und durch

$$E(\eta) = a + b \exp(c\eta) \quad (5.2)$$

extrapoliert. Die Fehler der durch Gleichung (5.2) bestimmten Energien zu den berechneten MRCI-Energien betrug Bruchteile von milli-Hartree. In den SD-Rechnungen, in denen wahlweise auch alle Konfigurationen selektiert werden konnten, konnte der Wert bei $\lambda = 0$ durch die Extrapolation auf 0,05 mH reproduziert werden.

Um nun den Effekt der höheren Anregungen zu untersuchen, haben wir die Größe des aktiven Raums verdoppelt, so dass nun auch die 3s3p-Schale Teil des Valenzraums wird. Ausgehend von diesem Referenzraum sind, wie in einer gewöhnlichen MRCI, Ein- und Zweifachanregungen in die verbleibenden externen Orbitale erlaubt. Die Größe des Referenzraums wuchs durch die Verdoppelung des aktiven Raums auf über 150000 Konfigurationen in allen Rechnungen. Wir haben daher eine Referenzselektion mit Schwellwerten $\lambda_r = 10^{-2}, 3 \cdot 10^{-3}, 10^{-3}, 7 \cdot 10^{-4}, 3 \cdot 10^{-4}$ durchgeführt. In allen Rechnungen war $\lambda_r = 10^{-3}$ ausreichend, eine Verkleinerung des Schwellwerts hatte keinen Effekt auf die berechneten Elektronenaffinitäten. Die Referenzselektion wurde iterativ in sechs Schleifen durchgeführt. In allen Fällen war die Referenzselektion spätestens nach der dritten Schleife konvergiert.

Der Hilbertraum einer typischen TQ-Rechnung für O_2 hatte 2100 Referenzen, die zu mehr als 6×10^7 Konfigurationen führten. Von diesen wurden bis zu $1,2 \times 10^7$ für den variationellen Subraum selektiert. Auf 64 Knoten der IBM-SP/SMP des SCC in Karlsruhe, brauchte die Rechnung 3,5 Stunden bis zur Konvergenz (inkl. 10 Schleifen für die Generierung der BSR-NOs, Vierindextransformation, etc.). Eine vergleichbare Rechnung für BO^- ($N_{\text{ref}} = 5048$, $N_{\text{det}} = 2,2 \times 10^8$, $N_{\text{sel}} = 7,4 \times 10^6$) benötigte 7,2 Stunden auf 8 Knoten eines massiv parallelen PC-Clusters (1,8 GHz Dual-Athlon mit SCI), was die gute Performance unseres Programms auch auf dieser relativ preiswerten Rechnerarchitektur demonstriert.

Motiviert von der bereits erwähnten CCSD(T)-Rechnung von Rienstra *et al.* aus dem Jahr 1997 [31], haben Wentholt *et al.* [32] im selben Jahr mit einem 315 nm Laser das Photoelektronenspektrum von BO gemessen und erhielten für die Elektronenaffinität $2,508 \pm 0,008$ eV. Die bislang bekannten MRCI Rechnungen lieferten auch in einer

	r_0 [Å]		ω [cm ⁻¹]		D_e [kcal/mol]	
	calc.	exp.	calc.	exp.	calc.	exp.
BO	1.216	1.205 [38]	1868	1875 [32]	188.4	194.9 [38]
BO ⁻	1.251	1.236 [32]	1709	1655 [32]	214.92	218.8 [38]
NO	1.158	1.151	1875	1904	146.2	152.4
NO ⁻	1.277	1.271	1347	1285	117.1	118.5
O ₂	1.215	1.208	1583	1580	117.5	120.1
O ₂ ⁻	1.364	1.341	1103	1089	93.2	95.9

Tabelle 5.1: Spektroskopische Konstanten von BO, NO und O₂ und ihrer Anionen berechnet mit einer (2s2p) CAS MR-ACPF in einer augmentierten cc-pVTZ Basis im Vergleich mit dem Experiment. Soweit nicht anders angezeigt, stammen die experimentellen Daten aus Ref. [39].

doppelt-augmentierten cc-pV5Z Basis einen Wert von 2,18 eV, wiesen also einen Fehler von fast 0,3 eV auf.

Wir haben für BO und BO⁻ zunächst die Bindungslänge im Grundzustand, die Schwingungsfrequenz in harmonischer Näherung und die Dissoziationsenergien mit einer MR-ACPF-Rechnung in einer cc-pVTZ Basis bestimmt. Beachtenswert ist, dass diese relativ unaufwändige Rechnung (20 Minuten auf *einem* 1,7 GHz Pentium III-Prozessor) einen bis auf 0,03 eV genauen Wert der Elektronenaffinität von BO liefert. Zur Bestimmung der Bindungslänge und der Schwingungsfrequenz haben wir die Grundzustandsenergien für verschiedene Bindungslängen in der Nähe der experimentellen Geometrie berechnet und durch ein Morse-Potential der Form

$$E(r) = E_0 + D_e (1 - \exp(-\gamma(r - r_0)))^2 \quad (5.3)$$

interpoliert (Abb. 5.4). Hieraus lässt sich dann der Gleichgewichtsabstand r_0 und die harmonische Frequenz

$$\omega = \frac{1}{2\pi c} \sqrt{\frac{k}{m}}, \quad (5.4)$$

mit

$$k = \left. \frac{d^2 E(r)}{dr^2} \right|_{r_0} \quad (5.5)$$

bestimmen. Die berechneten Bindungslängen, Frequenzen und Dissoziationsenergien sind in Tabelle 5.1 zusammengefasst und zeigen eine exzellente Übereinstimmung mit den experimentellen Ergebnissen: Die berechneten Bindungslängen sind 0,011 Å zu groß für BO und 0,015 Å zu groß für das Anion. Die berechneten Frequenzen stimmen mit den experimentellen Werten bis auf 7 Wellenzahlen bei BO bzw. 54 Wellenzahlen bei BO⁻ überein.

Die Dissoziationsenergie wurde berechnet, in dem der Limes $r_{\text{BO}} \rightarrow \infty$ gebildet wurde. Das Anion kann dabei prinzipiell auf zwei Weisen dissoziieren: Entweder zu B +

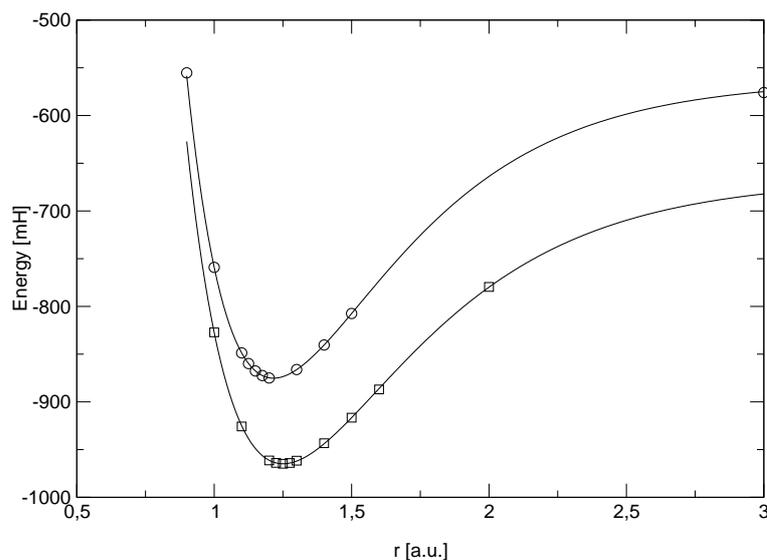


Abbildung 5.4: Potentialenergieoberflächen von BO und BO^- . Die Energien für BO (Kreise) und BO^- (Quadrate) wurden mit der MR-ACPF-Methode in einer aug-cc-pVTZ Basis für verschiedene Bindungslängen berechnet und in der Nähe des Minimums durch ein Morse-Potential interpoliert.

O^- oder zu $\text{B}^- + \text{O}$. Da das Sauerstoffatom die größere Elektronenaffinität hat, ist die erste Möglichkeit zu erwarten. Das spiegelt sich auch in unserer Rechnung wieder: Die Dissoziationsenergie für die zweite Konfiguration liegt ca. 30 kcal/mol höher. Das die Dissoziationsenergien etwas niedriger als die experimentellen Werte sind, ist nicht weiter verwunderlich: Chemische Genauigkeit (≤ 1 kcal/mol) für die Dissoziationsenergien wird in der Regel erst im Basissatzlimit erreicht, also bei Basissätzen von *quadruple zeta* oder *quintuple zeta* Qualität. Es handelt sich hier um ein generelles Problem der heutzutage verwendeten quantenchemischen Methoden.

Während man in einer MR-ACPF Rechnung bereits einen Wert von 2,52 eV für die Elektronenaffinität von BO erhält (Tabelle 5.2), weist eine gewöhnliche MRCI mit einer EA von 2,31 eV einen Fehler von mehr als 0,2 eV auf. Dieser wird durch die explizite Hinzunahme von höheren Anregungen auf 0,08 eV reduziert; man erhält einen Wert von 2,47 eV. Gleichzeitig nimmt der Effekt der Extensivitätskorrekturen bei expliziter Hinzunahme höherer Anregungen stark ab. So werden die berechneten Energien in der gewöhnlichen MRCI durch die Davidson-Korrektur um 9 mH (BO) bzw. 17 mH (BO^-) abgesenkt. In den TQ-Rechnungen reduziert sich dieser Effekt auf 2 bzw. 4 mH. In der MR-ACPF(TQ) Rechnung erhält man für die Elektronenaffinität einen Wert von 2,55 eV, also perfekte Übereinstimmung mit dem Experiment.

NO^- ist eines der am schwächsten gebundenen zweiatomigen Anionen (EA=0,026 eV) [34], die Berechnung seiner Elektronenaffinität ist daher ein guter Test für unser Verfahren. Aktuelle CCSD(T)-Studien [31, 33] war es unter Verwendung einer aug-cc-pVTZ

	neutral	anion	EA
MRCI	-99.86700	-99.95173	2.31
MRCI+Dav.	-99.87600	-99.96807	2.51
MR-ACPF	-99.87477	-99.96740	2.52
MRCI(TQ)	-99.87270	-99.96351	2.47
MRCI(TQ)+Dav.	-99.87484	-99.96755	2.52
MR-ACPF(TQ)	-99.87568	-99.96984	2.55

Tabelle 5.2: Totale Energien (in H) von BO und BO⁻ und adiabatische Elektronenaffinitäten (EA) (in eV) berechnet mit MRD-CI, MRD-CI + Davidson-Korrektur oder zustandsselektierender MR-ACPF, mit und ohne explizite Behandlung höherer Anregungen (TQ). Es wurden die Geometrien aus Tabelle 5.1 verwendet.

	neutral	anion	EA (el)	EA (+ZPC)
MRCI	-129.70995	-129.69293	-0.4631	-0.4286
MRCI+Dav.	-129.72981	-129.72511	-0.1278	-0.0933
MR-ACPF	-129.72984	-129.72886	-0.0266	0.0079
MRCI(TQ)	-129.72416	-129.71965	-0.1222	-0.0877
MRCI(TQ)+Dav.	-129.73071	-129.73069	-0.0005	0.0340
MR-ACPF(TQ)	-129.73028	-129.72957	-0.0193	0.0152

Tabelle 5.3: Totale Energien (in H) von NO und NO⁻ und adiabatische Elektronenaffinitäten (EA) (in eV) berechnet mit MRD-CI, MRD-CI + Davidson-Korrektur oder zustandsselektierender MR-ACPF, mit und ohne explizite Behandlung höherer Anregungen (TQ). Es wurden die Geometrien aus Tabelle 5.1 verwendet.

		neutral	anion	EA
MRCI	aug-cc-pVTZ	-150.11644	-150.11270	-0.10
MRCI+Dav.	aug-cc-pVTZ	-150.14254	-150.14855	0.16
MR-ACPF	aug-cc-pVTZ	-150.14230	-150.15174	0.26
MRCI+(SC) ²	aug-cc-pVTZ	-150.12958	-150.13805	0.23
MRCI	aug-cc-pVQZ	-150.15153	-150.14805	-0.09
MRCI+Dav.	aug-cc-pVQZ	-150.17938	-150.18591	0.18
MR-ACPF	aug-cc-pVQZ	-150.17924	-150.18931	0.27
MRCI(TQ)	aug-cc-pVTZ	-150.13579	-150.14992	0.38
MRCI(TQ)+Dav.	aug-cc-pVTZ	-150.14475	-150.15818	0.36
MR-ACPF(TQ)	aug-cc-pVTZ	-150.14441	-150.15762	0.36

Tabelle 5.4: Totale Energien (in H) von O_2 und O_2^- und adiabatische Elektronenaffinitäten (EA) (in eV) berechnet mit MRD-CI, MRD-CI + Davidson-Korrektur oder zustandsselektierender MR-ACPF, mit und ohne explizite Behandlung höherer Anregungen (TQ). Es wurden die Geometrien aus Tabelle 5.1 verwendet.

Basis möglich, den experimentellen Wert bis auf 0,01 eV zu reproduzieren, wenn man die entsprechenden Korrekturen für die Nullpunktsenergie (ZPC) berücksichtigt. MR-ACPF Rechnungen in derselben Basis und einem (2s2p)-CAS Raum konnten die Bindungslängen von NO und NO^- bis auf 0,01 Å, verglichen mit dem Experiment und früheren Rechnungen, reproduzieren. Die (harmonischen) Frequenzen von NO und seinem Anion wurden zu 1875 und 1347 cm^{-1} berechnet; die letztere fällt innerhalb der Fehlerbalken mit dem experimentellen Wert (1285–1370 cm^{-1}) zusammen. SD-MRCI Rechnungen in einem (2s2p)-CAS Raum liefern für die EA einen Fehler von fast 0,5 eV (Tab. 5.3). Dieser wird durch die Davidson-Korrektur auf 0,12 eV reduziert. Die zwar approximativ, jedoch selbstkonsistente Behandlung der höheren Anregungen auf MR-ACPF-Niveau vermag, den Fehler sogar auf 0,01 eV zu reduzieren. Der Fehler der Extensivitätskorrekturen ist nicht überraschend, schließlich werden die Energien von NO und NO^- durch sie um mehr als 0,5 eV abgesenkt. Diese Absenkung ist damit um mehr als eine Größenordnung größer als der gesucht Effekt.

Die explizite Hinzunahme von TQ-Anregungen im Hilbertraum führt zu einer besseren Approximation der zugrundeliegenden Referenzwellenfunktion, die im verwendeten (2s2p3s3p)-Referenzraum FCI Charakter aufweist. Der Fehler der MRCI-Rechnung wird hier auf 0,1 eV reduziert. Die Verwendung von Extensivitätskorrekturen führt auf Werte von 0,034 eV für MRD-CI(TQ)+Dav und 0,015 eV für MR-ACPF(TQ), die das experimentelle Ergebnis einrahmen.

Schließlich haben wir dieselbe Strategie zur Berechnung der Elektronenaffinität von O_2 verwendet. Wiederum (s. Tabelle 5.1) sind die berechneten Bindungslängen und harmonischen Frequenzen in exzellenter Übereinstimmung mit dem Experiment. Wir finden frühere Studien [35] bestätigt, da (2s2p) CAS-Rechnungen mit -0,10 bzw. -0,09 eV nicht das richtige Vorzeichen der Elektronenaffinität liefern. Mit der Hinzunahme von Exten-

sivitätskorrekturen erhält man zwar das richtige Vorzeichen, jedoch keine quantitative Übereinstimmung mit dem experimentellen Wert. So weist die MR-ACPF immer noch einen Fehler von fast 0,2 eV auf. Ein Vergleich der Rechnungen in aug-cc-pVTZ und aug-cc-pVQZ Basissätzen zeigt, dass es sich hierbei nicht um einen Basissatzeffekt handelt. Mit der expliziten Zunahme von höheren Anregungen verändert sich die Elektronenaffinität drastisch auf Werte von 0,38 eV, 0,36 eV und 0,36 eV für die MRD-CI(TQ), MRD-CI(TQ)+Dav und MR-ACPF(TQ). Addiert man eine Nullpunktskorrektur von 0,03 eV erhält man einen Wert von 0,39 eV, der im Vergleich zum durch Photoelektronenspektroskopie an O_2^- gemessenen Wert von $0,452 \pm 0,007$ eV einen Fehler von 0,06 eV aufweist.

Insgesamt korrigieren unsere Rechnungen die seit langem bekannte Schwäche der MR-CI-Methode, die elektronische Struktur anionischer Oxide zu beschreiben. Sie zeigen auch den Mechanismus dieses Fehlers auf: In sauerstoffhaltigen Molekülen tragen höhere Anregungen in Nicht-Valenzschalen wesentlich zu differentiellen Effekten in der elektronischen Struktur bei. Die explizite Behandlung dieser Anregungen, wie sie beispielsweise auch in den perturbativen Tripel-Korrekturen in der CCSD(T)-Methode enthalten ist, ist notwendig, um qualitativ und quantitativ richtige Resultate zu erhalten. Die von uns vorgeschlagene Strategie, in der die wichtigsten höheren Anregungen selektiv hinzugenommen werden, eröffnet einen gangbaren Kompromiss, um diese Effekte zu behandeln.

5.2 Endiine

Endiine werden seit einiger Zeit aufgrund ihrer Fähigkeit in einer Bergmanschen Ringschlussreaktion zu para-didehydro-Benzol Derivaten zu reagieren, intensiv untersucht. In dieser Formation besitzen Endiine die Fähigkeit, zelluläre DNA aufzulösen und so einen Tod der Zelle zu verursachen. Falls es gelingt, diese Eigenschaft in kranken oder krebserkrankten Zellen zu aktivieren, eröffnet das die Möglichkeit zur Entwicklung zellspezifischer Medikamente. Das Ziel der aktuellen Forschung ist es, eine Verbindung zu synthetisieren, die in Krebszellen spontan die Ringschlussreaktion eingeht und dann die befallenen Zellen vernichtet. Um nun die Effektivität einer speziellen Verbindung vorherzusagen, ist es sowohl nötig, die elektronische Struktur von Edukt und Produkt zu verstehen, als auch die Höhe der Barriere zwischen den beiden Strukturen zu kennen. Es hat sich gezeigt, dass eine quantenchemische Beschreibung der Ringschlussreaktion schwierig ist, da signifikante Unterschiede in den elektronischen Strukturen von Edukt und Produkt bestehen.

Wir haben deshalb eine MRD-CI Untersuchung der Ringschlussreaktion am einfachsten Mitglied der Endiin-Familie durchgeführt. Auf SCF-Niveau ist diese Reaktion, im Gegensatz zum experimentellen Befund, stark endotherm. Hier zeigt sich die Wichtigkeit der dynamischen Korrelationseffekte für die korrekte Behandlung dieser Substanzen. CCSD Rechnungen [42, 43] schaffen es, die Abweichung vom Experiment signifikant zu reduzieren, man erhält jedoch keine quantitative Übereinstimmung mit den experimentellen Werten. Erst CCSD(T)-Rechnungen, in denen teilweise auch Dreifachanregungen und damit statische Korrelationseffekte behandelt werden, liefern auch eine quantitative

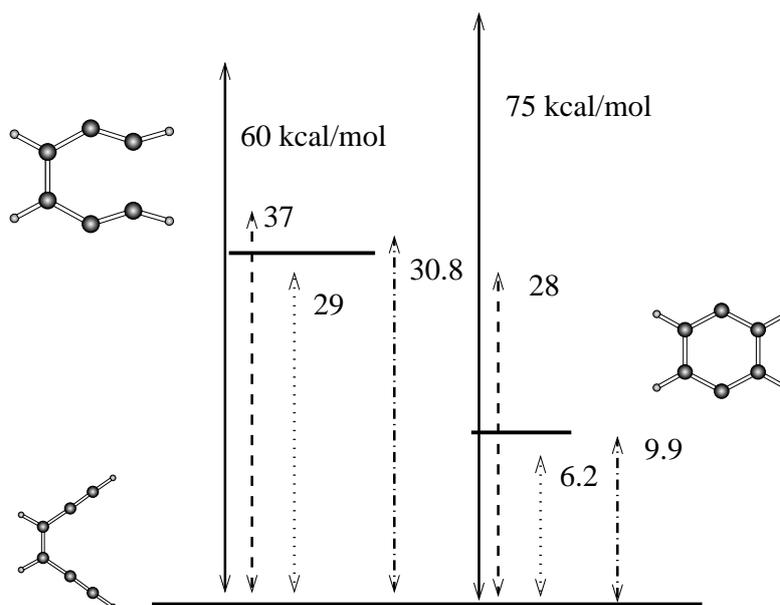


Abbildung 5.5: Schematische Darstellung der (rein elektronischen) Energien der Ringschlussreaktion von Endiin zu para-didehydro-Benzol. Die Rechnungen wurden in einer cc-pVDZ-Basis mit verschiedenen Methoden durchgeführt (SCF: —, CCSD: --, CCSD(T): ···, MRD-CI: - · -). Alle Energien sind in kcal/mol relativ zum Edukt angegeben.

Übereinstimmung zwischen Experiment und Theorie. Diese Beobachtung zeigt, dass für die Berechnung der Energien der Ringschlussreaktion eine ausgewogene Behandlung sowohl dynamischer als auch statischer Korrelationseffekte notwendig ist. Bedenkt man nun das Skalierungsverhalten von CCSD(T)-Rechnungen, welches proportional zu N^7 ist, so wirft das Schwierigkeiten für die Anwendung dieser Methode auf größere Systeme oder größere Basissätze auf.

Unsere Untersuchung war durch zwei Beobachtungen motiviert: Eine frühere MRCI-Studie [44] erreichte nicht einmal qualitative Übereinstimmung mit dem Experiment. Vermutlich reichte die Größe der verwendeten (unpolarisierten) DZ-Basis oder die Anzahl der verwendeten Referenzen nicht aus. Außerdem findet man in der Literatur keine Studien, in denen Basissätze größer als cc-pVDZ verwendet wurden. Unser Ziel war es daher, mit der MRD-CI-Methode für eine cc-pVDZ Basis die Übereinstimmung mit dem Experiment zu überprüfen, um dann die Methode auf größere Basissätze anzuwenden. So kann sichergestellt werden, dass die quantitativ richtigen Werte auf cc-pVDZ Niveau kein Zufall sind.

Wir haben MRD-CI Rechnungen an Produkt, Edukt und Übergangszustand in cc-pVDZ bzw. cc-pVTZ Basissätze von Dunning *et al.* durchgeführt [17, 36]. Dazu haben wir Geometrien verwendet, die mit Dichtefunktionalrechnungen (DFT) optimiert wurden [43]. Die Rechnungen wurden in approximativen natürlichen Orbitalen durchgeführt, die mit der BSR-Methode generiert wurden. Abb. 5.6 zeigt die Konvergenz der variationellen und

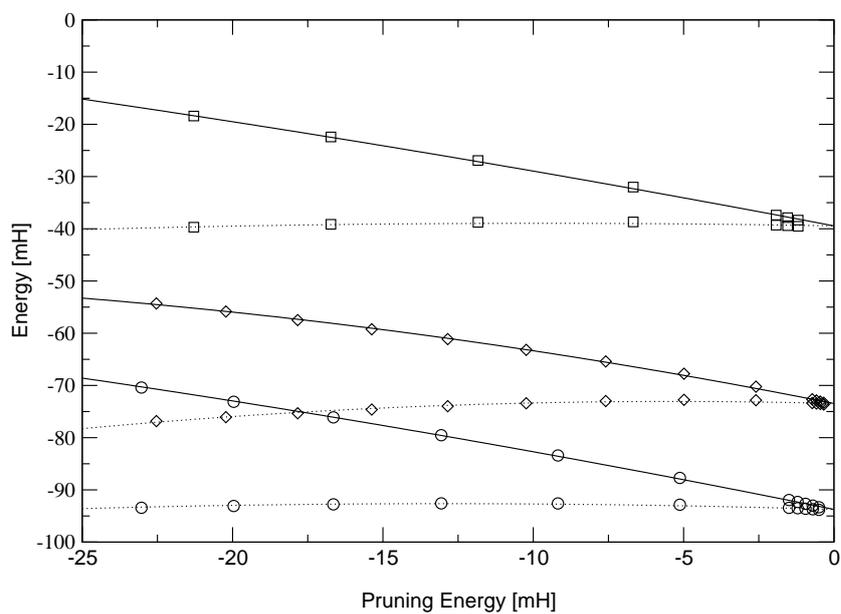


Abbildung 5.6: Konvergenz der variationellen (durchgezogen) und der totalen (gepunktet) MRD-CI Energien von Edukt (Rauten), Übergangszustand (Quadrate) und Produkt (Kreise) wie im Text beschrieben. Auf der vertikalen Achse sind die Energien in mH relativ zu $-230,0$ H angegeben. Die horizontale Achse zeigt die Größe der perturbativen Energiekorrektur (in mH).

	cc-pVDZ				cc-pVTZ				Exp
	MRD-CI	Diff	CI + Q	Diff	MRD-CI	Diff	CI + Q	Diff	Diff
EN	-93.7		-196.6		-250.8		-378.3	—	—
TS	-39.4	34.1	-137.8	30.8	-200.0	31.9	-333.2	28.3	30.1±0.5
PB	-73.5	12.72	-171.6	9.5	-234.9	9.9	-366.9	7.2	7.8 ± 1.0

Tabelle 5.5: Absolute Energien (in mH relativ zu -230,0 H) und relative elektronische Energien (in kcal/mol) des Edukts (EN), Übergangszustands (TS) und Produkts (PB) der Ringschlussreaktion von Endiin zu para-dihydro-Benzol, berechnet mit der MRD-CI und der Davidson-korrigierten MRD-CI (CI+Q). Die experimentellen Energien stammen von Roth *et al.* [40]. Sie wurden um die Nullpunktsenergien korrigiert [41].

Davidson-korrigierten MRD-CI Energie für alle drei Zustände in der cc-pVDZ Basis mit dem Selektionsschwellwert. Der Grenzwert einer verschwindenden perturbativen Energiekorrektur E_P entspricht einer nicht selektierenden MRCI. Die Linien zeigen die lineare Regressionen, die wir benutzt haben, um die Energien für $E_P \rightarrow 0$ zu extrapolieren. Bei einem gegebenen Selektionsschwellwert λ können die perturbativen Energiekorrekturen für die einzelnen Isomere sehr unterschiedlich ausfallen. Das bedeutet, dass die bei gegebenem λ berechneten Energien nicht miteinander vergleichbar sind. Stattdessen ist es notwendig, die Energien für $E_P \rightarrow 0$ zu extrapolieren.

In Tabelle 5.5 sind die absoluten elektronischen Energien von Edukt, Übergangszustand und Produkt in milli-Hartree und die Differenzen von Übergangszustand und Produkt zum Edukt in kcal/mol aufgelistet. Die experimentellen Werte stammen von Roth *et al.*. Sie wurden um die Energien am Temperaturnullpunkt korrigiert. Da für das Edukt und den Übergangszustand keine experimentellen Schwingungsfrequenzen bekannt sind, haben wir hierzu auf Nullpunktskorrekturen zurückgegriffen, die mit Dichtefunktionaltheorie (DFT) berechnet wurden [41]. Die Davidson-korrigierten MRD-CI Energien zeigen sowohl auf *double zeta*- als auch auf *triple zeta*-Niveau eine sehr gute Übereinstimmung mit den experimentellen Werten. Die in der cc-pVTZ Basis berechnete Reaktionsenthalpie von 7,2 kcal/mol fällt innerhalb der Fehlerbalken mit dem experimentellen Wert von $7,8 \pm 1,0$ kcal/mol zusammen. Die Aktivierungsenergie zwischen Edukt und Übergangszustand wird mit 28,3 kcal/mol um ca. 2 kcal/mol unterschätzt.

5.3 VF₂ und VCl₂

Seit langem ist bekannt, dass *ab initio* Berechnungen der elektronischen Struktur von Übergangsmetallverbindungen schwierig sind. Der Grund hierfür liegt darin, dass die Elektronen im Grundzustand und in den tiefliegenden angeregten Zuständen stark korreliert sind. Das Problem wird zusätzlich dadurch kompliziert, dass es sehr viele niedrigliegende elektronische Zustände gibt, die energetisch eng benachbart sind. Beispielsweise haben

$D_{\infty h}$	D_{2h}	QCI
Σ_g^+	A_g	A
Σ_u^+	B_{1u}	B
Σ_g^-	B_{1g}	G
Σ_u^-	A_u	H
Π_g	B_{2g}, B_{3g}	F, D
Π_u	B_{2u}, B_{3u}	C, E
Δ_g	A_g, B_{1g}	A, G
Δ_u	A_u, B_{1u}	H, B
Φ_g	B_{2g}, B_{3g}	F, D
Φ_u	B_{2u}, B_{3u}	C, E
Γ_g	A_g, B_{1g}	A, G
Γ_u	A_u, B_{1u}	H, B

Tabelle 5.6: Zuordnung der ersten Symmetrieklassen der $D_{\infty h}$ Gruppe zu den Symmetrieklassen der Gruppe D_{2h} . Die letzte Spalte gibt die Bezeichnung der Symmetrien innerhalb des QCI-Programms an, die sich ergibt, wenn man das Molekül entlang der z -Achse orientiert.

wir für VF_2 15 angeregte Zustände in einem Bereich von 2,0 eV über dem Grundzustand gefunden. Um die Übergangsmetallverbindungen quantitativ korrekt zu behandeln, ist daher die Verwendung großer Referenzräume angeraten, wodurch wiederum die Behandlung großer Hilberträume notwendig wird.

Wir haben mit der Berechnung der elektronischen Eigenschaften von VF_2 und VCl_2 eine systematische Untersuchung der Übergangsmetallhalogenide begonnen (MX_2 , $\text{M} = \text{Sc}, \dots, \text{Zn}$ und $\text{X} = \text{F}, \text{Cl}, \text{Br}$). Aufgrund ihrer interessanten Eigenschaften sind diese Verbindungen häufig experimentell untersucht worden [45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55]. Trotz ebenfalls sehr reger theoretischer Forschungen [56, 57, 58, 59, 60, 61, 62, 63, 64], sind bis heute viele Fragen bezüglich der theoretischen Beschreibung der Übergangsmetallhalogenide offen geblieben. Eine Reihe von theoretischen Studien, die zumindest teilweise elektronische Korrelationseffekte berücksichtigen [57, 62], hat Zweifel an der Zuordnung der Spektren, die mit Hilfe der Ligandenfeldtheorie (LFT) vorgenommen wurde, aufgeworfen. Eine aktuelle theoretische Untersuchung auf der Basis von Dichtefunktionaltheorie [64] hat einige der spektralen Neuzuordnungen bestätigt. Diese Untersuchung wies jedoch auch einige signifikante qualitative und quantitative Abweichung mit dem Experiment und mit früheren theoretischen Arbeiten bei der Berechnung von Bindungslängen, Dissoziationsenergien und Anregungsenergien auf.

Wir haben daher die vertikalen und adiabatischen Anregungsenergien von VF_2 und VCl_2 mit der MRD-CI Methode berechnet. Beide Moleküle sind linear, der experimentelle V–Cl Bindungsabstand beträgt 2,172 Å [65], für VF_2 ist uns keine experimentelle Messung der Bindungslänge bekannt. Aufgrund ihrer Linearität haben die Moleküle die Symmetrie $D_{\infty h}$. Da jedoch diese Symmetriegruppe von unserem Programm nicht unterstützt wird,

	A_g	B_{1u}	B_{2u}	B_{3g}	B_{3u}	B_{2g}	B_{1g}	A_u
	A	E	C	G	B	F	D	H
core(VF ₂)	4	2	2	0	3	0	0	0
core(VCl ₂)	6	3	3	0	5	1	1	0
inactive	2	1	1	0	2	1	1	0
active1	6	1	1	2	3	2	2	0
active2	4	1	1	2	1	2	2	0

Tabelle 5.7: Aktiver Raum bei der Berechnung der vertikalen und adiabatischen Anregungsenergien von VF₂ und VCl₂. Der mit *active2* bezeichnete Raum wurde benutzt, um einen Vergleich mit einer MOLPRO-Rechnung durchzuführen.

Sym($D_{\infty h}$)	Sym(D_{2h})	E_{ad} [eV]	E_v [eV]	E_{ad} (DFT) [eV]
$^4\Sigma_g^-$	$^4B_{1g}$	0	0	0
$^4\Pi_g$	$^4B_{3g}$	0.36	0.42	1.12
$^4\Phi_g$	$^4B_{3g}$	0.96	0.99	—
$^4\Delta_g$	4A_g ($^4B_{1g}$)	1.09 (1.26)	1.33 (1.51)	—
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.55 (1.63)	1.56 (1.63)	2.20
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.61 (1.65)	1.61 (1.65)	—
$^2\Pi_g$	$^2B_{3g}$	1.80	1.83	4.17

Tabelle 5.8: Adiabatische und vertikale Anregungsenergien von VF₂. Die Energien sind in eV relativ zum Grundzustand (-1142,526720 H) angegeben. Die Ergebnisse wurden mit der MRD-CI Methode in einer aug-DZP Basis von Roos *et al.* [66, 67] berechnet.

wurden die Rechnungen in D_{2h} -Symmetrie durchgeführt. In Tabelle 5.6 ist aufgezeigt, wie sich die ersten Symmetrieklassen der $D_{\infty h}$ -Gruppe auf die der D_{2h} -Gruppe verteilen. Zusätzlich ist dort die Bezeichnung der Symmetrien innerhalb unseres Programms angegeben, die sich ergeben, wenn die Moleküle entlang der z -Achse orientiert werden.

VF₂ besitzt 41, VCl₂ 49 Elektronen. Die Rechnungen wurden unter Verwendung von augmentierten *double zeta* und *triple zeta* Basissätzen von Roos *et al.* durchgeführt. Die SCF-Rechnung wurde in der Besetzung $7a_g^2 3b_{3u}^2 3b_{2u}^2 5b_{1u}^2 1b_{2g}^2 1b_{3g}^2 1a_g^1$ durchgeführt. Bei VCl₂ waren zusätzlich $2a_g^2 2b_{1u}^2 1b_{2u}^2 1b_{3g}^2 1b_{3u}^2 1b_{2g}^2$ besetzt, was der geschlossenen $2s2p$ -Schale der Chloratome entspricht.

Für die Berechnung der Potentialenergieoberflächen von VF₂ und VCl₂ haben wir einen aktiven Raum benutzt, wie er in Tabelle 5.7 beschrieben ist. Die Besetzung der *core*-Orbitale entspricht bei VF₂ den $1s$ -Orbitalen der Fluoratome sowie den $1s, 2s, 2p, 3s$ und $3p$ Orbitalen des Vanadiumatoms. Bei VCl₂ sind zusätzlich die $2s$ und $2p$ Orbitale der Chloratome eingefroren. Die Besetzung der inaktiven Orbitale entspricht der Tatsache, dass die Valenzschalen von Fluor bzw. Chlor voll besetzt sind. Die Wahl des aktiven Raums muss einen Kompromiss zwischen einer maximalen Größe des Referenzraums und

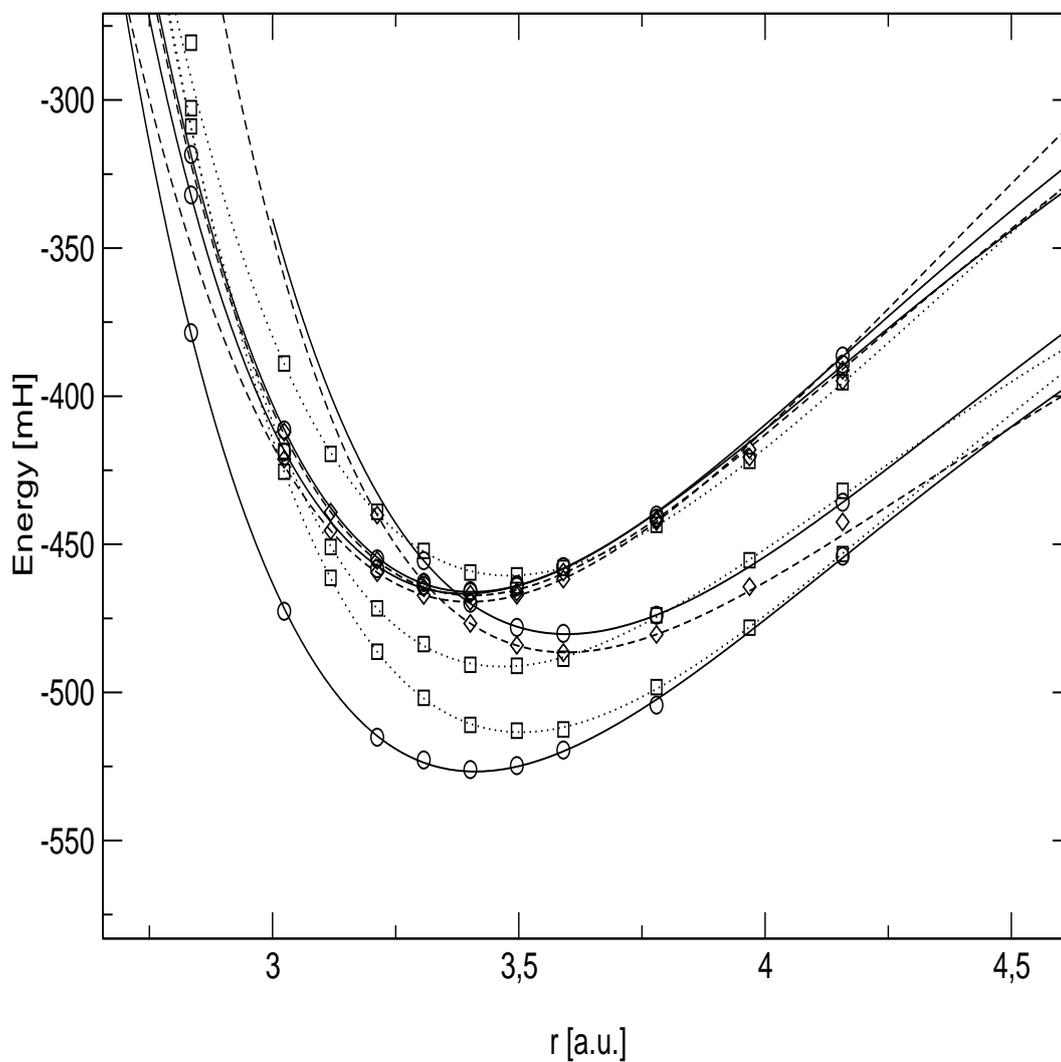


Abbildung 5.7: Potentialenergieoberfläche des Grundzustands und der ersten angeregten Zustände von VF₂. Auf der vertikalen Achse ist die Energie in mH relativ zu -1142,0 H aufgetragen, die horizontale Achse gibt den V-F-Bindungsabstand in a.u. an. Die Linientypen und Symbole geben die Symmetrie des Zustands an (durchgezogen, Kreise: B_{1g}; gepunktet, Quadrate: B_{3g}; gestrichelt, Rauten: A_g).

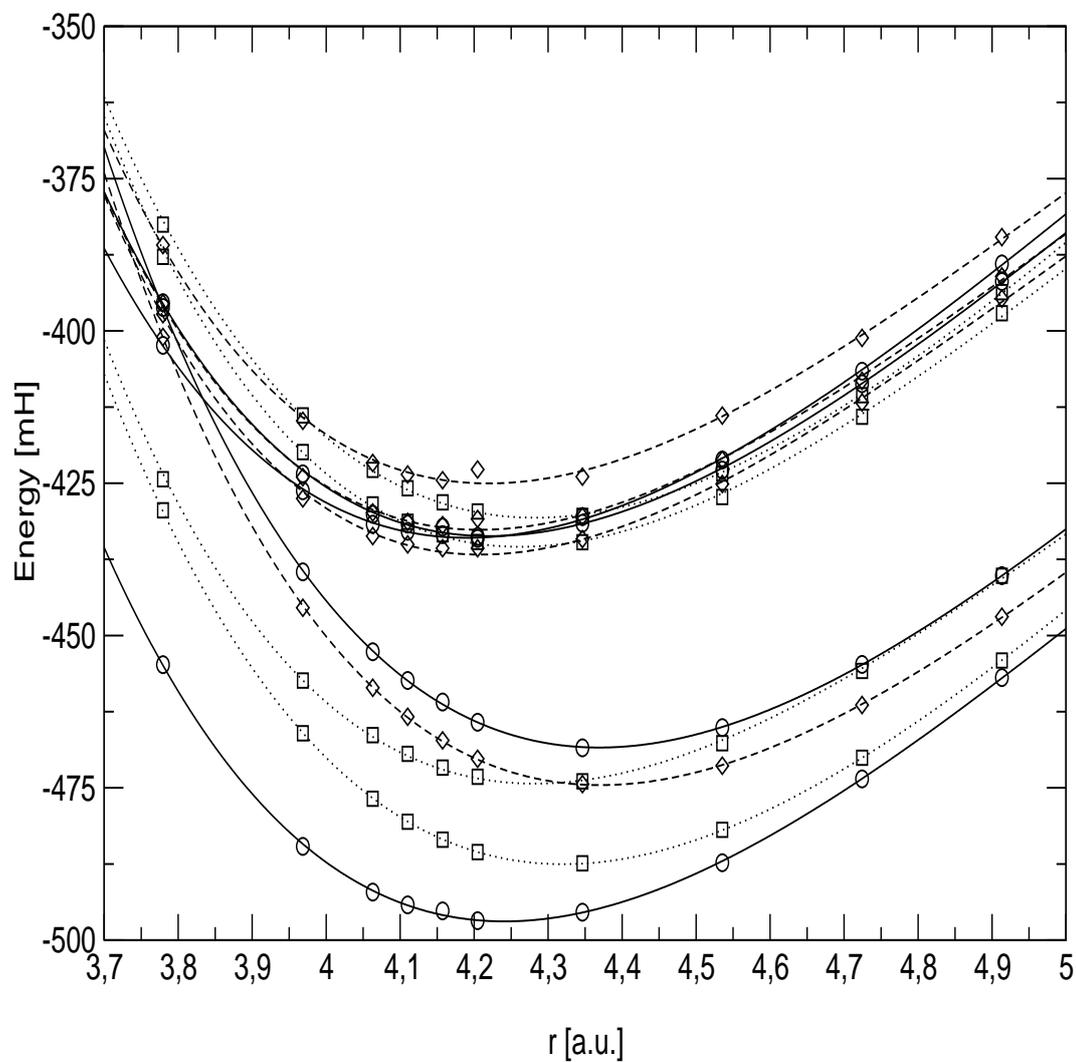


Abbildung 5.8: Potentialenergieoberfläche des Grundzustands und der ersten angeregten Zustände von VCl_2 . Auf der vertikalen Achse ist die Energie in mH relativ zu $-1862,0$ H aufgetragen, die horizontale Achse gibt den V-F-Bindungsabstand in a.u. an. Die Linientypen und Symbole geben die Symmetrie des Zustands an (durchgezogen, Kreise: B_{1g} ; gepunktet, Quadrate: B_{3g} ; gestrichelt, Rauten: A_g).

Sym($D_{\infty h}$)	Sym(D_{2h})	E_{ad} [eV]	E_{v} [eV]	$E_{\text{ad}}(\text{DFT})$
$^4\Sigma_g^-$	$^4B_{1g}$	0	0	0
$^4\Pi_g$	$^4B_{3g}$	0.26	0.28	0.81
$^4\Delta_g$	4A_g ($^4B_{1g}$)	0.60 (0.77)	0.67 (0.84)	—
$^4\Pi_g$	$^4B_{3g}$	0.61	0.62	—
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.63 (1.71)	1.64 (1.72)	1.82
$^4\Pi_g$	$^4B_{3g}$	1.67	1.67	—
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.74 (1.72)	1.75 (1.72)	—
$^2\Pi_g$	$^2B_{3g}$	1.80	1.80	3.35

Tabelle 5.9: Adiabatische und vertikale Anregungsenergien von VCl_2 . Die Energien sind in eV relativ zum Grundzustand (-1862,496919 H) angegeben. Die Ergebnisse wurden mit der MRD-CI Methode in einer aug-DZP Basis von Roos *et al.* [66, 67] berechnet.

einer vertretbaren Rechenzeit darstellen. Die in Tabelle 5.7 gezeigte Konfiguration von aktiven Orbitalen ist deutlich größer als der Valenzraum von Vanadium und umfasst die wichtigsten niedrigliegenden Orbitale.

Für jede Symmetrie wurden die Grundzustandsenergie und die Energien der ersten drei angeregten Zustände für 8–10 Bindungslängen in der Nähe des Minimums berechnet. Die beschriebene Wahl des aktiven Raums führte je nach Symmetrie auf 280–330 Referenzen und Hilberträume einer Dimension bis 30×10^6 Konfigurationen, von denen bei einem Selektionsschwellwert von 3×10^{-5} ca. 10^6 Konfigurationen für die Variationsrechnung selektiert wurden. Die perturbative Energiekorrektur betrug Bruchteile eines milli-Hartree. Die Rechnungen wurden auf 64 Knoten der IBM-SP/2 des SCC Karlsruhe durchgeführt; die Laufzeit einer Rechnung bis zur Konvergenz von vier Zuständen betrug knapp 2 Stunden.

Die berechneten Potentialenergieoberflächen sind in den Abbildungen 5.7 und 5.8 zu sehen. Die MRD-CI Energien wurden in der Nähe des Minimums durch ein Morse-Potential interpoliert; die Abweichung der interpolierten Energien von den berechneten Energien betrug in allen Fällen weniger als ein milli-Hartree. Für den V–F Abstand im Grundzustand haben wir eine Länge von 1,8051 Å erhalten. Dieser Wert ist etwa 0,04 Å größer als der in der Dichtefunktionaltheorie berechnete Wert [64]. Experimentelle Daten für die Bindungslänge von VF_2 sind unseres Wissens nicht bekannt. Der V–Cl Abstand ergab sich zu 2,244 Å und ist damit 0,06 Å größer als das entsprechende DFT-Resultat. Der experimentelle Wert für die Bindungslänge liegt bei 2,172 Å [65], also näher am DFT-Resultat als am MRD-CI Ergebnis.

Die berechneten adiabatischen und vertikalen Anregungsenergien finden sich in den Tabellen 5.8 und 5.9. Wie in der DFT-Rechnung finden wir in beiden Fällen einen $^4\Sigma_g^-$ -Zustand als Grundzustand und einen $^4\Pi_g$ -Zustand als ersten angeregten Zustand. Die berechneten Energien weichen jedoch signifikant von den DFT-Ergebnissen ab. Experimentelle Ergebnisse für die Anregungsenergien von VF_2 und VCl_2 gibt es unseres Wissens nicht.

Sym($D_{\infty h}$)	Sym(D_{2h})	E_v (QCI)[eV]	E_v (MOLPRO) [eV]	E_v (active1)
$^4\Sigma_g^-$	$^4B_{1g}$	0	0	0
$^4\Pi_g$	$^4B_{3g}$	0.61	0.58	0.42
$^4\Pi_g$	$^4B_{3g}$	1.00	1.02	0.99
$^4\Delta_g$	4A_g ($^4B_{1g}$)	1.43 (1.76)	1.57 (1.70)	1.33 (1.65)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.58 (1.61)	1.64 (1.57)	1.56 (1.51)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.60 (1.69)	1.70 (1.64)	1.61 (1.64)
$^2\Pi_g$	$^2B_{3g}$	1.92	1.88	1.83

Tabelle 5.10: Vertikale Anregungsenergien bei $r_{VF} = 1.75 \text{ \AA}$, die mit unserem Programm und MOLPRO berechnet wurden. In beiden Rechnungen wurde eine aug-DZP Basis von Roos *et al.* [66, 67] und ein aktiver Raum der Konfiguration *active2* aus Tab. 5.7 verwendet.

Sym($D_{\infty h}$)	Sym(D_{2h})	E_v (aug-DZP)[eV]	E_v (aug-TZP)[eV]
$^4\Sigma_g^-$	$^4B_{1g}$	0	0
$^4\Pi_g$	$^4B_{3g}$	0.43	0.47
$^4\Pi_g$	$^4B_{3g}$	1.09	1.06
$^4\Delta_g$	4A_g ($^4B_{1g}$)	1.36 (1.45)	1.31 (1.38)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.49 (1.53)	1.55 (1.62)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.55 (1.60)	1.57 (1.65)
$^4\Pi_g$	$^2B_{3g}$	1.80	1.81

Tabelle 5.11: Vertikale Anregungsenergien bei $r_{VF} = 1.8051976 \text{ \AA}$, berechnet in aug-DZP und aug-TZP Basissätzen von Roos *et al.* [66, 67]. Für die Rechnungen wurde der kleinere aktive Raum (*active 2*) verwendet.

Um unsere Ergebnisse zu überprüfen, haben wir die Anregungsenergien von VF_2 mit der intern-kontrahierten MRCI-Methode, die im kommerziellen Programmpaket MOLPRO implementiert ist, berechnet. Da jedoch der in Tabelle 5.7 beschriebene aktive Raum aufgrund seiner Größe nicht mehr von MOLPRO behandelt werden kann, musste als Kompromiss der in derselben Tabelle als *active2* beschriebene aktive Raum verwendet werden. Die MOLPRO-Rechnungen wurden von M. Vogel bei einem V-F Abstand von $1,75 \text{ \AA}$ in einer aug-DZP Basis von Roos *et al.* [66, 67] durchgeführt. Tabelle 5.10 zeigt einen Vergleich der MOLPRO-Ergebnisse mit Ergebnissen, die wir mit unserem Programm bei diesen Parametern erhalten haben. Die berechneten Energien zeigen eine exzellente Übereinstimmung; die maximale Abweichung beträgt $0,1 \text{ eV}$. In Tabelle 5.10 werden weiterhin Ergebnisse gezeigt, die bei $r_{VF} = 1.75 \text{ \AA}$ mit unserem Programm und dem größeren aktiven Raum (*active1*) erhalten wurden. Bis auf die Energie des ersten angeregten Zustands, die um fast $0,2 \text{ eV}$ abweicht, stimmen die Ergebnisse innerhalb von $0,1 \text{ eV}$ mit den im kleineren aktiven Raum berechneten Ergebnissen überein. Das zeigt, dass der kleinere aktive Raum (*active2*) bereits völlig ausreichend für die Behandlung von VF_2 ist.

Sym($D_{\infty h}$)	Sym(D_{2h})	$E_v(\text{VC})[\text{eV}]$	$E_v[\text{eV}]$
$^4\Sigma_g^-$	$^4B_{1g}$	0	0
$^4\Pi_g$	$^4B_{3g}$	0.33	0.43
$^4\Pi_g$	$^4B_{3g}$	0.88	1.09
$^4\Delta_g$	4A_g ($^4B_{1g}$)	1.19 (1.21)	1.36 (1.45)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.55 (1.62)	1.49 (1.53)
$^2\Delta_g$	2A_g ($^2B_{1g}$)	1.57 (1.63)	1.55 (1.60)
$^2\Pi_g$	$^2B_{3g}$	1.66	1.80

Tabelle 5.12: Vertikale Anregungsenergien von VF_2 bei $r_{\text{VF}} = 1.8051976 \text{ \AA}$ und in einer aug-DZP Basis, mit und ohne Valence-Core Korrekturen(VC). Für die Rechnungen wurde der kleiner aktive Raum (*active 2*) benutzt.

	$\omega_{\text{sym}} [\text{cm}^{-1}]$	$\omega_{\text{asym}} [\text{cm}^{-1}]$	$\omega_{\text{bend}} [\text{cm}^{-1}]$	$D_e [\text{eV}]$
VF_2 (MRD-CI)	582,5	769,7	189,5	10.7
VF_2 (DFT)	630	773	—	13.35
VCl_2 (MRD-CI)	322,8	499,3	90,9	9.1
VCl_2 (DFT)	345	492	48	9.53

Tabelle 5.13: Harmonische Frequenzen und Dissoziationsenergien für VF_2 und VCl_2 . Diese Frequenzen wurden aus den in den Abb. 5.7 und 5.8 gezeigten Potentialenergieoberflächen berechnet.

Um den Einfluss der verwendeten Basis zu untersuchen, haben wir bei $r_{\text{VF}} = 1.8051976$ die vertikalen Anregungsenergien von VF_2 unter Verwendung von aug-DZP und aug-TZP Basissätzen von Roos *et al.* [66, 67] berechnet. Für diese Rechnung haben wir den kleineren aktiven Raum vom Typ *active 2* verwendet. Die Ergebnisse sind in Tabelle 5.11 protokolliert. Es zeigt sich eine exzellente Übereinstimmung beider Rechnungen, so dass die Benutzung einer aug-DZP Basis für die Korrelationsrechnung ausreicht.

Schließlich haben wir noch Valenz-Core-Korrekturen (VC) untersucht, indem wir die 3s3p-Schale des Vanadiumatoms nicht mehr eingefroren haben. Stattdessen haben wir diese Orbitale als inaktive Orbitale behandelt. Auch für diese Rechnung haben wir, um Rechenzeit zu sparen, den kleineren aktiven Raum (*active 2*) benutzt. Die Ergebnisse dieser Rechnung sind in Tab. 5.12 dargestellt. Wiederum sind nur sehr geringe Abweichung zwischen den Rechnungen aufgetreten: Die größte Abweichung findet man für den letzten $^2B_{3g}$ -Zustand, dessen Anregungsenergie mit VC-Korrektur von 1,87 eV auf 1,66 eV sinkt. Alle anderen Energien stimmen bis auf 0,1 eV mit den vorher berechneten Energien überein.

Wir haben die berechneten Potentialenergieoberflächen dazu benutzt, um einige spektroskopische Größen von VF_2 und VCl_2 zu berechnen. Die Eigenschwingungen linearer dreiatomiger Moleküle sind in Abb. 5.9 dargestellt: In der symmetrischen Valenzschwingung schwingen die Halogenatome (X) in entgegengesetzte Richtungen, das Übergangsmetallatom (M) ruht. In der asymmetrischen Valenzschwingung schwingen die Haloge-

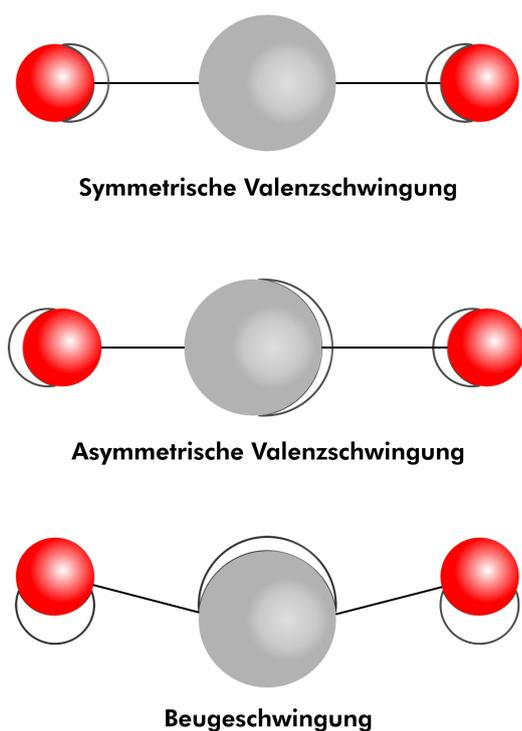


Abbildung 5.9: Eigenschwingungen eines linearen dreiatomigen Moleküls. In der symmetrischen Valenzschwingung schwingen die Halogenatome in entgegengesetzte Richtung, das Übergangsmetallatom ruht. In der asymmetrischen Valenzschwingung schwingen die Halogenatome entgegengesetzt zum Übergangsmetallatom. In der Beugeschwingung bleibt die Bindungslänge konstant und der Bindungswinkel verändert sich.

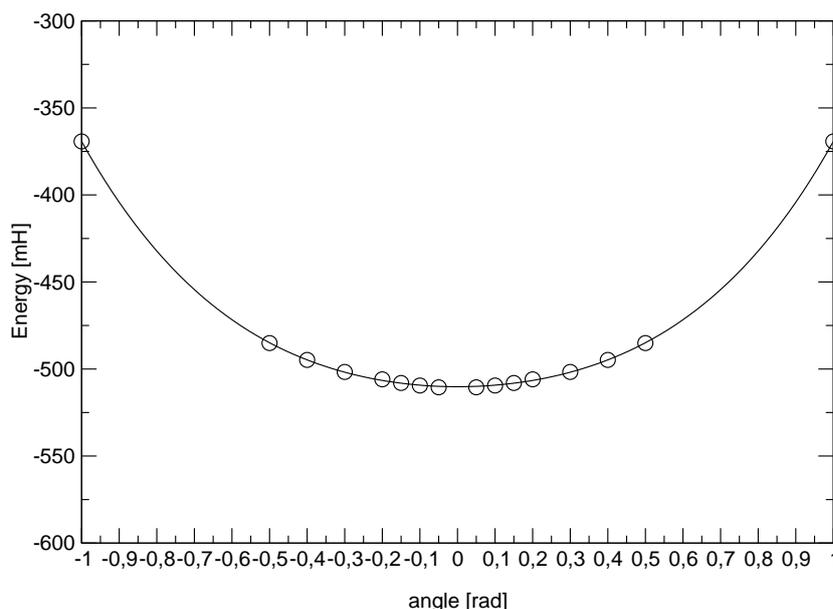


Abbildung 5.10: Potentialenergieoberfläche von VF_2 bei konstanter Bindungslänge $r = 1,8051976 \text{ \AA}$ und verschiedenen Bindungswinkeln. Die durchgezogene Linie ist ein polynomialer Fit der berechneten MRD-CI Energien.

natome entgegengesetzt zum Übergangsmetallatom. In der Beugeschwingung bleibt die Bindungslänge konstant und der Bindungswinkel verändert sich. Modelliert man das lineare dreiatomige Molekül durch einen gekoppelten harmonischen Oszillator, in dem die Halogenatome mit dem Übergangsatome jeweils durch Federn mit der Federkonstanten k verbunden sind, so erhält man die Frequenzen für die symmetrische und asymmetrische Valenzschwingung durch

$$\omega_{\text{sym}} = \frac{1}{2\pi c} \sqrt{\frac{k}{m_{\text{X}}}} \quad (5.6)$$

und

$$\omega_{\text{asym}} = \frac{1}{2\pi c} \sqrt{\frac{k(2m_{\text{X}} + m_{\text{M}})}{m_{\text{X}}m_{\text{M}}}}. \quad (5.7)$$

Die Kraftkonstante k lässt sich aus den berechneten Potentialenergieoberflächen (Abb. 5.7 und 5.8) berechnen durch

$$k = \frac{1}{2} \left. \frac{d^2\phi}{dr_{\text{MX}}^2} \right|_{r_0} \quad (5.8)$$

Der Vorfaktor $1/2$ resultiert daraus, dass das Potential ϕ für die symmetrische Dehnung *beider* M–X Bindungen berechnet wurde.

In der Beugeschwingung ist (in harmonischer Näherung) die rücktreibende Kraft proportional zur Änderung des Winkels δ :

$$F = -\kappa r_0 \delta. \quad (5.9)$$

Die Frequenz der Beugeschwingung erhält man aus der Kraftkonstanten κ durch

$$\omega_{\text{bend}} = \frac{1}{2\pi c} \sqrt{\frac{\kappa 2(2m_X + m_M)}{m_X m_M}}. \quad (5.10)$$

Um ω_{bend} zu berechnen, haben wir die Grundzustandsenergie von VF_2 und VCl_2 bei $r_{\text{VF}} = 1.8051976 \text{ \AA}$ und $r_{\text{VCl}} = 2.2441077 \text{ \AA}$ für Winkel $\delta = 0.01, 0.02, 0.05, 0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 1.0$ berechnet. Die berechnete Potentialenergieoberfläche wurde dann durch ein symmetrisches Polynom vierten Grades angefitet (Abb. 5.10).

Die berechneten Frequenzen sind zusammen mit den Dissoziationsenergien von VF_2 und VCl_2 in Tabelle 5.13 aufgelistet. Für die Frequenzen sind keine experimentellen Werte bekannt. Die von Wang und Schwarz mit DFT berechneten Frequenzen sind für die symmetrische Valenzschwingungen etwas größer als die von uns berechneten Werte. Sowohl bei VF_2 als auch bei VCl_2 stimmen die Werte für die asymmetrische Valenzschwingung fast überein. Um mehr als hundert Wellenzahlen unterscheiden sich die (insgesamt sehr niedrigen) Frequenzen der Beugeschwingung von den DFT-Resultaten. Für die Dissoziationsenergien haben wir $D_e(\text{VF}_2) = 10.7 \text{ eV}$ und $D_e(\text{VCl}_2) = 9.1 \text{ eV}$ gefunden. Diese Werte sind 1,3 eV bzw. 0,9 eV niedriger als der *geschätzte* experimentelle Wert $D_e(\text{VF}_2) = 12 \text{ eV}$ [64] und der gemessene Wert $D_e(\text{VCl}_2) = 10 \text{ eV}$ [65].

5.4 Konische Überschneidung von NO_2

Vibronische Wechselwirkungen zwischen verschiedenen Potentialenergieoberflächen sind eine generelle Eigenschaft mehratomiger Moleküle. Derartige Wechselwirkungen sind stark für entartete oder fast entartete elektronische Zustände. Die spektroskopischen und dynamischen Eigenschaften des Moleküls werden stark beeinflusst von den zu diesen Zuständen gehörenden nichtadiabatischen Kopplungen [68, 69, 70, 71]. Ein typisches Szenario in mehratomigen Molekülen ist das Entstehen von sog. *konischen Überschneidungen* (Abb. 5.11) von Potentialenergieoberflächen. Der dominierende Kopplungsterm ist hier linear in den Auslenkungen der Kerne. Bei der Behandlung eines solchen Systems müsste man theoretisch die Born-Oppenheimer Näherung hinter sich lassen und komplizierte gekoppelte Differentialgleichungen lösen, um gleichzeitig die Bewegung der Kerne auf mehreren elektronischen Zuständen zu betrachten. Auch wenn dieses Konzept in der Literatur schon lange bekannt ist, ist es bis heute äußerst schwierig, nichtadiabatische elektronische Zustände für mehratomige Moleküle zu berechnen [72].

NO_2 ist ein wohlbekanntes dreiatomiges Molekül mit außerordentlich komplexen spektroskopischen Eigenschaften. Die meisten theoretischen Untersuchungen dieses Systems beschäftigten sich mit dem Verständnis der hohen Dichte der Spektrallinien des optischen Spektrums. Bereits Mitte 1970 zeigten die theoretischen Arbeiten von Gillispie *et al.* [73] und Jackels und Davidson [74], dass sich die Potentialenergieoberfläche des elektronischen Grundzustand von NO_2 (\tilde{X}^2A_1) mit der mehrerer angeregter Dublettzustände (\tilde{A}^2B_2 ,

\tilde{B}^2B_1 und \tilde{C}^2A_2) kreuzt. Der \tilde{A}^2B_2 -Zustand ist bei relativ niedrigen Energien über die asymmetrische Valenzschwingung mit dem Grundzustand gekoppelt. Ihre Potentialenergieoberflächen bilden hier eine *konische Überschneidung*. Der elektronische Grundzustand überschneidet sich auch mit dem \tilde{B}^2B_1 -Zustand bei linearer Geometrie und bildet hier ein Renner-Teller Paar [75]. Der \tilde{A}^2B_2 und \tilde{B}^2B_1 -Zustand sind optisch aktiv, während ein Übergang zwischen dem \tilde{C}^2A_2 -Zustand und dem \tilde{X}^2A_1 -Grundzustand optisch nicht sichtbar ist. Der \tilde{C}^2A_2 -Zustand ist an den \tilde{B}^2B_1 -Zustand ebenfalls über die asymmetrische Valenzschwingung gekoppelt (jedoch bei viel höheren Energien); ihre Potentialenergieoberflächen bilden dort eine weitere konische Überschneidung.

Die Komplexität des optischen Absorptionsspektrums (mit steigender Energie) resultiert hauptsächlich aus den nichtadiabatischen Kopplungen, die durch die konische Überschneidung des \tilde{X}^2A_1 - und des \tilde{A}^2B_2 -Zustands verursacht werden. Aus diesem Grund konzentrieren sich fast alle theoretischen Studien des Absorptionsspektrums von NO_2 auf die gekoppelten Zustände \tilde{X}^2A_1 und \tilde{A}^2B_2 . NO_2 ist ein stark gebundenes Molekül, das C_{2v} Minima sowohl im \tilde{X}^2A_1 - als auch im \tilde{A}^2B_2 -Zustand aufweist. Um NO_2 in eine lineare Konfiguration zu bringen, ist eine Energiebarriere von ~ 1.83 eV [76] zu überwinden. Die Entartung zwischen \tilde{X}^2A_1 - und \tilde{B}^2B_1 -Zustand liegt damit energetisch deutlich über der minimalen Energie des \tilde{A}^2B_2 -Zustands. Es ist daher nicht zu erwarten, dass diese Kopplung starken Einfluss auf die niedrigenergetischen Schwingungen im Absorptionsspektrum hat.

Der Gleichgewichtszustand von NO_2 im \tilde{X}^2A_1 -Grundzustand stellt sich bei einer Bindungslänge von $r \approx 1.20$ Å und einem Bindungswinkel von $\phi \approx 134^\circ$ ein. Den Gleichgewichtszustand von \tilde{A}^2B_2 findet man bei $r \approx 1.26$ Å und $\phi \approx 102^\circ$. Aufgrund des relativ großen Unterschieds in den Bindungswinkeln, ist es schwierig, die niedrigliegenden vibronischen Anregungen des \tilde{A}^2B_2 -Zustands mit den Mitteln optischer Spektroskopie zu messen. Die Gleichgewichtsgeometrie des NO_2^- Anions liegt mit $r = 1.25 \pm 0.02$ Å und $\phi = 117.5 \pm 2^\circ$ wesentlich näher an der des \tilde{A}^2B_2 -Zustands. Aus diesem Grund konnte von Weaver *et al.* [77] die Photoelektronenspektroskopie an NO_2^- erfolgreich eingesetzt werden, um die niedrigliegenden Schwingungsübergänge des \tilde{A}^2B_2 -Zustands zu beobachten. Das Photoelektronenspektrum von NO_2^- für einen Übergang zum \tilde{X}^2A_1 -Zustand von NO_2 wurde auch von Ervin *et al.* [78] aufgenommen.

Wir haben adiabatische, dreidimensionale Potentialenergieoberflächen mit der MRD-CI Methode berechnet. Die Rechnungen wurden unter Verwendung einer cc-pVTZ Basis von Dunning *et al.* [17] durchgeführt. Die Wahl dieser Basis führte auf insgesamt 90 atomare Orbitale, aus denen mit der BSR-Methode approximative natürliche Orbitale generiert wurden. Die Rechnungen wurden auch dann in C_s -Symmetrie durchgeführt, wenn eigentlich eine C_{2v} -Symmetrie vorlag. Es ist bekannt, dass ein Wechsel der Symmetrie bei der Berechnung der Potentialenergieoberfläche zu Sprüngen führen kann. Um eine glatte Potentialenergieoberfläche zu erhalten, wurde daher ausschließlich in C_s -Symmetrie gerechnet.

Die SCF-Rechnungen für diese Studie wurden im niedrigsten ${}^2A'$ -Zustand durchgeführt.

Die zugehörige Konfiguration der Orbitale war $9a'^2 2a''^2 1a'^1$. Diese Konfiguration entspricht in C_{2v} -Symmetrie einem 2A_1 -Zustand für größere Bindungswinkel, jedoch einem 2B_2 -Zustand für kleinere Bindungswinkel. In der MRD-CI wurde ein *Core* von sechs Orbitalen, entsprechend den 1s-Orbitalen von Sauerstoff und Stickstoff, doppelt besetzt gehalten. Der Referenzraum bestand aus 5 inaktiven Orbitalen ($5a'$) und 5 aktiven Orbitalen ($2a' 3a''$). Der Hilbertraum enthielt 6,3 Mio Konfigurationen, von denen in den Rechnungen bei einer Schwelle von $\lambda = 10^{-4}$ jeweils ca. 1 % selektiert wurde. Die perturbative Energiekorrektur betrug jeweils ca. 10 mH.

Mit dem Davidson-Verfahren (s. 2.2) wurden die beiden niedrigsten Eigenwerte bis zu einer Genauigkeit von $1 \mu H$ bestimmt. Um numerische Schwierigkeiten in der Nähe der konischen Überschneidung zu vermeiden, haben wir eine spezielle Version des Davidson-Iterators entwickelt, in der die Wellenfunktionen Ψ_1 und Ψ_2 der beiden Zustände abwechselnd iteriert wurden, wobei in jeder Iteration die Orthogonalität von Ψ_1 und Ψ_2 verlangt wurde.

Wir haben die Energien auf einem Gitter in

$$\begin{aligned} r &= \frac{r_1 + r_2}{2}, \\ \delta r &= \frac{r_1 - r_2}{2} \quad \text{und} \\ \phi & \end{aligned} \tag{5.11}$$

berechnet, wobei r_1 und r_2 die Abstände zwischen dem Stickstoffatom und den beiden Sauerstoffatomen sind und ϕ der Bindungswinkel. Der symmetrische Bindungsabstand r lief von 1.1 bis 1.5 Å in Abständen von 0.05 Å, ϕ lief von 60° bis 180° in Abständen von 20°. Für δr haben wir Abstände von $\delta r = 0.0, 0.05, 0.1, 0.2, 0.4$ Å verwendet. Insgesamt führte das zu einem Gitter von 315 Datenpunkten.

Ausgehend von diesen Datenpunkten haben wir zunächst den Bindungswinkel $\phi_c(r)$, bei dem die konische Überschneidung auftrat, als Funktion von r gefittet. Ausgehend vom Ort der konischen Überschneidung $(r, \phi_c(r))$ haben wir dann die eindimensionalen Potentialenergieoberflächen $E_{1,2}(r, \phi_c(r), \delta r)$ als Funktion von $\delta r = 0.00, 0.02, 0.04, 0.08, 0.10, 0.20, 0.40$ berechnet.

Um eine glatte, analytische Darstellung der gesamten Potentialenergieoberfläche zu erhalten, haben wir als nächstes einen zweidimensionalen polynomialen Fit des Gaps

$$\Delta(r, \delta r) = E_1(r, \phi_c(r), \delta r) - E_2(r, \phi_c(r), \delta r) \tag{5.12}$$

in δr berechnet. Ausgehend von dieser analytischen Form von Δ haben wir die Größen $A(r, \phi, \delta r)$ und $B(r, \phi, \delta r)$ derart bestimmt, dass die Eigenwerte der Matrix

$$\begin{pmatrix} A(r, \phi, \delta r) & \Delta(r, \delta r) \\ \Delta(r, \delta r) & B(r, \phi, \delta r) \end{pmatrix} \tag{5.13}$$

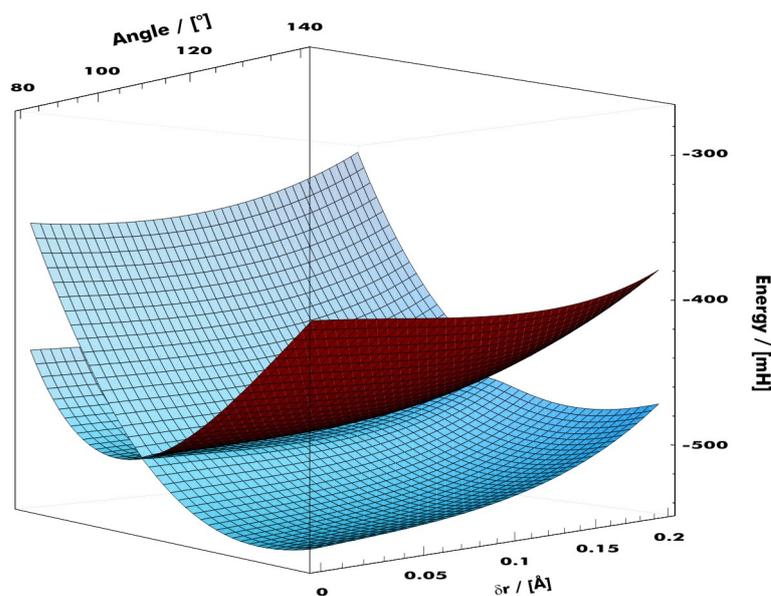


Abbildung 5.11: Zweidimensionaler Schnitt durch die dreidimensionale adiabatische Potentialenergieoberfläche des \tilde{X}^2A_1 und des \tilde{B}^2B_1 -Zustands von NO_2 in der Umgebung der konischen Überschneidung unter Benutzung der im Text beschriebenen Darstellung. Die mittlere Bindungslänge war $r = 1,25 \text{ \AA}$. Auf den beiden horizontalen Achsen sind der Bindungswinkel ϕ und die asymmetrische Dehnung δr angegeben. Auf der vertikalen Achse ist die Energie in mH relativ zu $E_0 = -204.0 \text{ H}$ aufgetragen.

gleich den berechneten Energien an allen Punkten des Gitters sind. Das nichtanalytische Verhalten der Potentialenergieoberfläche nahe der konischen Überschneidung wird durch diesen Fit gut reproduziert. Für C_{2v} -Geometrien, also $\delta r = 0$, sind A und B die Energien der beiden Zustände. Senkrecht zur C_{2v} -Oberfläche ist das Verhalten nahe der konischen Überschneidung gegeben durch $E(r, \phi_c(r), 0) + \Delta(r, \phi_c(r), \delta r)$. Insgesamt erhalten wir A , B und Δ als glatte und analytische Funktionen aller Variablen. Im Gegensatz dazu enthalten die ursprünglichen Energien E_1 starke nichtanalytische Komponenten.

Schließlich wurden A , B und Δ durch einen dreidimensionalen Spline-Fit interpoliert, so dass die Energien E_1 und E_2 für jeden Wert von r , ϕ und δr durch das Bestimmen der Eigenwerte der in Gleichung (5.13) gegebenen Matrix berechnet werden können. Ein zweidimensionaler Schnitt durch die in dieser Darstellung berechnete adiabatische Potentialenergieoberfläche ist in Abb. 5.11 dargestellt.

Die Energien und Geometrien verschiedener Punkte sind in Tabelle 5.14 aufgelistet. Der \tilde{A}^2B_2 -Zustand nimmt sein Minimum bei 1,207 eV relativ zum Minimum des \tilde{X}^2A_1 -Zustands an. Das Minimum der eindimensionalen Teilmenge der Potentialenergieoberfläche (im folgenden *Cusp* genannt), die durch die konische Überschneidung gegeben ist, tritt bei 1,287 eV relativ zum \tilde{X}^2A_1 -Zustand auf. Der Cusp liegt damit nur 0,08 eV über dem Minimum des \tilde{A}^2B_2 -Zustands. Das ist weniger als ein Quant der Beugeschwingung im

	\tilde{X}^2A_1		\tilde{A}^2B_2	
	MRD-CI	Exp ^a	MRD-CI	Exp ^b
$r[\text{Å}]$	1.186	1.194	1.247	–
$\phi[^\circ]$	133.6	133.9	102.6	—
$E[\text{eV}]$	0.0	0.0	1.207	1.210±0.012
	Cusp		NO ₂ ⁻	
	MRD-CI	Exp ^c	MRD-CI	Exp ^d
$r[\text{Å}]$	1.242	1.246	1.257	1.25±0.02
$\phi[^\circ]$	108.04	103.1	117.2	117.5±2
$E[\text{eV}]$	1.287	1.21± 0.09	–	–

Tabelle 5.14: Geometrien und Energien für verschiedene Punkte des \tilde{X}^2A_1 und des \tilde{A}^2B_2 -Zustands und für den $\tilde{X}^2A_1/\tilde{A}^2B_2$ Cusp von NO₂, die aus dem dreidimensionalen Fit der berechneten Potentialenergieoberfläche gewonnen wurden, zusammen mit den experimentellen Werten.

^a Werte aus Ref. [79, 76]

^b Werte aus Ref. [80]

^c Werte aus Ref. [81]

^d Werte aus Ref. [78]

\tilde{X}^2A_1 -Zustand. Da die nichtadiabatischen Effekte in der Nähe der konischen Überschneidung stark sind, ist zu erwarten, dass auch die niedrigliegenden vibronischen Anregungen des \tilde{A}^2B_2 -Zustands von den nichtadiabatischen Kopplungen beeinflusst werden.

Die Gleichgewichtsgeometrien von NO₂ in seinen verschiedenen elektronischen Zuständen sind von verschiedenen Autoren berechnet worden. In jeder Berechnung wurden leicht unterschiedliche Werte erhalten, von denen hier einige wenige erwähnt werden sollen: Die Gleichgewichtsgeometrie von Hirsch *et al.* [82] tritt bei $r = 1,203 \text{ Å}$ und $\phi = 133,6^\circ$ für den \tilde{X}^2A_1 -Zustand und bei $r = 1,277 \text{ Å}$ und $\phi = 101,9^\circ$ für den \tilde{A}^2B_2 -Zustand auf. Gillespie *et al.* [73] erhielten $r = 1,199 \text{ Å}$ und $\phi = 145,5^\circ$ für \tilde{X}^2A_1 und $r = 1,263 \text{ Å}$ und $\phi = 101,7^\circ$ für \tilde{A}^2B_2 . Aus den Resultaten in Tabelle 5.14 kann man ablesen, dass die von uns berechneten Geometrien gut mit den oben erwähnten Resultaten und mit den experimentellen Ergebnissen übereinstimmen. Für die strukturellen Daten des \tilde{A}^2B_2 -Zustands sind keine experimentellen Werte bekannt. Ihre Bestimmung ist schwierig, da das Minimum dieses Zustands sehr nah am $\tilde{X}^2A_1/\tilde{A}^2B_2$ -Cusp liegt und deshalb stark von den nichtadiabatischen Wechselwirkungen beeinflusst wird. Für die Energie des \tilde{A}^2B_2 -Minimums berechneten Gillespie *et al.* 1,18 eV; Hirsch *et al.* erhielten 1,08 eV. Aus Tabelle 5.14 kann man ablesen, dass unser Ergebnis (1,207 eV) näher an dem experimentellen Wert für diese Größe liegt. Der $\tilde{X}^2A_1/\tilde{A}^2B_2$ -Cusp wurde von Hirsch *et al.* bei $r = 1,259 \text{ Å}$ und $\phi = 107,4^\circ$ gefunden; Gillespie *et al.* berechneten $r = 1,26 \text{ Å}$ und $\phi = 114^\circ$. Die Energien, bei denen der Cusp gefunden wurde, betragen 1,14 eV und 1,48 eV bei Hirsch *et al.*, bzw. Gillespie *et al.*. Aus Tabelle 5.14 kann man ablesen, dass der von uns berechnete Wert von r leicht niedriger als die oben erwähnten Werte ist. Er ist jedoch näher an

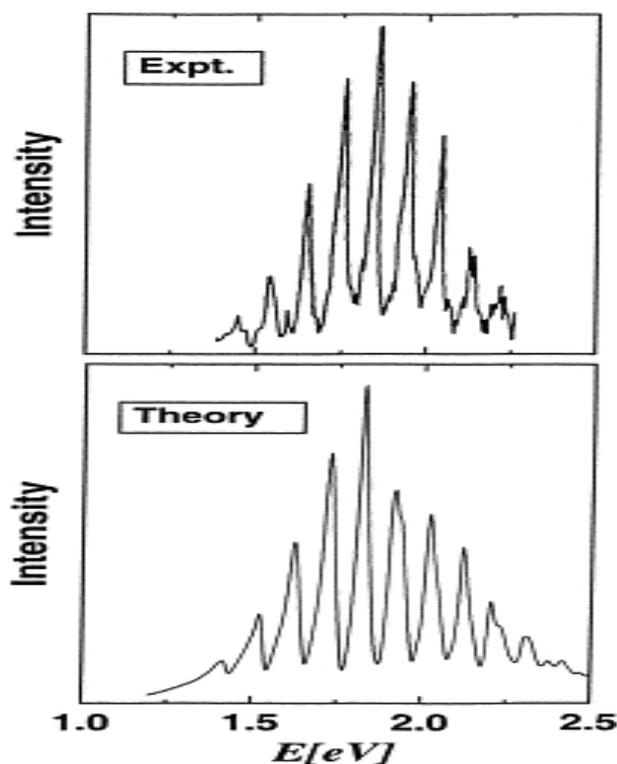


Abbildung 5.12: Photoemissionsspektrum von NO_2^- für einen Übergang zum \tilde{A}^2B_2 -Zustand von NO_2 . Unten ist das von Mahapatra und Köppel berechnete Spektrum dargestellt, oben die experimentellen Befunde von Weaver *et al.* [77].

den experimentellen Resultaten. Auch die von uns berechnete Energie am Cusp stimmt innerhalb der Fehlerbalken sehr gut mit den experimentellen Resultaten überein.

Ausgehend von der berechneten $\tilde{X}^2A_1/\tilde{A}^2B_2$ -Potentialenergieoberfläche haben Mahapatra und Köppel [83] das Photoemissionsspektrum von NO_2^- für einen Übergang zum \tilde{A}^2B_2 -Zustand von NO_2 mit zeitabhängiger Wellenfunktionspropagation berechnet. Die Details ihrer Untersuchungen gehen über den Rahmen dieser Arbeit hinaus. Das von Mahapatra und Köppel berechnete Photoemissionsspektrum ist in Abb. 5.12, verglichen mit den experimentellen Resultaten von Weaver *et al.*, dargestellt. Die berechneten Spektrallinien haben einen Abstand von ca. 0.1 eV, was einem Quant der Beugeschwingung von NO_2 im \tilde{A}^2B_2 -Zustand entspricht. Mahapatra und Köppel haben eine nichtadiabatische Kopplung zwischen dem \tilde{X}^2A_1 - und dem \tilde{A}^2B_2 -Zustand berücksichtigt. Diese führte zu einer Verbreiterung der berechneten Spektrallinien. Das berechnete Photoemissionsspektrum zeigt eine gute Übereinstimmung mit den experimentellen Werten.

Kapitel 6

Zusammenfassung

Methoden wie die MRCI, in denen dynamische Korrelationseffekte akkurat behandelt werden, hatten in den vergangenen Jahren einen bedeutenden Einfluss auf die Entwicklung der Quantenchemie. Da jedoch der Rechenaufwand dieser Methoden schnell mit der Systemgröße wächst ($\sim n_e^6$), kann nur durch die Verwendung der heutzutage leistungsfähigsten verfügbaren Rechnerarchitekturen sichergestellt werden, dass sie auch in Zukunft noch Verwendung finden können. In der absehbaren Zukunft wird es keine Rechnerarchitekturen geben, die einen größeren Durchsatz als massiv parallele Rechnerarchitekturen mit verteiltem Speicher haben. Es ist daher wünschenswert, für quantenchemische Methoden genau diese Rechnerarchitekturen nutzen zu können. Die Entwicklung einer skalierbaren Implementierung einer der populärsten Varianten der MRCI-Methode ist ein wichtiger Schritt in diese Richtung.

Während die Schwierigkeiten, eine Standard-MRCI-SD zu parallelisieren, die *alle* Ein- und Zweifachanregungen der Referenzen variationell behandelt, daraus resultieren, dass der Zustandsvektor auf alle Knoten des Parallelrechners verteilt werden muss, profitiert die zustandsselektierende MRCI von der Kompaktheit des selektierten Zustandsvektors. Schwierigkeiten für die Implementierung entstehen hier aus der Tatsache, dass der selektierte Subraum keinerlei Struktur aufweist. Die von uns entwickelte residuengetriebene Berechnung der Matrixelemente stellt eine effiziente Methode zur Konstruktion der von Null verschiedenen Matrixelemente dar. Der Residuenbaum ist gleichzeitig auch der Schlüssel zur erfolgreichen Parallelisierung des Algorithmus.

Wir haben Benchmarkrechnungen durchgeführt, die das gute Skalierungsverhalten des Programms auf der IBM SP/2, IBM SP/3-SMP und Cray-T3E, den wichtigsten heute verfügbaren Architekturen für Parallelrechner, demonstriert haben. In der ursprünglichen Version des Programms stellte die Notwendigkeit, auf jedem Knoten alle Integrale im Hauptspeicher zu halten, einen Flaschenhals für die Behandlung größerer Systeme dar. Diesen konnten wir durch die Entwicklung eines effizienten integralgetriebenen Algorithmus beseitigen. Das Skalierungsverhalten des Programms wurde hierdurch nur geringfügig verschlechtert.

Wir haben akkurate Berechnungen der Elektronenaffinitäten von BO, NO und O₂ durchgeführt, indem wir über die Standard-MRCI hinausgehend selektiv auch höhere Anregungen betrachtet haben. Für die Elektronenaffinitäten von BO und NO erhielten wir eine exzellente Übereinstimmung mit den experimentellen Daten, für O₂ verblieb eine Diskrepanz von 0,06 eV. Unsere Resultate korrigieren die lang bekannte Schwierigkeit der MRCI, sauerstoffhaltige Anionen korrekt zu beschreiben. Auch der Grund für diese Schwierigkeit wird aufgezeigt: Höhere Anregungen in Nicht-Valenzorbitale tragen grundlegend zu differentiellen Effekten in der elektronischen Struktur bei. Aus diesem Grund ist es der Standard MRCI auch im Basissatzlimit nicht möglich, die Elektronenaffinitäten quantitativ richtig wiederzugeben. Quantitativ fast richtige Ergebnisse erhält man mit der MR-ACPF Methode, in der die höheren Anregungen zwar approximativ, aber selbstkonsistent behandelt werden.

Zu einer Untersuchung der Bergmanschen Ringschlußreaktion des einfachsten Mitglieds der Endiin-Familie wurden wir dadurch motiviert, dass eine frühere MRCI-Studie nicht einmal qualitative Übereinstimmung mit dem Experiment erreichte. Außerdem sind in der Literatur bislang keine Ergebnisse in einer Basis mit mehr als *double zeta*-Qualität zu finden. Unsere Ergebnisse, die wir in cc-pVDZ und cc-pVTZ Basissätzen berechnet haben, stimmen bis auf maximal 1 kcal/mol mit den experimentellen Werten überein.

Den Beginn einer systematischen Untersuchung der Übergangsmetallhalogenide markiert die Berechnung der adiabatischen und vertikalen Anregungsenergien von VF₂ und VCl₂. Im Hinblick auf Untersuchungen weiterer Übergangsmetallhalogenide haben wir für VF₂ den Einfluß von Basissatz, Größe des aktiven Raums und Valenz-Core-Korrekturen untersucht. Der Einfluß dieser Effekte auf die berechneten Anregungsenergien war in allen Fällen gering. Die von uns berechneten vertikalen Anregungsenergien zeigten eine exzellente Übereinstimmung mit Energien, die wir mit dem intern-kontrahierten MRCI-Programm MOLPRO berechnet haben. Die qualitative Reihenfolge der Zustände, die in einer aktuellen DFT-Studie berechnet wurde, konnte von uns reproduziert werden, jedoch zeigen sich große quantitative Unterschiede in den berechneten Anregungsenergien.

Die spektroskopischen Eigenschaften von NO₂ sind außerordentlich komplex. Der Grund dafür ist eine konische Überschneidung zwischen Grundzustand und erstem angeregten Zustand. Um das Photoelektronenspektrum von NO₂ theoretisch vorherzusagen, ist eine genaue Kenntniss der Potentialenergieoberfläche notwendig. In der Literatur sind bislang nur Potentialenergieoberflächen von *double zeta*-Qualität bekannt. Die von uns berechnete cc-pVTZ PES wurde von Mahapatra und Köppel benutzt, um die spektralen Eigenschaften von NO₂ mit Hilfe zeitabhängiger Wellenfunktionspropagation zu berechnen. Das theoretisch berechnete Photoelektronenspektrum zeigt eine gute Übereinstimmung mit den experimentellen Werten.

Zusammenfassend kann man feststellen, dass es gelungen ist, eine effiziente vollständig parallelisierte Implementierung der zustandsselektierenden MRCI zu entwickeln. Die vorliegende Implementierung erlaubt die Behandlung von Hilberträumen und Systemen, die

Implementierungen der zustandsselektierenden MRCI auf traditionellen Rechnerarchitekturen verschlossen sind. Die Laufzeit für mittelgroße Systeme wird gegenüber anderen MRCI-Implementierungen signifikant reduziert. Mit der Möglichkeit Hilberträume bis zu einer Größe von 10^9 Konfigurationen zu behandeln, von denen bis zu $2 \cdot 10^7$ für die Variationsrechnung selektiert werden können, erschließt sich der MRCI ein breites Spektrum von Problemstellungen, die eine ausgewogene Behandlung von statischen und dynamischen Korrelationseffekten erfordern.

Literaturverzeichnis

- [1] W. J. Hehre, L. Radom, P. v. R. Schleyer, and J. A. Pople, *Ab Initio Molecular Orbital Theory*, Wiley, (New York 1986). 2.1
- [2] R. J. Buenker and S. D. Peyerimhoff, *Theor. Chim. Acta* **12**, 183 (1968). 2.1
- [3] R. J. Buenker and S. D. Peyerimhoff, *Theor. Chim. Acta* **35**, 33 (1974). 2.1
- [4] R. J. Buenker and S. D. Peyerimhoff, *Theor. Chim. Acta* **39**, 217 (1975). 2.1
- [5] P. Stampfuß, H. Keiter, and W. Wenzel, *J. Comput. Chem.* **20**, 1559 (1999). 2.1
- [6] E. R. Davidson, *J. Comp. Physics* **17**, 87 (1975). 2.2
- [7] P.-O. Löwdin, *Phys. Rev.* **97**, 1509 (1955). 2.3
- [8] C. Bender and E. Davidson, *J. Chem. Phys.* **47**, 4972 (1967). 2.3
- [9] W. Wenzel and K. G. Wilson, *Phys. Rev. Letters* **68**, 800 (1992). 2.4
- [10] E. R. Davidson, in R. Daudel and B. Pullmann, editors, *The world of Quantum Chemistry*, Reidel, Dordrecht, (1974). 2.5
- [11] S. R. Langhoff and E. R. Davidson, *Int. J. Quantum Chem.* **8**, 61 (1974). 2.5
- [12] J. A. Pople, R. Seeger, and R. Krishnan, *Int. J. Quantum Chem. Symp.* **11**, 149 (1977). 2.5
- [13] L. Meissner, *Chem. Phys. Letters* **146**, 204 (1988). 2.5
- [14] W. Butscher, S. Shih, R. J. Buenker, and S. D. Peyerimhoff, *Chem. Phys. Letters* **52**, 457 (1977). 2.5, 5.1
- [15] P. J. Bruna, S. D. Peyerimhoff, and R. J. Buenker, *Chem. Phys. Letters* **72**, 278 (1980). 2.5
- [16] J. Olsen, P. Jorgensen, H. Koch, A. Balkova, and R. J. Bartlett, *J. Chem. Phys.* **104**, 8007 (1996). 2.5, 2.6
- [17] T. H. Dunning, *J. Chem. Phys.* **90**, 1007 (1989). 2.5, 5.1, 5.2, 5.4

- [18] R. Gdanitz and R. Ahlrichs, Chem. Phys. Letters **143**, 413 (1988). 2.5
- [19] P. Szalay and R. J. Bartlett, J. Chem. Phys. **103**, 3600 (1995). 2.5
- [20] J. Sánchez-Marín, I. Nebot-Gil, J. P. Malrieu, J. L. Heully, and D. Maynau, Theor. Chim. Acta **95**, 215 (1997). 2.5
- [21] J. P. Daudey, J.-L. Heully, and J. P. Malrieu, J. Chem. Phys. **99**, 1240 (1993). 2.5
- [22] J. P. Malrieu, J. P. Daudey, and R. Caballol, J. Chem. Phys. **101**, 8908 (1994). 2.5
- [23] J. Meller, J. P. Malrieu, and J. L. Heully, Chem. Phys. Letters **244**, 440 (1995). 2.5
- [24] F. Stephan and W. Wenzel, J. Chem. Phys. **108**, 1015 (1998). 3.1
- [25] S. Krebs and R. J. Buenker, J. Chem. Phys. **103**, 5613 (1995). 4.3
- [26] M. Hanrath and B. Engels, CP **225**, 197 (1997). 4.3
- [27] H.-J. Werner and P. Rosmus, in R. Bartlett, editor, *Comparison of Ab-Initio Quantum Chemistry with Experiment*, Reidel, Dordrecht, (1985). 5.1
- [28] L. A. Curiss, R. Raghavachari, G. W. Trucks, and J. A. Pople, J. Chem. Phys. **94**, 7221 (1991). 5.1
- [29] D. E. Jensen, JCP **52**, 3305 (1970). 5.1
- [30] R. D. Srivastava, O. Uy, and M. Farber, Trans. Faraday Soc. **67**, 2941 (1971). 5.1
- [31] J. Rienstra and H. S. III, J. Chem. Phys. **106**, 4472 (1997). 5.1, 5.1, 5.1
- [32] P. G. Wentholt, J. Kim, K.-L. Jones, and W. C. Lineberger, J. Phys. Chem. A **101**, 4472 (1997). 5.1, 5.1, 5.1
- [33] A. Papakondylis and A. Mavridis, Chem. Phys. Letters **341**, 382 (2001). 5.1, 5.1, 5.1
- [34] M. J. Travers, D. C. Cowles, and G. B. Ellison, Chem. Phys. Letters **164**, 449 (1989). 5.1, 5.1
- [35] R. González-Luque, M. Merchán, M. P. Fülcher, and B. O. Roos, Chem. Phys. Letters **204**, 323 (1993). 5.1, 5.1
- [36] R. A. Kendall, T. H. Dunning, and R. J. Harrison, J. Chem. Phys. **96**, 6796 (1992). 5.1, 5.2
- [37] M. C. McCarthy, J. W. R. Allington, and K. Griffith, Chem. Phys. Letters **289**, 156 (1998). 5.1
- [38] M. Tanimoto, S. Saito, and E. Hirota, J. Chem. Phys. **84**, 1210 (1986). 5.1

- [39] K. P. Huber and G. Herzberg, *Constants of Diatomic Molecules*, Van Nostrand, New York, (1979). 5.1
- [40] W. Roth, H. Hopf, and C. Horn, *Chem. Ber.* **127**, 1765 (1994). 5.5
- [41] J. Gräfenstein, A. Hierpe, E. Kraka, and D. Cremer, *J. Phys. Chem* **104**, 1748 (2000). 5.5, 5.2
- [42] E. Kraka and D. Cremer, *JACS* **116**, 4929 (1994). 5.2
- [43] E. Kraka and D. Cremer, *JACS* **122**, 8245 (2000). 5.2, 5.2
- [44] N. Koga and K. Morokuma, *JACS* **113**, 1907 (1991). 5.2
- [45] R. A. Kent, J. D. McDonald, and J. L. Margrave, *JPC* **70**, 874 (1960). 5.3
- [46] C. E. Moore, *Atomic Energy Levels*, volume 2 of *NSRDS-NBS#35*, US GPO, (Washington, DC 1971). 5.3
- [47] F. Ramondo, V. Rossi, and L. Bencivenni, *J. Mol. Struct.* **64**, 513 (1988). 5.3
- [48] F. Ramondo, L. Bencivenni, and S. N. Cesaro, *J. Mol. Struct.* **192**, 83 (1989). 5.3
- [49] V. P. Spiridonov, A. G. Gershikov, and V. S. Lyutsarev, *J. Mol. Struct* **79**, 220 (1990). 5.3
- [50] O. V. Boltalina, A. Y. Borshchevskii, and L. N. Sidorov, *Russ. J. Phys. Chem* **65**, 466 (1991). 5.3
- [51] O. V. Boltalina, A. Y. Borshchevskii, and L. N. Sidorov, *Russ. J. Phys. Chem* **66**, 1223 (1992). 5.3
- [52] A. J. Ross, R. Bacis, A. J. Bouvier, S. Churassy, J. C. Coste, P. Crozet, and I. Russier, *J. Mol. Spectrosc.* **158**, 27 (1993). 5.3
- [53] M. P. Barnes, R. T. Carter, N. M. Lakin, and J. M. Brown, *J. Chem. Soc. Faraday Trans.* **89**, 3205 (1993). 5.3
- [54] D. L. Hildenbrand, *JCP* **103**, 2634 (1995). 5.3
- [55] D. L. Hildenbrand, *High Temp. Mater. Sci.* **35**, 151 (1996). 5.3
- [56] S. Y. Shashkin and W. A. Goddard, *JPC* **90**, 225 (1986). 5.3
- [57] C. W. Bauschlicher and B. O. Roos, *J. Chem. Phys.* **91**, 4785 (1989). 5.3
- [58] M. Kaupp, P. R. Schleyer, H. Stroll, and H. Preuss, *JACS* **113**, 6012 (1991). 5.3
- [59] R. J. Deeth, *J. Chem. Soc. Dalton Trans.* page 1061 (1993). 5.3
- [60] H. C. F. Rogemond and D. R. Salahub, *CPL* **219**, 228 (1994). 5.3

- [61] M. Hassett and C. J. Marsden, *J. Mol. Struct.* **346**, 249 (1995). 5.3
- [62] B. O. Roos, K. Andersson, M. P. Fülcher, P.-A. Malmqvist, L. Serrano-Andres, K. Pierloot, and M. Merchán, *Adv. Chem. Phys.* **93**, 219 (1996). 5.3
- [63] D. R. S. C. Blanchet, H.A. Durante, *J. Chem. Phys.* **106**, 8778 (1997). 5.3
- [64] S. G. Wang and W. H. E. Schwartz, *J. Chem. Phys.* **109**, 7252 (1998). 5.3, 5.3, 5.3
- [65] K. Kuchitsu, *Landolt-Börnstein: Structure Data of Free Polyatomic Molecules*, Springer-Verlag, Berlin, (1992). 5.3, 5.3, 5.3
- [66] R. Pou-Amerigo, M. Merchán, I. Nebot-Gil, P. Widmark, and B. Roos, *Theor. Chim. Acta* **92**, 149 (1995). 5.8, 5.9, 5.10, 5.11, 5.3, 5.3
- [67] P. Widmark, P. Malmqvist, and B. Roos, *TCA* **77**, 291 (1990). 5.8, 5.9, 5.10, 5.11, 5.3, 5.3
- [68] H. Köppel, W. Domcke, and L. S. Cederbaum, *Adv. Chem. Phys.* **57**, 59 (1984). 5.4
- [69] R. L. Whetten, G. S. Ezra, and E. R. Grant, *Annu. Rev. Phys. Chem.* **36**, 277 (1985). 5.4
- [70] I. B. Bersuker and V. Z. Polinger, *Vibronic Interactions in Molecules and Crystals*, Spinger, Berlin, (1989). 5.4
- [71] W. Domcke and G. Stock, *Adv. Chem. Phys.* **100**, 1 (1997). 5.4
- [72] L. S. C. T. Pacher, *Adv. Chem. Phys.* **84**, 293 (1993). 5.4
- [73] G. D. Gillispie, A. Khan, A. C. Whal, R. P. Hosteney, and M. Krauss, *J. Chem. Phys.* **63**, 3425 (1975). 5.4, 5.4
- [74] C. F. Jackels and E. R. Davidson, *JCP* **64**, 2908 (1976). 5.4
- [75] C. Jungen and A. J. Merer, in *Molecular Spectroscopy: Modern Research*, page 127, K. Narahari Rao, (academic press, new york edition, (1976). 5.4
- [76] J. L. Hardwick and J. C. D. Brand, *CPL* **21**, 458 (1973). 5.4, 5.14
- [77] A. Weaver, R. B. Metz, S. E. Bradforth, and D. M. Neumark, *J. Chem. Phys.* **90**, 2070 (1989). 5.4, 5.12
- [78] K. M. Ervin, J. Ho, and W. C. Lineberger, *J. Phys. Chem.* **92**, 5405 (1988). 5.4, 5.14
- [79] Y. Morino, M. Tanimoto, S. Saito, E. Hirota, R. Awata, and T. Tanaka, *J. Mol. Spectry.* **98**, 331 (1983). 5.14
- [80] E. R. Davidson, *Int. J. Quantum Chem.* **8**, 83 (1974). 5.14

- [81] M. Krauss, R. J. Cellota, S. R. Mielczarek, and C. E. Kuyatt, *CPL* **27**, 285 (1974).
5.14
- [82] G. Hirsch, R. Buenker, and C. Petrongolo, *Mol. Phys.* **73**, 1085 (1991). 5.4
- [83] S. Mahapatra, H. Köppel, L. Cederbaum, P. Stampfuß, and W. Wenzel, *Chem. Phys.*
259, 211 (2000). 5.4

An dieser Stelle möchte ich allen danken, die mich bei der Entstehung und Durchführung dieser Arbeit hilfreich unterstützt haben.

Insbesondere danke ich Herrn Priv.-Doz. Dr. Wolfgang Wenzel für die Themenstellung und für die vielen Diskussionen, denen wichtige Impulse für meine Arbeit entsprungen sind. Ohne die Bereitstellung seines Quantenchemie-Programmpaketes wäre die Durchführung dieser Arbeit nicht möglich gewesen. Desgleichen danke ich auch den übrigen Personen, die an der Entstehung des Quantenchemiepakets mitgearbeitet haben: Dr. Matthew Steiner, Dr. Frank Stephan und Dr. Heiko Krabbe.

Mein besonderer Dank gilt ebenfalls Herrn Prof. Dr. Hellmut Keiter für seine Unterstützung und sein Interesse am Zustandekommen dieser Arbeit.

Für die hilfreichen Diskussionen über Übergangsmetallverbindungen und Gruppentheorie danke ich Michael Vogel und Dr. Nickolay Dobrodey.

Jesko Sirker und Dr. Bernd Burghardt danke ich, da sie die mühevollen Aufgabe übernommen haben, diese Arbeit korrekturlesen. Dr. D. Lüke danke ich, die verbliebenen Schreibfehler aufgedeckt zu haben.

Weiterhin bin ich allen anderen Mitgliedern des Lehrstuhls "Theoretische Physik I" für die kollegiale Atmosphäre sowie ihre ständige Diskussions- und Hilfsbereitschaft zu Dank verpflichtet.