

Impressive title littered with flowery and technically dense language

by

Benny Chen

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Science
in
Chemistry

Waterloo, Ontario, Canada, 2024

© Benny Chen 2024

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Abstract

A diabaticization method utilizing GAMESS multi-configurational quasidegenerate perturbation theory for purposes of obtaining spin-orbit coupling and n-th order couplings was implemented in Python. Iron pentacarbonyl was studied with a triple zeta polarized Sapporo basis set.

Acknowledgements

Neil's tips for thesis writing: - Start out with where you are confident in, say the script and completed models - If you get stuck on compiling say a table, just take a photo of a handwritten one and stick it in as temporary placeholder - Have many rough passes

Dedication

Contents

List of Figures	ix
List of Tables	x
List of Illustrations	xi
List of Abbreviations	xii
List of Symbols	xiii
Nomenclature	xiv
Introduction	1
1 Introduction	1
1.1 Motivation	1
1.1.1 Open-shell transition metal complex spectra and thermochemistry . .	1
1.2 Objectives	1
1.2.1 Vibronic Models: Quadratic-order CASSCF GMCPT SOC Protocol .	1
Methods and Context	2
2 Methods and Context	2
2.1 Electronic structure theory	2
2.1.1 TDSE, BOA	3
2.1.2 Adiabatic and Diabatic Basis Connection	6
2.1.3 MCSCF, CASSCF, GMC-QDPT	8
2.2 Diabatization	9
2.2.1 Adiabats and diabats	10
2.2.2 TD-DFT excited states model limitations	10
2.3 Calculation of Spectra	10
2.3.1 Time-correlation functions	10

2.3.2	MCTDH	10
2.3.3	VECC	10
2.4	Relativistic effects	10
2.4.1	Spin Orbit Coupling (SOC)	10
2.5	Thermal effects	10
2.5.1	TF-VECC: Thermal Field Vibrational Electronic CC	10
2.6	Computational Details	10
2.6.1	GAMESS-US	11
2.6.2	nlogn, ComputeCanada (Computational Resources)	12
Vibronic Model Development		14
3	Vibronic Model Development	14
3.1	CASSCF	15
3.2	Diabatization protocol Python code explanation	15
3.2.1	diabatize() and mctdh() functions	19
3.2.2	Distorting coordinates and calculating up to quadratic coupling constants	19
3.2.3	Fitting on n-th order models	19
3.2.4	Generating spectra via MCTDH and VECC	19
Full Computational Study on Large System		20
4	Full Computational Study on Large System	20
4.1	Results	20
4.1.1	Vibronic spectra of test case pnictogen hydrides	20
4.1.2	Spectra of D_{3h} SOC transition metal complexes (CoF_3 and RhF_3) . .	20
4.2	Iron pentacarbonyl, $\text{Fe}(\text{CO})_5$	20
References / Bibliography		22
Appendices		23
Glossary		24
Index		25
4.2.1	https://pubs.acs.org/doi/10.1021/jacs.2c01469	26
4.2.2	https://pubs.acs.org/doi/10.1021/jp992474u	26
4.2.3	https://pubs.aip.org/aip/jcp/article/75/6/2560/791508/The-Jahn-Teller-effect-in-the-photoelectron	26

List of Figures

List of Tables

2.1	Step 1 Information	12
4.1	Character Table for the D_{3h} Point Group	21

List of Illustrations

List of Symbols

SOC - Spin Orbit Coupling

GAMESS, GMS - General Atomic and Molecular Electronic Structure System

CC - ComputeCanada

CAS(m,n), 2e1o - Complete Active Space m Electrons and n Orbitals Configuration

DMO - Diabatic Molecular Orbitals

GMC-QDPT - Generalized Multiconfiguration - Quasi-Degenerate Perturbation Theory

MR - Multireference

MC - Multiconfiguration

SCF - Self-consistent Field

FeCO - Iron Pentacarbonyl

Nomenclature

Chapter 1

Introduction

1.1 Motivation

1.1.1 Open-shell transition metal complex spectra and thermochemistry

1.2 Objectives

1.2.1 Vibronic Models: Quadratic-order CASSCF GMCPT SOC Protocol

Fancy method, otherwise why not just DFT B3LYP 6-31G* @ everything ...

Analogy: we did not build the car, we merely are a driver, make sure we diligently perform routine maintenance on it, know what engine is and how to fill gas, etc.

We do not construct a car from the ground up, not feasible... monumental task

Chapter 2

Methods and Context

2.1 Electronic structure theory

2.1.1 TDSE, BOA

Let the full molecular Schrödinger's Equation be defined as:

$$\hat{H}(\vec{R})\Psi_n(\vec{r}; \vec{R}) = E_n\Psi_n(\vec{r}; \vec{R}) \quad (2.1)$$

The full molecular Hamiltonian:

$$\hat{H} = -\sum_{\alpha}^M \frac{1}{2M_{\alpha}} \nabla_{\alpha} \cdot \nabla_{\alpha} - \sum_i^N \frac{1}{2m_e} \nabla_i \cdot \nabla_i + \sum_{i,j>i}^N \frac{1}{r_{ij}} + \sum_{i,\alpha}^{N,M} \frac{-Z_A}{r_{i\alpha}} + \sum_{\alpha,\beta \neq \alpha}^M \frac{Z_A Z_B}{R_{\alpha\beta}} \quad (2.2)$$

Which can concisely be represented by kinetic and potential energy operators \hat{T} and \hat{V} :

$$\hat{H} = \hat{T}_N + \hat{T}_e + \hat{V}_{ee} + \hat{V}_{Ne} + \hat{V}_{NN} \quad (2.3)$$

Note the subscripts e, N are for electronic and nuclear correspondingly.

$$\hat{H} = \hat{T}_N + \hat{H}_{elec} \quad (2.4)$$

$$\hat{H}_{elec} = \hat{T}_e + \hat{V}_{ee} + \hat{V}_{Ne} + \hat{V}_{NN} \quad (2.5)$$

Let the wavefunction, Ψ , be the sum over the electronic eigenstates $\phi_{\lambda}(\vec{r}; \vec{R})$ and nuclear eigenstates $\chi_{\lambda}(\vec{R})$:

$$\Psi(\vec{r}; \vec{R}) = \sum_{\lambda} \phi_{\lambda}(\vec{r}; \vec{R}) \chi_{\lambda}(\vec{R}) \quad (2.6)$$

Here, \vec{r} indicates electron coordinates, and \vec{R} refers to nuclear configuration. The subscripts λ and μ denote electronic states. To describe a parametric dependence of \vec{r} on \vec{R} , $(\vec{r}; \vec{R})$ is used. Signifying that although an explicit dependence on \vec{R} is not present in the function, \vec{r}

values are predicated at a single \vec{R} parameter [szabo2012modern]. Therefore, if \vec{R} changes, so does $\phi_\lambda(\vec{r}; \vec{R})$.

Proceed with full expansion of Hamiltonian acting on the wavefunction:

$$\begin{aligned} & \hat{H}(\vec{R})\Psi(\vec{r}; \vec{R}) \\ &= \left[-\sum_{\alpha} \frac{1}{2M_{\alpha}} \vec{\nabla}_{\alpha} \cdot \vec{\nabla}_{\alpha} + \hat{H}_{elec} \right] \left[\sum_{\lambda} \phi_{\lambda}(\vec{r}; \vec{R}) \chi_{\lambda}(\vec{R}) \right] \end{aligned} \quad (2.7a)$$

$$\begin{aligned} &= \left(-\sum_{\alpha} \frac{1}{2M_{\alpha}} \right) \sum_{\lambda} [\chi_{\lambda}(\vec{R}) \vec{\nabla}_{\alpha}^2 \phi_{\lambda}(\vec{r}; \vec{R}) + \vec{\nabla}_{\alpha} \phi_{\lambda}(\vec{r}; \vec{R}) \vec{\nabla}_{\alpha} \chi_{\lambda} \\ &\quad + \phi_{\lambda}(\vec{r}; \vec{R}) \vec{\nabla}_{\alpha}^2 \chi_{\lambda}(\vec{R}) + \vec{\nabla}_{\alpha} \chi_{\lambda} \vec{\nabla}_{\alpha} \phi_{\lambda}(\vec{r}; \vec{R}) + E_{\lambda}(\vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) \chi_{\lambda}(\vec{R})] \end{aligned} \quad (2.7b)$$

$$\begin{aligned} &= \left(-\sum_{\alpha} \frac{1}{2M_{\alpha}} \right) \sum_{\lambda} [\chi_{\lambda}(\vec{R}) \vec{\nabla}_{\alpha}^2 \phi_{\lambda}(\vec{r}; \vec{R}) \\ &\quad + \phi_{\lambda}(\vec{r}; \vec{R}) \vec{\nabla}_{\alpha}^2 \chi_{\lambda}(\vec{R}) + 2\vec{\nabla}_{\alpha} \phi_{\lambda}(\vec{r}; \vec{R}) \vec{\nabla}_{\alpha} \chi_{\lambda} + E_{\lambda}(\vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) \chi_{\lambda}(\vec{R})] \end{aligned} \quad (2.7c)$$

Integrating against $\phi_{\mu}(\vec{r}; \vec{R})$ over all electronic coordinates, \vec{r} , and asserting that these states are orthonormal at a single fixed \vec{R} , produces a sum of four terms:

$$= \sum_{\lambda} \int \phi_{\mu}^*(\vec{r}; \vec{R}) \sum_{\alpha} -\frac{1}{2M_{\alpha}} \vec{\nabla}_{\alpha}^2 \phi_{\lambda}(\vec{r}; \vec{R}) dr \chi_{\lambda}(\vec{R}) \quad (\mathbf{A})$$

$$+ \sum_{\lambda} \sum_{\alpha} \delta_{\lambda\mu} \frac{-1}{2M_{\alpha}} \vec{\nabla}_{\alpha}^2 \chi_{\lambda}(\vec{R}) \quad (\mathbf{B})$$

$$+ \sum_{\lambda} \sum_{\alpha} \int \phi_{\mu}^*(\vec{r}; \vec{R}) \vec{\nabla}_{\alpha} \phi_{\lambda}(\vec{r}; \vec{R}) dr \cdot -\frac{1}{2M_{\alpha}} \vec{\nabla}_{\alpha} \chi_{\lambda}(\vec{R}) \quad (\mathbf{C})$$

$$+ \sum_{\lambda} \delta_{\lambda\mu} E_{\lambda}(\vec{R}) \chi_{\lambda}(\vec{R}) \quad (\mathbf{D})$$

$$= \sum_{\lambda} E_{\lambda}(\vec{R}) \delta_{\lambda\mu} \chi_{\lambda}(\vec{R}) \quad (2.8)$$

The BOA facilitates setting the expansion in terms of a single electronic state, hence $\lambda = \mu$. Terms **A** and **C** in the BOA are also ignored. The **A** term represents a diagonal Born-Oppenheimer correction to potential, useful in specialized instances—as in accounting for different M_{α} (e.g. isotopes) [nooijennotes]. It is typically proximate enough to zero

for it to be neglected. Likewise, \mathbf{C} is zero for real normalized wavefunctions. \mathbf{C} is referred to as the off-diagonal non-adiabatic coupling (NAC) terms. In other words, only \mathbf{B} and \mathbf{D} terms remain because terms with the presence of $\int d\vec{r} \vec{\nabla}_\alpha \phi_\lambda(\vec{r}; \vec{R})$ are effectively zero if there exists little change in ϕ_λ with respect to \vec{R} , thus implying no coupling between electronic wavefunctions and nuclear configuration [nooijennotes].

As for a small example, so far the BOA shows:

$$\begin{bmatrix} \hat{T}_N + E_1(\vec{R}) + A_1(\vec{R}) & N.A.C. \\ N.A.C. & \hat{T}_N + E_2(\vec{R}) + A_2(\vec{R}) \end{bmatrix} \quad (2.9)$$

Ultimately, the Schrödinger equation with the BOA applied appears as:

$$\left[\hat{T}_N + E_\lambda(\vec{R}) \right] \Psi_{\lambda,n}(\vec{R}) = E_n \Psi_{\lambda,n}(\vec{R}) \quad (2.10)$$

Where label n is for rotational-vibrational levels and recalling λ is for electronic states. Attention to the term $E_\lambda(\vec{R})$, which is a function of nuclear configuration. $E_\lambda(\vec{R})$ is defined to be a single point sufficiently able to define a potential energy surface (PES), $E_\lambda(\{\vec{R}\})$. Now the electronic wavefunction $\phi_\lambda(\vec{r}; \vec{R})$, wavefunction $\chi_n(\vec{R})$ (detailing nuclear motion), and potential energy surface $E_\lambda(\vec{R})$ in eq:10 can be solved for at a single electronic state λ with any given fixed-geometry \vec{R} . In practice of using BOA concepts, fig:fig1 demonstrates how the electronic ground state PES can be theoretically constructed. Understanding surfaces in this fashion gives rise to adiabatic states or true Born-Oppenheimer surfaces [villanueva2020spectroscopic].

2.1.2 Adiabatic and Diabatic Basis Connection

Earlier discussions on the importance of non-adiabatic dynamics to vibronic models and spectra can now deepen. Although the BOA was highlighted to reduce calculation complexity (i.e. avoiding evaluation of integral terms), it fails when the NACs are relevant—fig:fig2(a), occurring mainly in two situations:

- i. The adiabatic state character changes rapidly as a function of geometry.
- ii. Degenerate electronic states, separations on the order of vibrational energy level values.

When these conditions exist, Yarkony has described how conical intersections and avoided crossings between separate single excited states develop, and in order to simulate spectra successfully, the model needs to account for neglected terms and be governed by multiple PES to describe nuclear motion [**yarkony2001conical**].

Recall, term **C** contains an integral, which is inversely proportional to energy separation:

$$\int \phi_\mu^*(\vec{r}; \vec{R}) \vec{\nabla}_\alpha \phi_\lambda(\vec{r}; \vec{R}) d\vec{r} \propto \frac{1}{E_\mu - E_\lambda} \quad (2.11)$$

Ultimately, the Schrödinger equation with the BOA applied appears as:

$$\left[\hat{T}_N + E_\lambda(\vec{R}) \right] \Psi_{\lambda,n}(\vec{R}) = E_n \Psi_{\lambda,n}(\vec{R}) \quad (2.12)$$

$$\int \phi_\mu^*(\vec{r}; \vec{R}) \vec{\nabla}_\alpha \phi_\lambda(\vec{r}; \vec{R}) d\vec{r} \propto \frac{1}{E_\mu - E_\lambda} \quad (2.13)$$

$$\begin{bmatrix} \hat{T}_N + E_1(\vec{R}) + A_1(\vec{R}) & N.A.C. \\ N.A.C. & \hat{T}_N + E_2(\vec{R}) + A_2(\vec{R}) \end{bmatrix} \Rightarrow \begin{bmatrix} \hat{T}_N + E_{11}(\vec{R}) & E_{12}(\vec{R}) \\ E_{21}(\vec{R}) & \hat{T}_N + E_{22}(\vec{R}) \end{bmatrix} \quad (2.14)$$

Obtaining this matrix of diabatic form, it can be utilized to describe potentials and a coupled set of vibrational wavefunctions $\chi_a(\vec{R})$:

$$\begin{pmatrix} \hat{T}_N + E_{11}(\vec{R}) & E_{12}(\vec{R}) \\ E_{21}(\vec{R}) & \hat{T}_N + E_{22}(\vec{R}) \end{pmatrix} \begin{pmatrix} \chi_1(\vec{R}) \\ \chi_2(\vec{R}) \end{pmatrix} = E \begin{pmatrix} \chi_1(\vec{R}) \\ \chi_2(\vec{R}) \end{pmatrix} \quad (2.15)$$

$$\Psi_a(\vec{r}; \vec{R}) = \sum_{\lambda} U_{\lambda a}(\vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) \quad (2.16)$$

$$S_{\lambda\mu}(\vec{R}) = \int \phi_{\mu}^*(\vec{r}; \vec{R}_{q_0}) \phi_{\lambda}(\vec{r}; \vec{R}_{q_0 \pm dq_i}) \quad (2.17)$$

$$\sum_{\lambda} \int \phi_{\mu}^*(\vec{r}; \vec{R}) \hat{H}_{elec} \Psi_a(\vec{r}; \vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) d\vec{r} \chi_{\lambda}(\vec{R}) = \sum_{\lambda} E_{\mu\lambda} \chi_{\lambda}(\vec{R}) \quad (2.18)$$

$$E_{\mu\lambda}(q) = E_{\mu\lambda}(0) + \sum_i^M E_{\mu\lambda}^i q_i + \frac{1}{2} \sum_{i>j}^M E_{\mu\lambda}^{ij} q_i q_j + \dots \quad (2.19)$$

$$E_{\mu\lambda}^i = \frac{E_{\mu\lambda}(\vec{R}_{q_0} + dq_i) - E_{\mu\lambda}(\vec{R}_{q_0} - dq_i)}{2dq_i} \quad (2.20)$$

2.1.3 MCSCF, CASSCF, GMC-QDPT

Diabatic Molecular Orbitals

Configurational uniformity is a concept devised by Hisao Nakamura and Donald Truhlar to manufacture diabats from adiabats. In their framework, which expands on previous work by Atchity and Rudenberg, diabaticization-adapted molecular orbitals (DMOs) are obtained and then subsequent DSFs (diabatic state functions) are constructed. The idea is to optimize for uniformity in electronic configurations that originate from DMOs.

GMC-QDPT

Haruyuki Nakano and coworkers have devised the Generalized Multiconfiguration - Quasi Degenerate Perturbation Theory (GMC-QDPT) scheme. GMC-QDPT itself is classified as a composite method that aims to recover both static and dynamical correlation. The 'generalized' label refers to zero-restrictions in variational space. This means beyond CASSCF and RASSCF, so general MCSCF wavefunctions are applicable for submission as input. The QDPT latter half indicates a subsequent multireference calculation that proceeds. In short, it is intended for multistate calculations, and optimal for heavy/relativistic calculation purposes. Interestingly, when benchmarked with CAS-QDPT, there is only a 0.1 eV and 1 kcal/mol Δ . Nakano has prepared an excellent student reference guide for performing GAMESS GMCPT and GMC-QDPT calculations: <https://ccl.scc.kyushu-u.ac.jp/~nakano/gmcpt.html>. An important consideration is that FCI is often intractable due to the computational cost growing factorially in terms of reference space determinants, so this efficient scheme is more feasible instead.

Active Space Design

Designing the active space is a pedantic problem.

Cool paper: <https://arxiv.org/pdf/2111.15587>: :text=ORMAS

2.2 Diabatization

Loops in python are notoriously slow. So in collaboration with Neil Raymond, we refactored the excessive looping that distorted the modes into array operations. This drastically improved the performance. Furthermore, it also made the entire program more dynamic in a sense that it can adapt and be extended for a variety of purposes. An example here is fitting. This situation arose where distortion beyond-quadratic-order at a single mode was needed, and the recasted arrays handled this trivially as compared to looping. Another key aspect of this change was the fact that the arrays forced us to understand exactly what dimensions are required in the diabatization code’s various calculation modules. We have stitched together various functionalities such as: text processing, coordinate distortion, input file generation, GAMESS job submission, and operator file generation.

We originally opted to extract data using the slow pythonic ‘read’ method. However, this proved to be debilitating when operating on large output files to a point where the bottleneck could be hours when running the script. Therefore we opted to use grep instead. When it came down to the full computational study on FeCO, surprisingly even grep was deemed too slow. The average size of the FeCO output file (there are 1000 outputs) is 70 megabytes or around 2 million lines of text. When we grepped the SOC data, it took around 4 hours. So finally we pivoted to using memory-mapping. This not only made the program more performant, reducing the runtime of the script processing all 28 modes of FeCO from 10300 seconds (with grep) to a mere hour.

If diabatic energy, we need to extract along the diagonal of the states array (a,a). If diabatic coupling, we need to extract along the off-diagonal of the upper triangular states array (a,b). This needs to be performed for linear, quadratic, and bilinear.

2.2.1 Adiabats and diabats

2.2.2 TD-DFT excited states model limitations

2.3 Calculation of Spectra

2.3.1 Time-correlation functions

2.3.2 MCTDH

2.3.3 VECC

VECC Paper: <https://arxiv.org/pdf/2312.14164.pdf> Our model comparisons have the following assumptions: although we can submit more information to MCTDH than VECC (e.g. off-diagonal coupling of state energies and excited state-to-state transition dipole moments), our comparisons are limited to whatever VECC accepts. Therefore, the difficulty arises from finding a hybrid format compatible to both dynamics-simulation-schemes that can capture the model. A technical challenge, mainly.

2.4 Relativistic effects

2.4.1 Spin Orbit Coupling (SOC)

2.5 Thermal effects

2.5.1 TF-VECC: Thermal Field Vibrational Electronic CC

2.6 Computational Details

2.6.1 GAMESS-US

GAMESS is an ab initio quantum chemistry package. The general pipeline of how calculations are run: HF/DFT/GVB/MCSCF \Rightarrow CI/MP2/CC (these are correlation corrections) \Rightarrow SOC/DK (these are relativistic effect corrections)

Having reservations about a job is needed, oftentimes a double or even triple check is prudent.

Table 2.1: Step 1 information.

Command Line	Input	Output
<code>python3 step0.py</code>		
Manual execution: <code>python3 step1.py</code>	<code>molecule_optfreq.com</code>	<code>molecule_optfreq.log</code> <code>master_values.json</code>
Automated execution: <code>python3 automate_scripts.py -1</code>		

2.6.2 nlogn, ComputeCanada (Computational Resources)

On nlogn, the available compute is as follows: `cpu[005-006] intel,westmere, X5670 12 48000`
`io003 intel,ivybridge,E5-2 16 128000 io[001-002] intel,westmere,X5670 12 48000`

On ComputeCanada Graham cluster, the available compute per node is 32 CPUs with 125GB of memory.

The GAMESS memory requirements for parallelized calculations is as follows: $((\text{mwords} * p + \text{memddi}) * 8) / 1024 = \text{TOTAL NUMBER OF GB}$. Where p is the number of processors, mwords is the amount of memory (in Words) for each processor, memddi is the distributed memory for optimization runs, and $8/1024$ to account for the conversion of words to gigabytes. Therefore, on nlogn, the max amount of compute per run would be 16 CPUs at 4 gigabytes. For reference, see <https://docs.alliancecan.ca/wiki/GAMESS-US>.

To compile Zeng’s GMC-QDPT compatible GAMESS version on ComputeCanada clusters, it is complicated. First, the *gmcpt.src* module must be obtained from Zeng. The next step is to *module load intel/2023.2.1* and run *./config* in GMS’ source directory; this will prompt a set of questions in which the answers are as follows: `linux64`, `enter`, `enter`, `enter`, `ifort`, `10`, `none`. After you msut enter the `ddi/` directory and run *./compddi* and move the

ddick executable one directory above. Finally, you must run `./lked` to link GAMESS to your path.

Chapter 3

Vibronic Model Development

3.1 CASSCF

To elucidate the active space without prior intuition is a challenging task. Many iterations on assessing the orbitals and adjusting the dimensions of the active space were needed. This human in the loop

3.2 Diabatization protocol Python code explanation

combined_toolkit.py is simply a collection of 8 utilities that are useful in preparing the

diabatization template: (1) OPTIMIZED RHF

(2) OPTIMIZED ROHF

(3) Semi-canonical MOs

(4) MCSCF Natural Orbitals

(5) OPTIMIZED MCSCF

(6) DMO group

(7) REFDET GROUP

(8) EQUILIBRIUM GEOMETRY

Listing 3.1: Toby Zeng’s handcrafted \$REFDET group in

RhF3_SPK_gmcpt_C1_e_mult5_diis_15st_diab_refG.inp

```
1 $REFDET
2 15
3 1 1
4 206 -1.000000
5 2 4
6 134 -0.500000
7 167 0.500000
8 186 -0.500000
9 81 0.500000
```

10	3 6		
11		59	−0.408248
12		26	−0.408248
13		98	−0.408248
14		45	0.408248
15		112	0.408248
16		146	−0.408248
17	4 4		
18		6	−0.500000
19		11	0.500000
20		31	−0.500000
21		86	0.500000
22	5 1		
23		1	−1.000000
24	6 1		
25		207	1.000000
26	7 4		
27		82	0.500000
28		135	−0.500000
29		168	0.500000
30		196	−0.500000
31	8 6		
32		99	0.408248
33		46	−0.408248
34		156	0.408248
35		27	0.408248
36		122	−0.408248

37		69	0.408248
38	9 4		
39		7	−0.500000
40		16	0.500000
41		91	0.500000
42		36	−0.500000
43	10 1		
44		2	1.000000
45	11 1		
46		208	1.000000
47	12 4		
48		83	−0.500000
49		188	−0.500000
50		136	0.500000
51		197	0.500000
52	13 6		
53		114	0.408248
54		123	−0.408248
55		61	−0.408248
56		176	0.408248
57		28	0.408248
58		70	0.408248
59	14 4		
60		13	−0.500000
61		106	0.500000
62		51	−0.500000
63		17	0.500000

64	15	1	
65		8	1.000000
66	\$END		

algorithm algpseudocode

Algorithm 1: An example algorithm

```

1 [1]
2 Require  $n$ 
3  $\text{geq } 0$ 
4 Ensure  $y = x^n$ 
5 State  $y$ 
6 gets 1
7 State  $X$ 
8 gets  $x$ 
9 State  $N$ 
10 gets  $n$ 
11 While  $N$ 
12  $\text{neq } 0$ 
13 If  $N$  is even
14 State  $X$ 
15 gets  $X$ 
16 times  $X$ 
17 State  $N$ 
18 gets
19  $\text{frac}N2$ 
20 Comment This is a comment
21 ElseIf  $N$  is odd
22 State  $y$ 
23 gets  $y$ 
24 times  $X$ 
25 State  $N$ 
26 gets  $N - 1$ 
27 EndIf
28 EndWhile

```

Toby Zeng created a custom DMO using symmetrized reference geometry orbitals. 39 (a1'), 45,46 (e'') 47,48 (e') 51 (a1'*) 53,54 (e'*) became the SOC's 44, 45,46, 47,48, 49, 50,51.

3.2.1 diabatize() and mctdh() functions

3.2.2 Distorting coordinates and calculating up to quadratic coupling constants

3.2.3 Fitting on n-th order models

Bells and whistles

3.2.4 Generating spectra via MCTDH and VECC

To crack this problem, rewritten program with Neil. The program is not perfect, but we can still use it to do relatively functional calculations and build models.

Chapter 4

Full Computational Study on Large System

4.1 Results

4.1.1 Vibronic spectra of test case pnictogen hydrides

4.1.2 Spectra of D_{3h} SOC transition metal complexes (CoF_3 and RhF_3)

4.2 Iron pentacarbonyl, $\text{Fe}(\text{CO})_5$

Using $\text{Fe}(\text{CO})_5$ as an example: we first recognized that the full valence active space was 11o18e. However, doing a oneshot for even the ground state equilibrium geometry calculation did not result in anywhere near convergence. Instead, we had to slowly build up the transition metal compound's ground state geometry by filling in bunches of electrons and orbitals at a time. The initial calculation was a 4+ cation, with a multiplicity of 1. Next, a calculation of 1+ charge and multiplicity of 4, due to three unpaired electrons between a set of a'' and e' of orbitals 46 through 48. This calculation resolved the orbital

ranking: a'' was lower than e' . Finally, a third calculation was ultimately performed on the fully electron-filled molecule and this neutral model yielded the full equilibrium geometry.

CHEMICAL INTUITION NEEDED!

We first started by doing an optimization of $\text{Fe}(\text{CO})_5^{4+}$. Then, we introduced 2 electrons (multiplicity of 3), and then made it neutral. Observed that the irreducible representation

orbital ranking of HOMO was e' and LUMO was a'_1 . After obtaining the equilibrium geometry, we proceeded to fill the active space. 4o4e, 5o8e, 6o8e, 6o10e, 8o10e, 9o12e, and the finally the full valent 11o18e. However, 11o18e did not seem to converge no matter what we tried. So we opted for 8o10e to tide us over and achieve a reasonable spectrum.

Curse of dimensionality when doing 33 modes ... gigantic ... ML-MCTDH is needed according to Prateek: <https://mattermodeling.stackexchange.com/questions/535/what-is-the-largest-system-for-which-vibrationally-resolved-electronic-spectra-h>

So we screened modes using effective vibronic coupling magnitude above a threshold. Linear couplings only. See JCTC singlet and triplet fission paper. Heart-wrenching that MCTDH could only compile up to 220k parameters when we needed 260k for our full model. Emailed Prateek and H.D.-Meyer. We can assert that only these 12 modes belong to the model. Incredible struggle, but nah, we'd win.

We are using ORMAS+GMCPT! <https://ccl.scc.kyushu-u.ac.jp/nakano/gmcpt.html>
Normally we use Douglas-Kroll (DK) in the *\$relfun* group, however Sapporo basis sets like RELWFN=LUT-IOTC instead.

D_{3h} Character Table

Table 4.1: Character Table for the D_{3h} Point Group

D_{3h}	E	$2C_3$	$3C'_2$	σ_h	$2S_3$	$3\sigma_v$
A'_1	1	1	1	1	1	1
A'_2	1	1	-1	1	1	-1
A''_1	1	1	1	-1	-1	-1
A''_2	1	1	-1	-1	-1	1
E'	2	-1	0	2	-1	0
E''	2	-1	0	-2	1	0

References / Bibliography

Appendices

Glossary

Index

SOC: <https://pubs.aip.org/aip/jcp/article/146/14/144103/195065> Metal trifluorides:
<https://www.sciencedirect.com/science/article/pii/S002222679700047> Diabatic states: G. J. Atchity and K. Ruedenberg, Theor. Chem. Acc.
97, 47 (1997).<https://doi.org/10.1007/BF00771111>

<https://pubs.aip.org/aip/jcp/article/115/22/10353/946057>/The-direct-calculation-of-
diabatic-states-based-on UV spectra:

<https://onlinelibrary.wiley.com/doi/10.1002/qua.10724> Linear vibronic models:

<https://pubs.acs.org/doi/10.1021/acs.jctc.1c00022> (Santoro)

We are using ORMAS+GMCPT! Student guide that really crystallized my knowledge of

GMC-QDPT: <https://ccl.scc.kyushu-u.ac.jp/~nakano/gmcpt.html>

file:///Users/bjc/Downloads/configurational_uniformity_hisao.pdf –
configurational_uniformity_paper

J Comput Chem 23: 1166–1175, 2002 : GMC-QDPT paper

4.2.1 <https://pubs.acs.org/doi/10.1021/jacs.2c01469>

- this is the iron pentacarbonyl modern paper

4.2.2 <https://pubs.acs.org/doi/10.1021/jp992474u>

4.2.3 <https://pubs.aip.org/aip/jcp/article/75/6/2560/791508>/The- Jahn-Teller-effect-in-the-photoelectron

Iron pentacarbonyl, haruyuki nakano and mark s gordon

<https://pubs.rsc.org/en/content/articlepdf/1999/cp/a808518h>

This is a very good diagram showing interdependencies:

https://en.wikipedia.org/wiki/Complete_active_space_perturbation_theory

This is a good consideration for future work, running the calculations via GPU acceleration:

<https://www.nvidia.com/es-la/data-center/gpu-accelerated-applications/gamess/>

MCTDH troubleshooting

<https://www.pci.uni-heidelberg.de/tc/usr/mctdh/doc.86/mctdh/trouble.html>[redim](#)

GAMESS input documentation:

[https://www.msg.chem.iastate.edu/gamess/GAMESS_{Manual}/docs – input.txt](https://www.msg.chem.iastate.edu/gamess/GAMESS_manual/docs-input.txt)