

Chapter 1

Introduction

1.1 Vibronic Models Background

Since the breakthrough of quantum physics, it was claimed that the whole of chemistry was in essence solved. The Schrödinger equation had it all, as long as you were able to solve it. Despite the rapid growth in computing, for the majority of meaningful and sizeable chemical systems, the calculation of its exact solutions remains intractable. Therefore, modern quantum chemistry relies on clever mathematical approximations to calculate energies sufficient for predictive and chemical accuracy. In 1927, Born and Oppenheimer proposed the Born-Oppenheimer Approximation (BOA), a method that separates the nuclear and electronic regimes and allows one to calculate the energy of adiabatic potential energy surfaces (PES). The transformed electronic Hamiltonian was more manageable. However, many realistic chemical processes are innately non-adiabatic, and have electronic states that are not well separated and are nearly degenerate. Prime examples where non-adiabatic effects show an important role are the photo-isomerization of retinal and transition metal complex spectra. Vibrational-electronic (**vib-ronic**) coupling models aim to simulate the mechanics revolving around non-adiabatic dynamics. Maturation in both post-Hartree-Fock and computing methods have since enabled quantum and computational chemists alike to explore sophisticated non-adiabatic phenomena such as conical intersections and intersystem crossings.

Research in non-adiabatic chemistry has since gained momentum in recent decades leading to the consequent construction of vibronic models that can go beyond the BOA and can capture a proper non-adiabatic picture. Vibronic models in a diabatic basis is theorized to be one viable approach to non-adiabatic treatments. This procedure is referred to as a diabatization. Diabatic states (diabats) have the advantage of diminishing the tricky non-adiabatic couplings (NACs) that arise from the BOA adiabatic states (adiabats), and maintaining smooth character along the PES. To solve for dynamics, many proceed to use the classic Multiconfiguration Time-dependent Hartree (MCTDH) algorithm. A long term goal of the Nooijen group is to formulate novel alternative operator techniques like Vibrational Electronic Coupled Cluster (VECC) to calculate vibrationally-resolved electronic spectra and statistical mechanics based on a vibronic model framework [1].

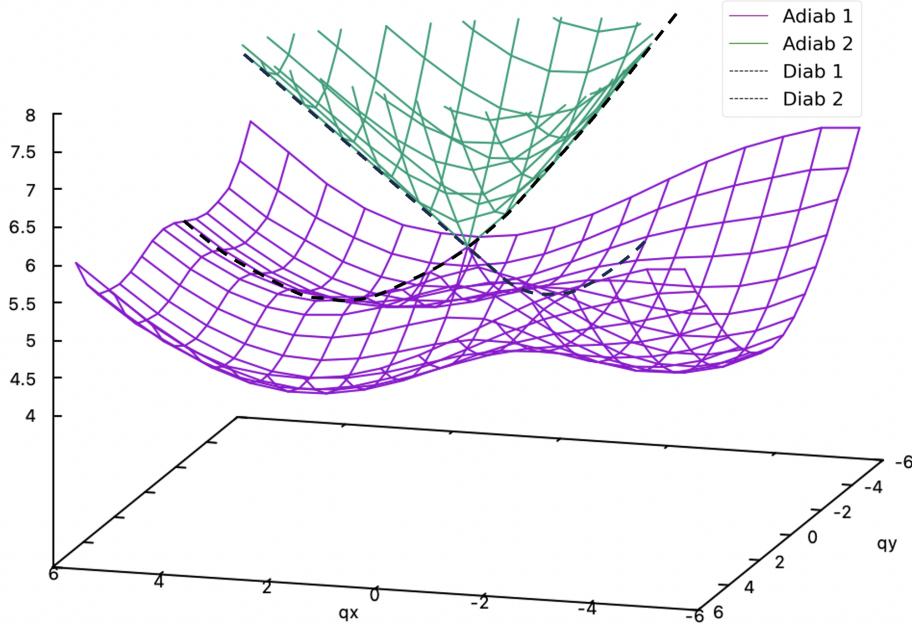


Fig 1. Stereotypical depiction of a conical intersection (CoIn) involving both adiabatic states (adiabats) and diabatic states (diabats) commonly seen in vibronic models. Source: Zeng.

The central goal in this project is to produce an accessible diabatization protocol that can supply MCTDH and VECC with vibronic model Hamiltonians suitable for a wide range of applications such as spin-orbit coupling (SOC) effects and spectroscopy. For the past two years at the Nooijen and Zeng group, we have jointly endeavoured to pursue this goal.

1.2 Modern Vibronic Models - Timeline

The pioneering work of Cederbaum, Köppel, and Domcke gave rise to the original idea of vibronic models in their 1981 seminal paper [2]. Determination of a vibronic model Hamiltonian that simultaneously coupled vibrational modes with an arbitrary set of electronic states was given in an *ab initio* procedure. Their study proceeded to simulate spectroscopy and demonstrated the need for the inclusion of vibronic coupling effects to match with findings in experimental spectra. Neugebauer and Nooijen found this approach useful and further developed a diabatization scheme suitable for Time-dependent Density Functional Theory (TD-DFT) absorption spectrum [3, 4]. Vibronic models were used at this point to model multiple potential energy surfaces in a limited region of configuration space, akin to a harmonic oscillator approximation applied to a single surface. Moreover, this established a formalized and standard method for extracting (from quantum chemistry calculations) vibronic model Hamiltonians in a routine fashion. Within the Nooijen group, Chang, Endicott, and Golac have iterated on vibronic methods. This scheme inspired Santoro and coworkers, who in recent years recasted Nooijen's scheme into a single-reference linear order Gaussian-MCTDH TD-DFT diabatization scheme purposed for studying the excited state dynamics of hexahelicene [5]. However, hexahelicene is merely an organic molecule. Ongoing frontier research by Domcke and Mondal aims to apply vibronic models for transition metal complex spectra, a notoriously difficult task due to the dilemma of low-lying degenerate electronic states. It is difficult to resolve the states and PES character for such systems. Therefore, a more ideal approach is to pivot towards a diabatization scheme that is multi-reference in nature, and can capture relativistic effects like spin-orbit coupling.

Such existence of a diabatization protocol that includes spin-orbit coupling was first devised by Toby Zeng, an expert who specializes in vibronics and has massively contributed to the field of arbitrarily large Hamiltonian formalisms and Jahn-Teller (JT) interactions [6]. The foundation of the Zeng SOC protocol derives from previous work done by Atchity

and Rudenberg on configurational uniformity, and specifically uses GAMESS-US package Generalized Multi-Configuration Quasidegenerate Perturbation Theory (GMC-QDPT) calculations. Diabatic states are directly calculated using Nakamura and Truhlar's version of configurational uniformity in GAMESS, whereby diabatic molecular orbitals (DMOs) are first prepared, and then computation of diabatic state functions follows. Calculations involved here are typed as multi-state multi-reference perturbation theory calculations in a given active space reference as generalized and implemented by Nakano. GMC-QDPT addresses the requisite Multi-configuration Self Consistent Field (multi-configurational self-consistent field (MCSCF)) component of the model by performing Occupation Restricted Multiple Active Space (ORMAS) SCF. At this point, Zeng assembles all these pieces and opens up the possibility of adding spin orbit coupling terms to the model Hamiltonian. The basis of the vibronic models in this project will use the diabatic energies and couplings yielded by this process.

Familiarity of vibronic models with SOC and spectroscopy is not recognized as new. Siebrand and Zgierski modelled vibronic SOC interactions in their 1979 pyrazine study [7]. On transition metals, Glaesemann, Gordon (of GAMESS), and Nakano scrutinized the behaviour of $FeCO^+$ in a wavefunction study. First author Mondal has in particular discussed E x e Jahn-Teller and SOC effects in transition-metal fluorides, such as CoF_3 with a 17 orbital and 30 electron active space - CAS(30,17). Their analysis included generation of D_{3h} trigonal planar symmetry vibronic model Hamiltonians and spectra. Previous applications of the Zeng SOC protocol were towards pnictogen hydride cations with trigonal symmetry: PH_3^+ , AsH_3^+ , and SbH_3^+ for the purposes of investigating vibronic coupling and SOC characteristics. No spectroscopy components were included. So far, this context and state of research converges on a case for which the project's diabatization protocol should be applied to: a large D_{3h} iron-carbonyl complex with low-lying degenerate electronic states. The candidate molecule is $Fe(CO)_5$. If successful, the appealing challenge of $Fe(CO)_5$ should exhibit demonstrable JT, SOC, and photochemical effects as observed in the literature.

1.3 Project Scope: Pivot to Zeng Protocol and Limitations of Single-reference Linear Vibronic Models

To capitalize on all the work presented so far, I have created a novel automated diabatization code in Python to be incorporated into the Zeng SOC protocol. This is my main contribution. In previous work, the Santoro protocol was used to generate vibronic models in Gaussian 16 for MCTDH testing. Naturally, they did not suffice for transition metal complexes. As shown in Fig 2, the vibronic models were found to not resolve any fine structure in metallic vibronic spectra, and for cases such as caffeine, the spectra was still inaccurate despite meticulous checking of the model. Santoro has likewise acknowledged the inaccuracies with using the linear protocol due to complications in finding correct parameters for calculations. The most pressing concern of using the Santoro protocol at the Nooijen lab was that the output vibronic model Hamiltonian (operator file) format was incompatible for VECC input.

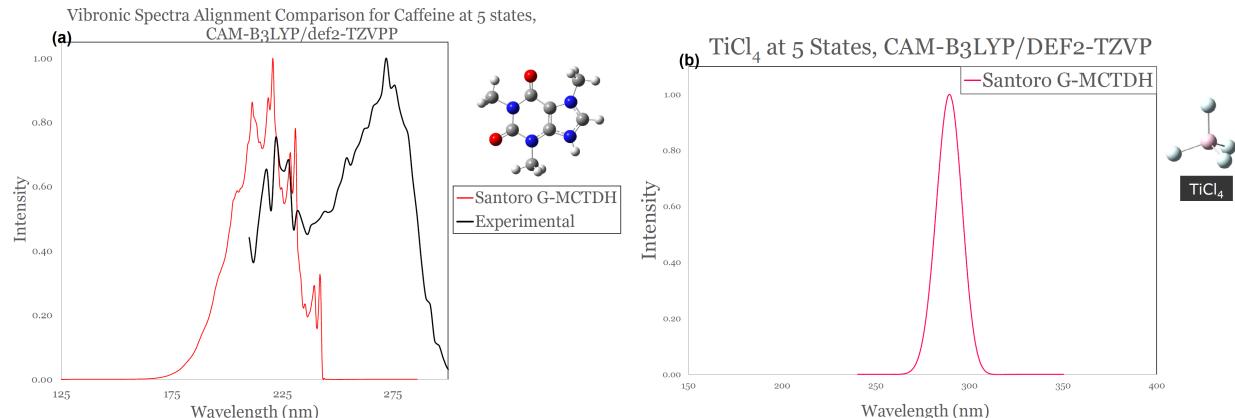


Fig 2. (a) Caffeine. Slightly matching peaks in the sub 240nm region observed, appears to need a linear red-shift for better agreement. (b) Titanium tetrachloride. The predictive accuracy here of a smooth single peak is recognized to be nowhere near reality, due to limitations of single reference TD-DFT methods applied to transition metal complexes.

To improve on the capabilities of the vibronic models available for VECC, such limitations of the Santoro G-MCTDH method prompted for a pivoting towards the Zeng protocol. Prior to this project, only the top half path shown in Fig 3. (Santoro-MCTDH-Spectra) was explored. Here, VECC was also able to perform excited state simulations like MCTDH,

however VECC's statistical mechanics ideally necessitate Zeng models for proper functionality. Hence to reiterate the declared task: we shall construct a composite approach where the Zeng protocol is first used to manufacture vibronic model Hamiltonians to be plugged in as input, and the latter half of non-adiabatic simulation shall be taken care of by VECC. A Zeng-VECC-Spectra pathway. Once the Zeng protocol can be successfully connected to both MCTDH and VECC, then it will allow for a variety of models and dataset to be used for practical comparison of the gold standard MCTDH method with the new VECC method.

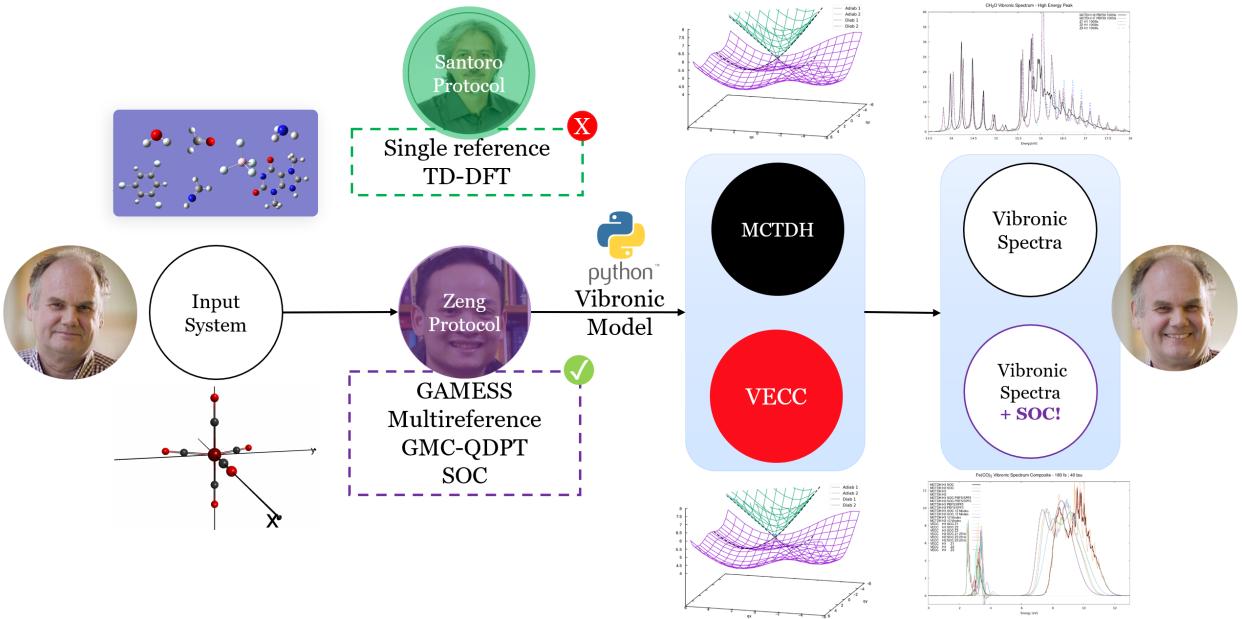


Fig 3. Original proposed research plan, the right half motivates the left half.

Formerly, the Zeng diabatization protocol was not automated. Each calculation in the series needed for full diabatization was handcrafted. In early 2023, Zeng completed the first step towards accomplishing the task: a prototype 'machine gun' Bash code that automates diabatization. This version could effectively produce up to quadratic order Zeng non-SOC models, and was implemented to be extracted from GAMESS calculations. Neil Raymond and I have iterated on the prototype and designed a seamless operating protocol. The code was first translated into Python, and then received a wide range of upgrades to ensure that it is SOC-compatible and readily plugged into MCTDH/VECC for spectroscopy.

1.4 Outline of Thesis

In this thesis:

1. The first chapter gives an overview of vibronics and the research project objectives.
2. Fundamental quantum chemistry concepts and mathematics is presented.
3. Full elucidation of Python code to automate the Zeng SOC protocol.
4. Applications of the protocol to realistic transition metal systems and vibronic spectroscopy.
5. Conclusion and future outlook.

Chapter 2

Theoretical Overview and Methodology

2.1 Electronic structure theory

In an ideal world, the Schrödinger equation would be readily solved by computers and the exact solution for energy would be known.

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 [8]. Therefore, if \vec{R} changes, so does $\phi_{\lambda}(\vec{r}; \vec{R})$.

Proceed with full expansion of Hamiltonian acting on the wavefunction:

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

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

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

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) [9]. It is typically proximate enough to zero for it to be neglected. Likewise, **C** is zero for real normalized wavefunctions. **C** is referred to as the off-diagonal non-adiabatic coupling (NAC) terms. In other words, only **B** and **D** terms remain because terms with the presence of $\int dr \vec{\nabla}_{\alpha} \phi_{\lambda}(\vec{r}; \vec{R})$ 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 [9].

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 Equation (2.10) can be solved for at a single electronic state λ with any given fixed-geometry \vec{R} . In practice of using BOA concepts, Figure 2.1 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 [10].

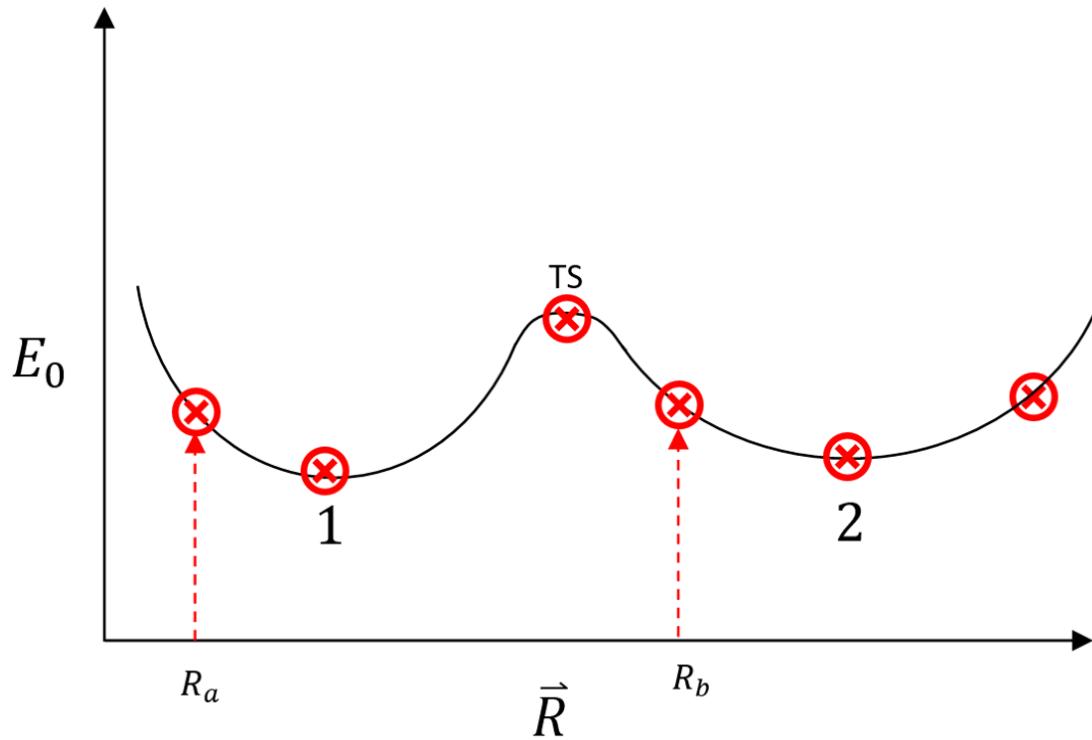


Figure 2.1: A single electronic ground state surface within the BOA leads to sampling and subsequently calculating points at fixed nuclear configurations $(\vec{R}_a, \dots, \vec{R}_b, \dots)$.

2.1.1 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—??(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 [11].

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

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

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

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

$$\begin{bmatrix} \hat{T}_N + E_1(\vec{R}) + A_1(\vec{R}) & N.A.C. \\ N.A.C. & \hat{T}_N + E_2(\vec{R}) + A_2(\vec{R}) \end{bmatrix} \Rightarrow \begin{bmatrix} \hat{T}_N + E_{11}(\vec{R}) & E_{12}(\vec{R}) \\ E_{21}(\vec{R}) & \hat{T}_N + E_{22}(\vec{R}) \end{bmatrix} \quad (2.13)$$

Obtaining this matrix of diabatic form, it can be utilized to describe potentials and a coupled

set of vibrational wavefunctions $\chi_a(\vec{R})$:

$$\begin{pmatrix} \hat{T}_N + E_{11}(\vec{R}) & E_{12}(\vec{R}) \\ E_{21}(\vec{R}) & \hat{T}_N + E_{22}(\vec{R}) \end{pmatrix} \begin{pmatrix} \chi_1(\vec{R}) \\ \chi_2(\vec{R}) \end{pmatrix} = E \begin{pmatrix} \chi_1(\vec{R}) \\ \chi_2(\vec{R}) \end{pmatrix} \quad (2.14)$$

$$\Psi_a(\vec{r}; \vec{R}) = \sum_{\lambda} U_{\lambda a}(\vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) \quad (2.15)$$

$$S_{\mu\lambda}(\vec{R}) = \int \phi_{\mu}^*(\vec{r}; \vec{R}_{q_0}) \phi_{\lambda}(\vec{r}; \vec{R}_{q_0 \pm dq_i}) \quad (2.16)$$

$$\sum_{\lambda} \int \phi_{\mu}^*(\vec{r}; \vec{R}) \hat{H}_{elec} \Psi_a(\vec{r}; \vec{R}) \phi_{\lambda}(\vec{r}; \vec{R}) dr \chi_{\lambda}(\vec{R}) = \sum_{\lambda} E_{\mu\lambda} \chi_{\lambda}(\vec{R}) \quad (2.17)$$

The diabatic potential energy matrix is $E_{\mu\lambda}(\vec{q})$. Vibronic coupling terms can be obtained as Taylor series expansions of $E_{\mu\lambda}$, commonly up to quadratic order

$$E_{\mu\lambda}(\vec{q}) = E_{\mu\lambda} \delta_{\mu\lambda} + \sum_i^M E_{\mu\lambda}^i q_i + \frac{1}{2} \sum_{i>j}^M E_{\mu\lambda}^{ij} q_i q_j + \dots \quad (2.18)$$

Vibronic coupling terms can be obtained as Taylor series expansions of The coefficients of the coupling terms can be calculated by numerical differentiation:

$$E_{\mu\lambda}^i = \frac{E_{\mu\lambda}(\vec{R}_{q_0} + dq_i) - E_{\mu\lambda}(\vec{R}_{q_0} - dq_i)}{2dq_i} \quad (2.19)$$

Vibronic models - short summary?

Possibly have some equations here if you need to refer to them later? Optional

2.2 Building Wavefunctions

2.2.1 Multi-configuration Self Consistent Field (MCSCF)

To improve on the traditional SCF method, when a linear combination of Slater determinants (instead of single) are used to construct the reference wave function as seen in MCSCF, this enables static correlation to be expressed. MCSCF wavefunctions are useful in cases where electrons show high correlation, such as excited states.

CASSCF & ORMASSCF

Complete Active Space Self Consistent Field (CASSCF) is one special type of MCSCF. The characteristic feature of CASSCF is that one must proceed to take the entire chosen active space and consider all electronic configurations within it. If there is limitation in the active space, this is Restricted Active Space (RAS). ORMASSCF on the other hand, is a subset of CASSCF, but provides comparable accuracy at a cheaper cost. In order to use ORMASSCF reference wavefunctions in GAMESS, the REFTYP=ORMAS flag must be set within the \$GMCPT group.

2.2.2 Active Space Design

Designing the active space is a pedantic problem. 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 are expected to be performed in order to nail down the active space. This human in the loop part. What does ORMAS address? Cool paper: <https://arxiv.org/pdf/2111.15587#:~:text=ORMAS%20is%20a%20construction%20of, and%20minimum%20number%20of%20electrons.>

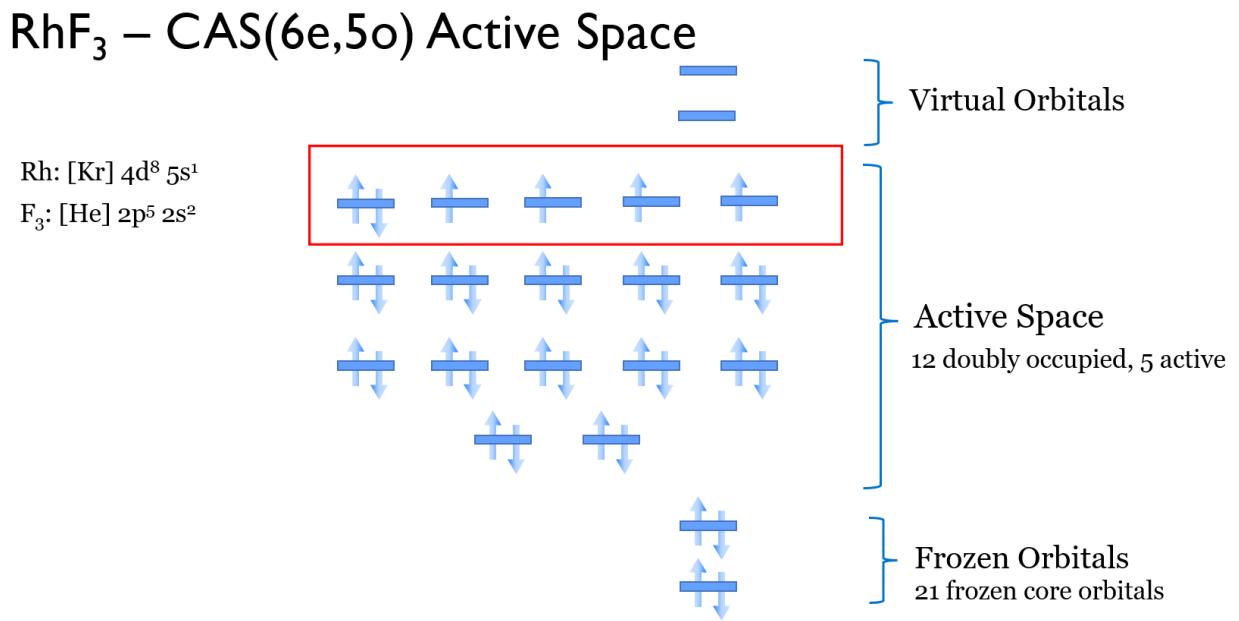


Figure 2.2: Quantized energy level diagram showing orbitals and electrons in a $\text{CAS}(6,5)$ active space of RhF_3 .

GMC-QDPT

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 static and dynamical correlation alike. 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. This is needed because multireference calculations will save the dynamic correlation aspect. Perturbation theory (PT) can "correct the many MCSCF states simultaneously". 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>. As per the GAMESS input documentation manual, to invoke GMC-QDPT, one needs to use \$GMCPT group in GAMESS (6 letters due to Fortran limit). To achieve the second order quasidegenerate PT correction, one needs to use the MPLEVL=2 and MRPT=GMCPT flags.

An important consideration is that FCI is often impossible for large systems due to the computational cost growing factorially in terms of reference space determinants, so this efficient scheme is more feasible instead. As Rahm remarked, the full level of theory considered here is therefore "a ridiculously high GMC-QDPT ORMAS SOC" ¹.

CASSCF/MRMP level

Diabatic Molecular Orbitals

Configurational uniformity's implementation in GAMESS is a concept devised by Nakamura and Truhlar to manufacture diabats from adiabats. In their framework, which expands on previous work by Atchity and Rudenberg,a diabatization-adapted molecular orbitals (DMOs) representation is obtained, and then subsequent DSFs (diabatic state functiosn) are constructed. The idea is to optimize for uniformity in electronic configurations that originate from DMOs. Model space

¹Prof. Martin Rahm presentation at York University

2.3 Jahn-Teller and Relativistic Effects

2.3.1 Group Theory and Symmetry

2.3.2 E x e JT problems

2.3.3 Spin Orbit Coupling (SOC)

SOC scales with Z, heavier the atom, more effect. Since it is a relativistic effect and a small correction to the H, we call this 'perturbation'. Goeppert Mayer's analogy of twirling dancers, clockwise/counter-clockwise

2.4 Calculation of Spectra

2.4.1 Traditional Franck-Condon

2.4.2 Time-correlation functions

2.4.3 MCTDH

Consultation with Dieter Meyer of MCTDH

2.4.4 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.

Focusing on VECC, it is potentially advantageous due to its generation of results in a theoretical $O(n^2)$ time scaling, where

n is the number of normal modes. In comparison, the modern multi-layer MCTDH approach is computationally uneconomical with its exponential scaling and requires extensive user input to dictate efficient calculation. Despite VECC being less systematic, it has the critical advantage of demanding very few user choices; limited to simple on/off switches. VECC's present implementation requires construction of vibronic model Hamiltonians to be plugged in as input, which is likewise an expensive and tedious task.

2.4.5 Thermal Effects: Thermal Field Vibrational Electronic CC (TF-VECC)

2.5 Computational Hardware and Software

2.5.1 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). One interesting thing about GAMESS is that recently there is possibility for GPU-acceleration.

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

2.5.2 nlogin, ComputeCanada (Computational Resources)

On nlogin, 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: $((\text{mwords} * \text{p} + \text{memddi}) * 8) / 1024 = \text{TOTAL NUMBER OF GB}$. 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 nlogin, 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 must enter the ddi/ directory and run *./compddi* and move the ddikick executable one directory above. Finally, you must run *./lked* to link GAMESS to your path.

Chapter 3

Diabatization Code Development

The newly developed software package will be described, along with how it is used to produce vibronic models and compute spectra. The majority of the development process was done in collaboration with Neil Raymond, and the software is publicly available on [GitHub](#) [12]. As previously outlined in Section 1.3, this is my main contribution, and the centerpiece of the project.

The Zeng protocol can be considered to have two broad stages: first, we employ a multi-configurational self-consistent field (MCSCF)-style method to obtain a ‘template’ (primarily the diabatic states¹) and then we proceed to express the Hamiltonian in those diabatic states, (i.e. obtain a Hamiltonian in the diabatic representation). I briefly outline the individual steps of the MCSCF calculations, and **DiabToolkit** which automates this process, in Section 3.1. Next, Section 3.2 explains how the **dist** code performs the diabatization, by walking through an example step-by-step (NH_3). In addition, I describe in detail the practical challenges encountered, as well as their solutions, in automating this extensive and unwieldy process. Workflows and software related to spectroscopy are discussed in Section 3.3. Finally, some implementation details and other concerns are briefly touched upon in Section 3.4.

¹as well as a reference geometry model, frequencies and other assorted parameters.

3.1 MCSCF Pipeline

The MCSCF process can be broken down into four individual steps:

1. Appendix 5.1.1: Geometry optimization + frequency (Hessian) calculation
2. Appendix 5.1.2: GMC-QDPT calculation
3. Appendix 5.1.3: DMO calculation
4. Appendix 5.1.4: REFDET calculation

with some steps requiring very specialized knowledge of GMC-QDPT and ORMASSCF intricacies². The approach taken here is a slightly modified version as seen in Zeng's work [cite](#). A visual representation of the entire Zeng protocol (diabatization process) in presented in Figure 3.1, with the four MCSCF steps highlighted in purple.

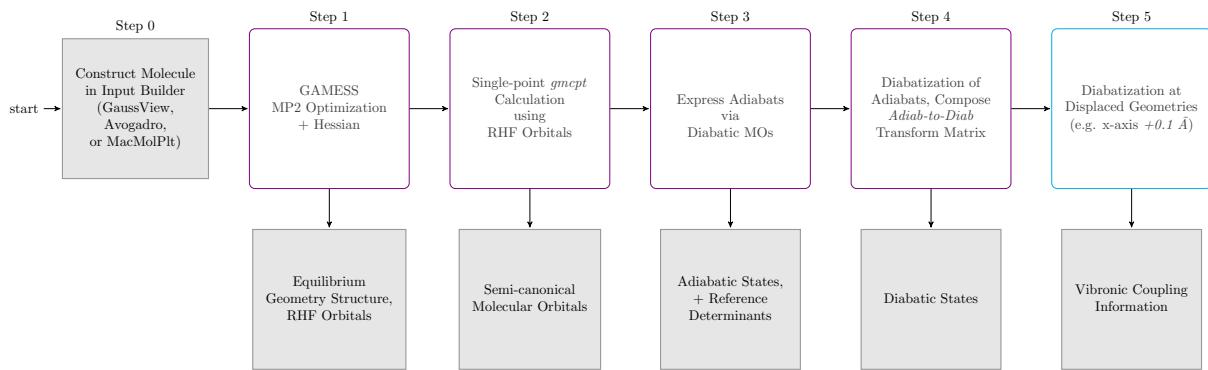


Figure 3.1: GMC-QDPT ORMASSCF vibronic coupling model scheme formulated by Zeng.

We begin by using standard computational chemistry software (e.g. Gaussian, ORCA, GAMESS) to obtain the equilibrium co-ordinates and the frequencies based on the input geometry. In the second step we use a GMC-QDPT calculation to obtain the adiabatic energies and the semi-canonical molecular orbitals (of which a subset are diabatic orbitals). The diabatic molecular orbitals are used to calculate reference determinants in step three. Finally, we can calculate the diabatic states using all the parameters obtained in the previous calculations (e.g. ref geom, SC MO's, ref det's). After these four steps, all of the necessary

²Consultation of Nakano's student guide [13] is highly advised.

components for a diabatic model have been calculated, and are then used in the diabatization procedure as described below in Section 3.2.

Initially the process for performing these calculations was an assortment of (X) (e.g. manual copying, shell scripts, excel editing) which was excessively slow, error-prone, and obscure to non-experts. Because the method is approximate and quite complex, it can be quite time consuming to distinguish between operator error and systematic properties. As a consequence the overhead of on-boarding was exceptionally high and roadblocks encountered during execution of the method required diagnosing by a specialist.

The most problematic elements of this process were the composition of inputs for steps three and four. These typically required extracting out information from the preceding calculation’s ‘punched-out’ orbitals or energies. For simple models such as H₂O (three electronic states and three vibrational modes) manually performing this process is burdensome, but not prohibitive. Commonly there are a dozen or so output files from GAMESS, roughly 6000 lines (or X kB), and it takes on the order of 15 minutes to manually compose the inputs. For more sizeable systems (but still chemically small molecules / but still only 11 atoms?) such as Fe(CO)₅, manual processing is absurd. There are X output files, each of which are millions of lines long (70 MB), and in the past the entire MCSCF process would take between one to two weeks to execute. Compared to the current approach which can obtain a diabatic model for Fe(CO)₅ in under 24 hours.

To address these obstacles a package of software scripts **DiabToolkit**([ref to GitHub page](#)) was developed. The package provides eight programs which assist with executing the protocol such as: extracting information, performing *simple* mathematical calculations, preparing input files for further steps, providing helpful suggestions and warnings to the operator, and automating the validation of data within minimum specifications (values are in correct ranges, no anomalous coefficients, reality checks, etc.). These scripts in **DiabToolkit** have saved an incredible amount of time and frustration. They have shortened the on-boarding process drastically increasing the likelihood of adoption of this method by new

researchers. By reducing the time spent on **mechanical duties** more time is available to invest into innovate research. Most importantly, by automating the process human error is almost entirely eliminated. A detailed walkthrough of each step in the MCSCF process, using **DiabToolkit** for a RhF₃ SOC system, is provided in the appendix 5.1. Once Step 4 is completed, vibronic coupling information at reference geometry is directly available, and the Step 4's input file can serve as a template for the diabatization to proceed.

3.2 Automated Machine Gun Diabatization Code:

As shown in Figure 3.1 the Zeng protocol can be broken down into five steps. The previous four steps (highlighted in purple) were covered, and it is prudent to note that they merely touch upon the MCSCF component of the complete diabatization process. Here, we will expand on the final step (outlined in blue) and how, in automating this complicated process, the **dist** code was developed. Given a template, as described in the previous section, the current **dist** code generates a vibronic model Hamiltonian (with up to quadratic order coupling constants) in a suitable manner for both VECC and MCTDH ³.

The diabatization of the template can be broken down into six steps (Figure 3.2):

- A. Selection of key parameters
- B. Section 3.2.1 Extraction of frequencies and reference structure (refG) coordinates, storing them inside dictionaries.
- C. Section 3.2.2 Submit refG calculation.
- D. Section 3.2.3 Precomputation of distortions and subsequent submission of calculations in 'machine gun' fashion.
- E. Section 3.2.4 Collection of diabatic vibronic model data.
- F. Section 3.2.5 Fitting PES plots for higher order modes (if necessary).

In the first step (A) the user must carefully select certain pivotal parameters such as: which $3N - 6$ modes (if any) to ignore, the magnitude of the displacement (**qsize**), and the grid upon which the displacements are evaluated. For example, a 'screened' version of Fe(CO)₅ might only use 12 out of the 27 vibrational modes present in the template, a

³Described in more detail in section 2.4.

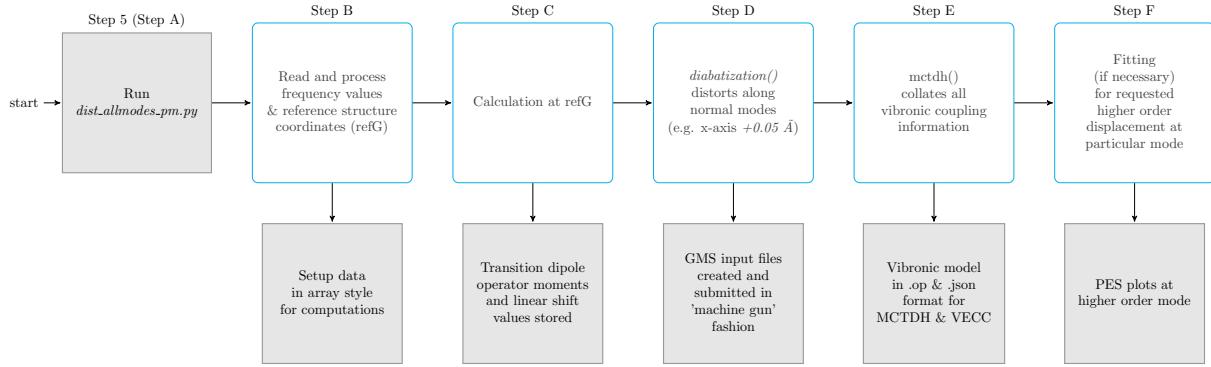


Figure 3.2: The six step process automated by the `dist` code.

`qszie` of 0.3, and for a particularly important mode (23) use a `x8` grid (meaning it mode 23 is evaluated at 8 points: `qszie`, `2*qszie`, `3*qszie`, etc.). Triple checking of parameters is recommended before executing the code. The code then automatically executes and manages all of the remaining five steps (B–F).

3.2.1 Step B: Frequencies and Reference Structure

In this step, the frequencies (q_i no?) and reference structure's coordinates (\vec{R}_{q_0}) provided by the geometry optimization and Hessian calculation are extracted into dictionaries. These are necessary for computing the overlap matrix ($S_{\mu\lambda}$) and eventually new vibronic matrix elements

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

as shown in Section 2.1.1.

Sometimes when the input symmetry of the molecule intrinsically produces a saddle point, the first frequency becomes imaginary and has an ‘I’ attached to its value. The code accommodates for this special case.

```

109      ***** EQUILIBRIUM GEOMETRY LOCATED *****
110  COORDINATES OF SYMMETRY UNIQUE ATOMS (ANGS)
111    ATOM   CHARGE      X          Y          Z
112  -----
113  N       7.0  0.0000000000 -0.1693806842  0.0000000000
114  H       1.0  -0.4653267700  0.2564602281  0.8059696078
115  H       1.0   0.9306535400  0.2564602281  0.0000000000
116  COORDINATES OF ALL ATOMS ARE (ANGS)
117    ATOM   CHARGE      X          Y          Z
118  -----
119  N       7.0  0.0000000000 -0.1693806842  0.0000000000
120  H       1.0  -0.4653267700  0.2564602281 -0.8059696078
121  H       1.0  -0.4653267700  0.2564602281  0.8059696078
122  H       1.0   0.9306535400  0.2564602281  0.0000000000

```

Listing 3.1: Optimized geometry of NH₃ coordinates of NH₃ at equilibrium geometry in units of angstroms.

3.2.2 Step C: Calculation at reference geometry

The third step is essentially a repeat of the REFDET calculation performed in the MCSCF process. However, this step is crucial for ensuring whether or not the template setup has any flaws. If the `refG` calculation does not terminate successfully, then `dist` will exit, as proceeding is useless. Diabatization **cannot** proceed until the user has isolated and corrected any mistakes in the template ⁴.

Transition dipole moment (operator) coefficients are taken from the output calculation as viable TDM values. Note that, theoretically, Rydberg orbitals and states are necessary to obtain ‘true’ TDM values. The linear shift in energy is also calculated in the manner shown in Listing 3.2.

```

1 # 1st diabat's energy
2     D1E = extract_diabatic_energy(ref_geom_path, a_pattern.format(col=1))
3     linear_shift = (D1E - GSE) * ha2ev

```

Listing 3.2: Linear shift adjustment calculation

In regards to VECC, its current usage strictly requires TDM values of 0.1 eV. Once again, the workflow and repository code accounts for this distinct condition.

⁴Explained in more detail in X.

3.2.3 Step D: Submit diabatization displacement calculations

When computing the

If we analyze 2.19, shown inside the equation are $+dq_i$ and $-dq_i$. Plus/minus distortions are precomputed on the geometry via linear, bilinear, and quadratic order displacements along the modes as seen in Listing 3.3. The new coordinates for the system are calculated, and then stored inside their respective input file. Specifically, for both the linear and quadratic case, the code will generate a +/- calculation for each mode from 1 to n , where n is the highest order requested at a mode. On the bilinear case, the code will proceed to generate 4 calculations at each mode-mode pair displacement: "++": 'pp', "+-": 'pm', "-+": 'mp', "--": 'mm'. Once this processing is complete, all eight calculations are submitted to GAMESS. However, if in the selected mode list there are higher order displacements requested at a certain mode (e.g. x8 instead of the standard x2), then additional +/-3 ... +/-8 calculations will be precomputed and submitted for that mode.

Cooking up a new recipe for distortions

```
1 # -----
2 # NEIL MAGIC CODE (^o^)
3 displacements = {
4     "+1": reference[:, NEW] + 1.0 * R_array[NEW, :] * mode_array[:, :],
5     "-1": reference[:, NEW] - 1.0 * R_array[NEW, :] * mode_array[:, :],
6     "+2": reference[:, NEW] + 2.0 * R_array[NEW, :] * mode_array[:, :],
7     "-2": reference[:, NEW] - 2.0 * R_array[NEW, :] * mode_array[:, :],
8 }
9 #           (1,   N)           (Z*3, N)
10 #          (9,   1)          (1,   3)          (9,   3)
11 #          (12,  1)          (1,   6)          (12,  6)
12
13 # -----
```

Listing 3.3: Precomputing linear displacements

Loops in python are notoriously slow. We initially emulated the original Bash prototype implementation of the distortion calculations using loops. So we refactored the excessive looping that distorted the modes into efficient vectorized 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. Without this directed learning, much of the pedagogical value in the exercise of reformulating a new diabatization protocol and 'learning the ropes' would not have occurred. Circumstantially, the code would also be unwieldy and resistant to accommodating changes.

3.2.4 Step E: Harvest Results for MCTDH/VECC

As per standard MCTDH Hamiltonian format, the operator file containing the so-called model Hamiltonian is constructed in various sections. The first section here is the operator-section, which initializes the title of the MCTDH run. Next, the frequencies, state energies (diagonal/off-diagonal pair), and electronic TDM values are defined.

```

1 operator-section
2 frequencies
3 EO_energies (diag/off-diag pair)
4 electronic TDM
5 parameter-section
6 linear coupling
7 bilinear coupling
8 quadratic coupling
9 SOC linear
10 SOC bilinear
11 SOC quadratic
12 hamiltonian-section
13 hamiltonain-section_Operator

```

Listing 3.4: MCTDH operator file format

Inside the parameter section are the linear, bilinear, and quadratic coupling term values. If the calculation is SOC enabled, then SOC version of the couplings will similarly be defined. Diabatically, these energies appear in the GAMESS output as 'GMC-PT-LEVEL DIABATIC ENERGY=', and critically, if they are diagonal in the states array ([a,a]) then it is extracted in Hartree units. If they are off-diagonal ([b,a]) diabatic couplings, then they will be extracted

in eV since at displaced geometries, the printed values are relative to the reference geometry. Moreover, the extracted off-diagonal values will fill the upper-triangular part of the matrix with rows and columns defined by states.

```

1     extracted_Hartree_1D = extract_in_Hartrees(path, a_pattern.format(col=0+1), cheat=True)
2         for a in range(A):
3             array[a, a] = extracted_Hartree_1D[a]
4
5     extracted_eV_2D = extract_in_eV(path, ba_pattern.format(row=0+1, col=1+1), cheat
6     =True)
7         for a, b in upper_triangle_loop_indices(A, 2):
8             array[a, b] = extracted_eV_2D[a, b]
```

Listing 3.5: Differentiation of linear vibronic coupling values extraction

Linear coupling terms

The linear coupling terms are calculated as shown in Equation (3.1).

$$E_{\mu\lambda, \text{Linear}}(q) = E_{\mu\lambda}(0) + \sum_i^M E_{\mu\lambda}^i q_i ; E_{\mu\lambda}^i = \frac{E_{\mu\lambda}(\vec{R}_{q_0} + dq_i) - E_{\mu\lambda}(\vec{R}_{q_0} - dq_i)}{2dq_i} \quad (3.1)$$

```

1     linear_ev = (temp_dict["+1"] - temp_dict["-1"]) / (2*qsize)
2     for a in range(A): # make sure we multiply the diagonal by ha2ev
3         linear_ev[a, a] *= ha2ev
```

Listing 3.6: Evaluating linear coupling term and conversion of units

Quadratic coupling terms

In the quadratic terms Listing 3.7, we copy over the entire +2/-2 arrays, and only multiply the diagonal by conversion factor $ha2ev$.

$$E_{\mu\lambda, \text{Quadratic}}(q) = E_{\mu\lambda}(0) + \frac{1}{2} \sum_{i>j}^M E_{\mu\lambda}^{ij} q_i q_j + \dots ; E_{\mu\lambda}^i = \frac{E_{\mu\lambda}(\vec{R}_{q_0} + dq_i) - E_{\mu\lambda}(\vec{R}_{q_0} - dq_i)}{2dq_i} \quad (3.2)$$

```

1  def _compute_using_array_style(i, temp_dict):
2      # until we change the extraction
3      temp_fake_E0_array = np.copy(E0_array_eV)
4      for a in range(A): # make sure the diagonal is in Hartrees
5          temp_fake_E0_array[a, a] = E0_array_au[a, a]
6
7      # have to because they have different E0's (their energy scales are all
8      # different)
9      quad_ev = temp_dict["+2"] + temp_dict["-2"] - 2.0 * temp_fake_E0_array
10     for a in range(A): # make sure we multiply the diagonal by ha2ev
11         quad_ev[a, a] *= ha2ev
12
13     quad_ev /= (2. * qsize) ** 2
14     for a in range(A): # make sure we subtract the vibron_ev on the diagonal
15         quad_ev[a, a] -= vibron_ev[i]
16
17     return quad_ev

```

Listing 3.7: Evaluating quadratic coupling term and conversion of units

Bilinear coupling terms

```

1  def _compute_using_array_style(temp_dict):
2      """
3          Remember that the diagonal of 'temp_dict' are in hartrees,
4          while the off-diagonal are in eV.
5      """
6      bilin_ev = np.zeros(shape)
7
8      bilin_ev += temp_dict['++'] - temp_dict['+-']
9      bilin_ev += temp_dict['--'] - temp_dict['-+']
10
11     for a in range(A): # make sure we multiply the diagonal by ha2ev
12         bilin_ev[a, a] *= ha2ev
13
14     bilin_ev /= (2. * qsize) ** 2
15
16     return bilin_ev

```

Listing 3.8: Evaluating bilinear coupling term and conversion of units

3.2.5 Step F: Fitting on n-th order models (Conditional)

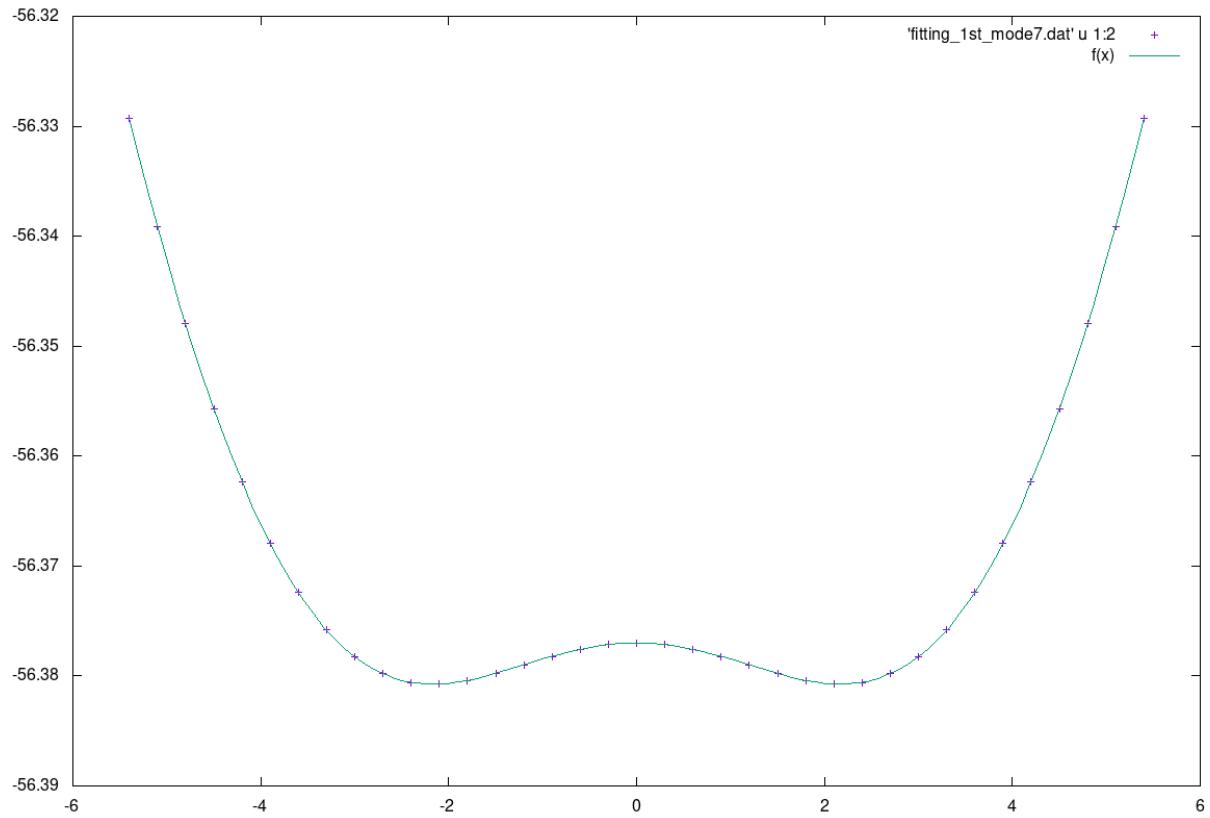


Figure 3.3: PES of D_{3h} NH_3 , Mode 7, 10 Displacements, qsize = 0.30

3.3 Generating spectra via MCTDH and VECC

Negative magnitude in spectra = too high of a tau Useful for MCTDH and VECC alike.

incorporate this sentence: Much of the output format for `dist` was developed to seamlessly connect with Raymond's `project` scripts, and in a mutable style based on model parameters.

3.4 Assorted comments and considerations

3.4.1 Memory mapping

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 the grep command 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 1712 outputs) is 70 megabytes or around 2 million lines of text. When we grepped the SOC data, it took around 4 hours. So to accomplish the intrinsic need for speed up over grepping, 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.

3.5 In Summary

We have stitched together various functionalities such as: text processing, coordinate distortion, input file generation, GAMESS job submission, operator file generation, and even fitting plots. These gratifying extra functionalities were only possible due to the manner in which we wisely implemented the code.

Chapter 4

Realistic Applications to Systems of Interest

4.1 Details about running calculations

The spectra simulations shown here have many parameters that go into influencing their overall shape and appearance. That is to say, they are highly influenced by whichever input model parameters are given and tuned. The interplay of the parameters sometimes even causes unphysical results (negative intensity) or incompatibility. The most important factors include symmetry of the original system, ionic or neutral character, and inclusion of quadratic and higher order terms. However, an important reminder here is that the objective of the research is to not produce beautiful (i.e. resolved, coherent) spectra, nor accurate to experimental. The sole focus is to demonstrate the vibronic model process and its ability to produce demonstrably different results when extra parameters are inputted into the Hamiltonian. A prime example is a case where say a combination of lower SPF/PBF yields a more resolved spectra. Although this might be the case, it is not a good spectra due to its lacking of rdgcheck and rdgpop results. Additionally, in that case we are only sampling the boundary and it is not correct. FINETUNING IS EVERYTHING, THE GOAL IS NOT

TO REPRODUCE EXPERIMENTAL SPECTRA

4.2 Details about producing spectra

A 3–5 sentence paragraph that briefly describes (cite wikipedia or a textbook or something) the basic idea of producing spectra. As previously described (ref to introduction/background somewhere) spectra are produced by fourier transforming auto correlation function (ACF). I used MCTDH's `autospec86` program to do this. Explain more.

Highlight the parameters and how they're connected to the equation (show equation from `autospec` documentation) (borrow stuff from Neil's thesis)

JULY 7th: THE WHOLE PROBLEM FOR DISCREPANCY BETWEEN MCTDH AND VECC WITH SOC IS DUE TO cross (NOT auto) IN INPUT FILE!!!

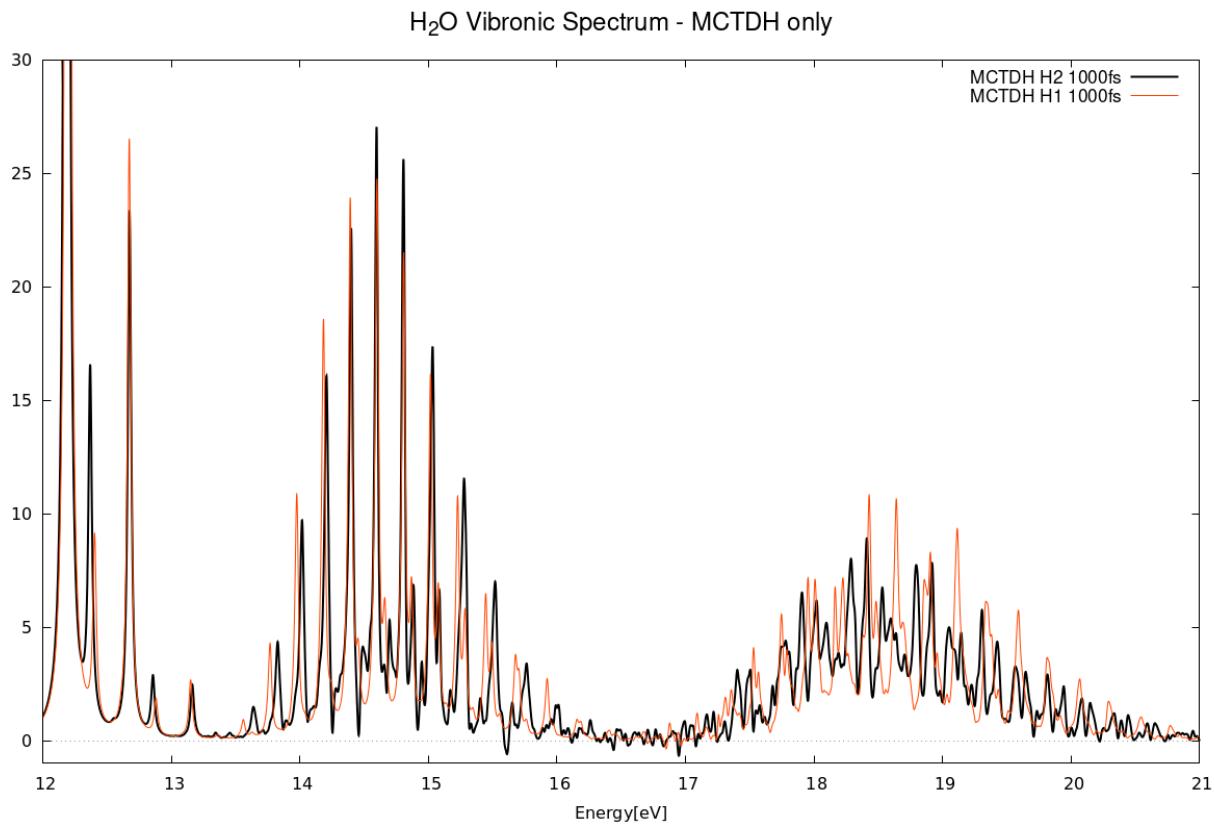
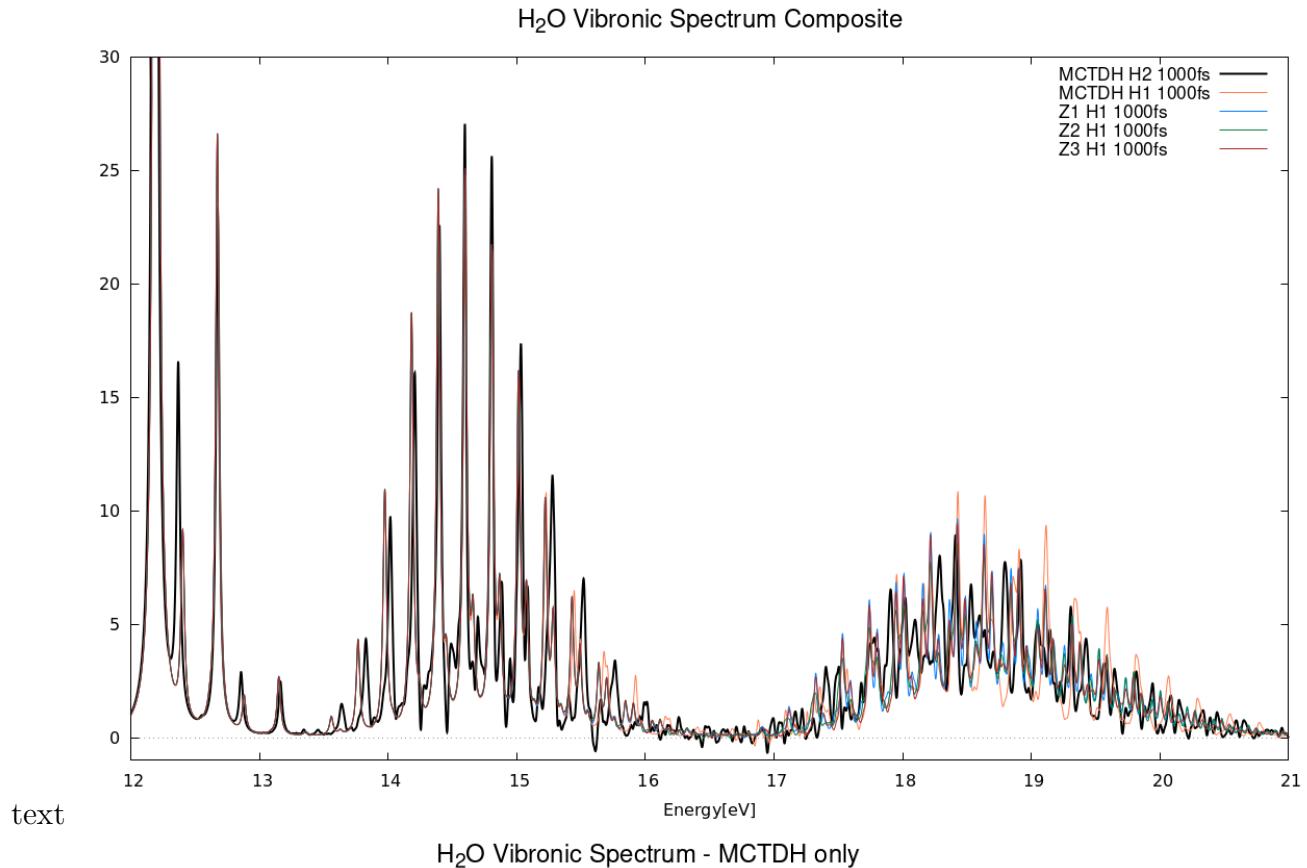
4.3 any other random stuff you need to explain before the systems?

4.4 Preliminary systems

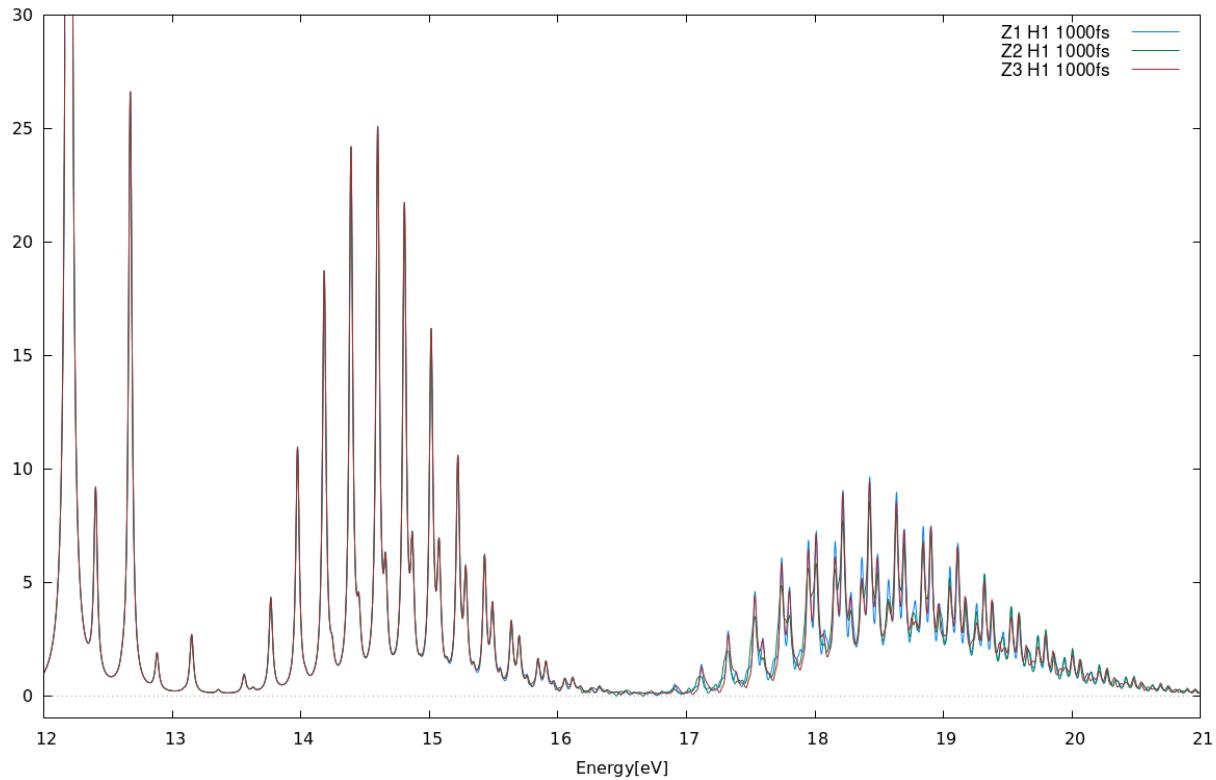
brief description of each system and results you expect and this is to prove that it has no issues in easy cases.

Here I show results for n systems which are all relatively small. The purpose of these systems is to show $|x_i\rangle$. In particular Water is interesting because ...

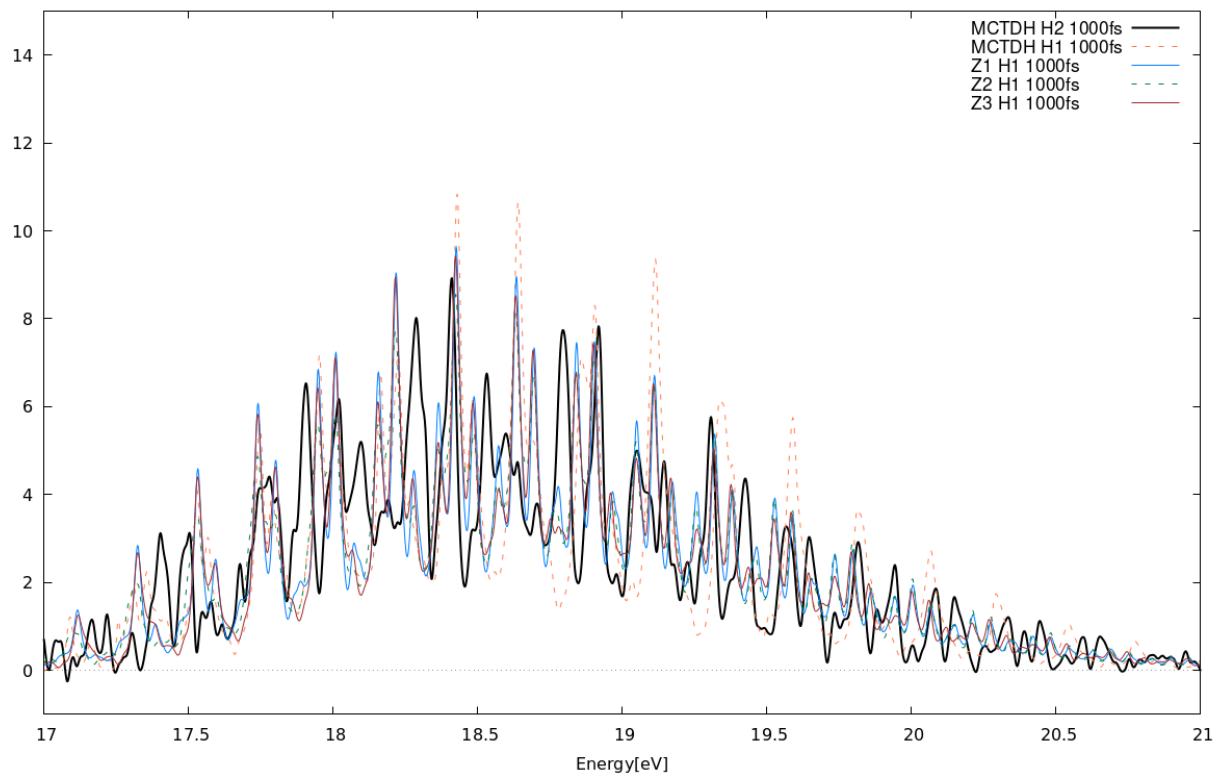
4.4.1 H₂O



H₂O Vibronic Spectrum - VECC only



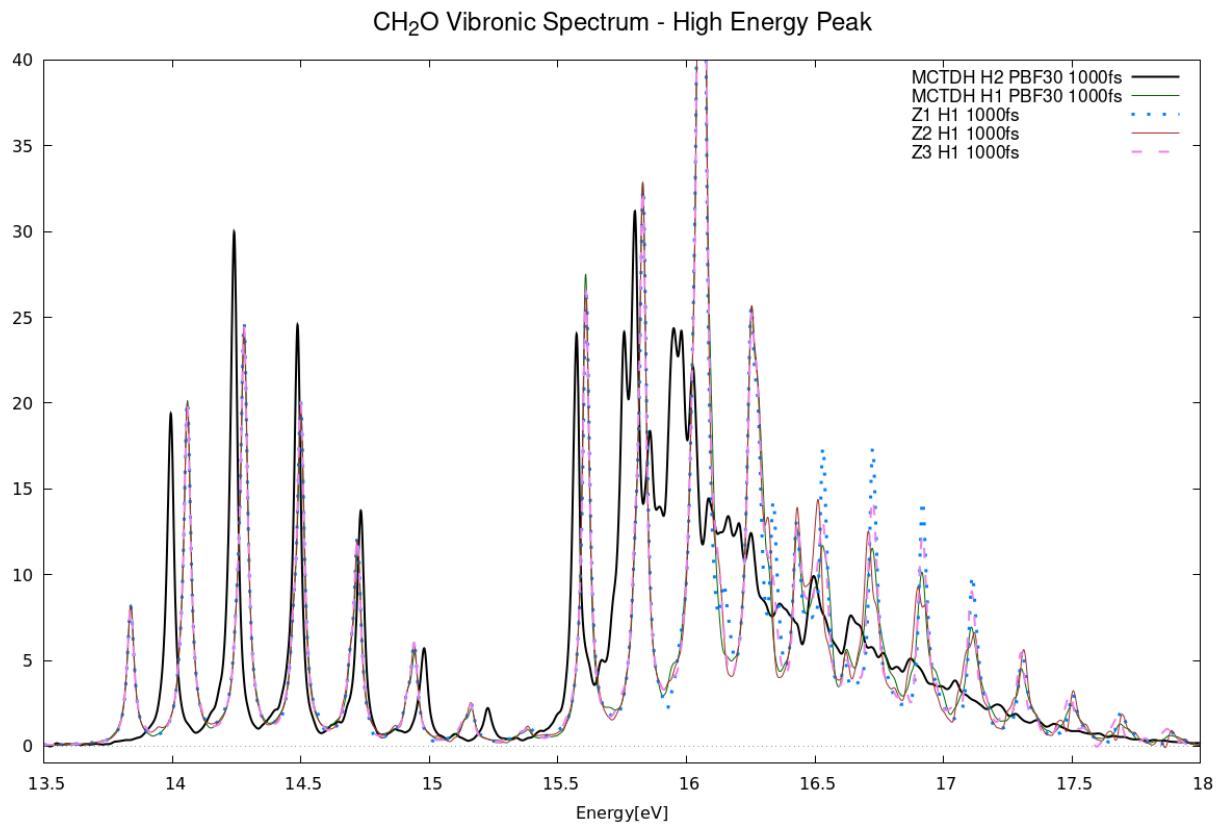
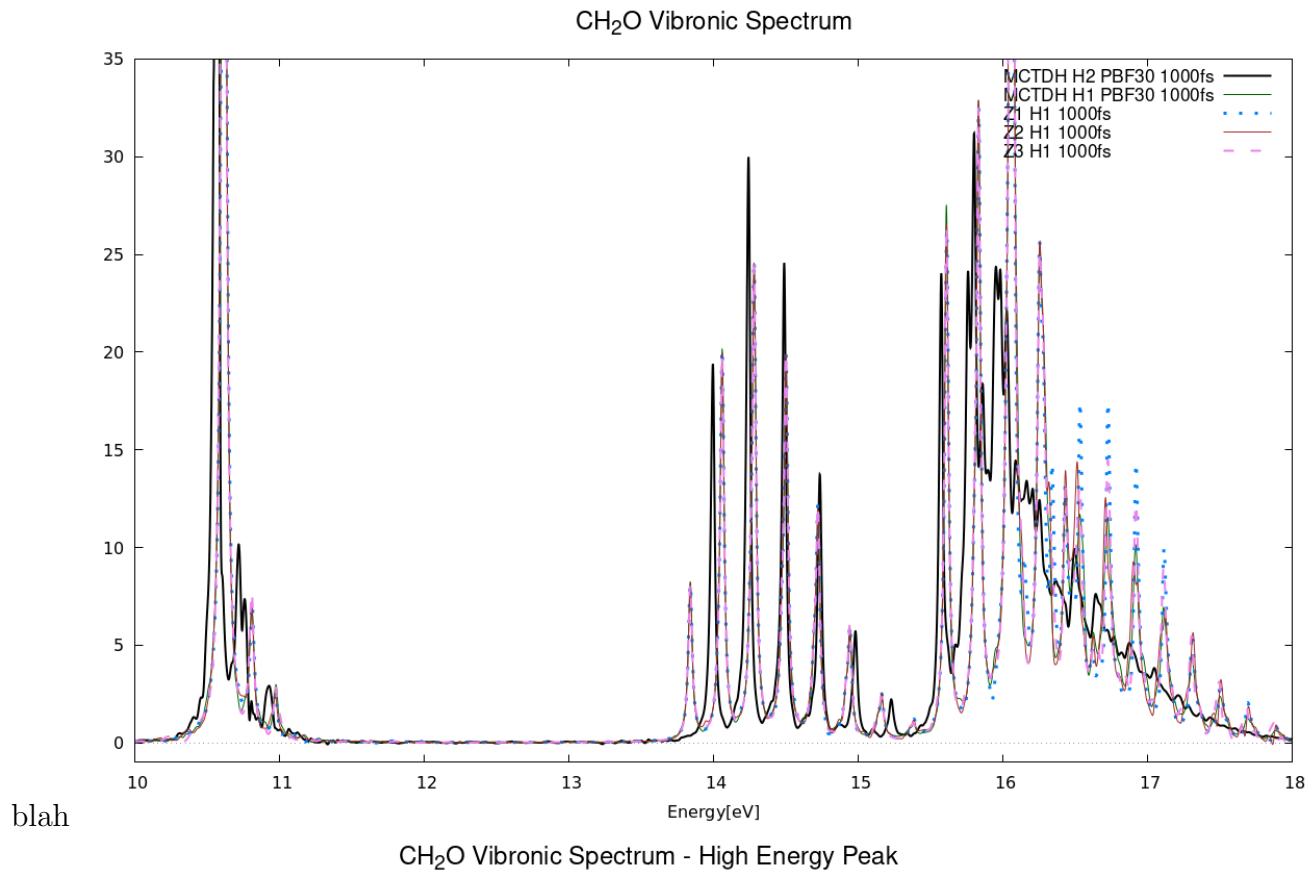
H₂O Vibronic Spectrum - 17 to 21 eV



Water was the first molecular species investigated in this research project and was the original test case for running GAMESS calculations. The active space of H_2O^+ is 6o7e.

```
1      nmofzc=0  nmodoc=1  nmoact=6  nelact=7  mstart(1)=2  icharg = 1  mult = 2  
      gbasis = ccd  state(1) = 4
```


4.4.2 CH₂O

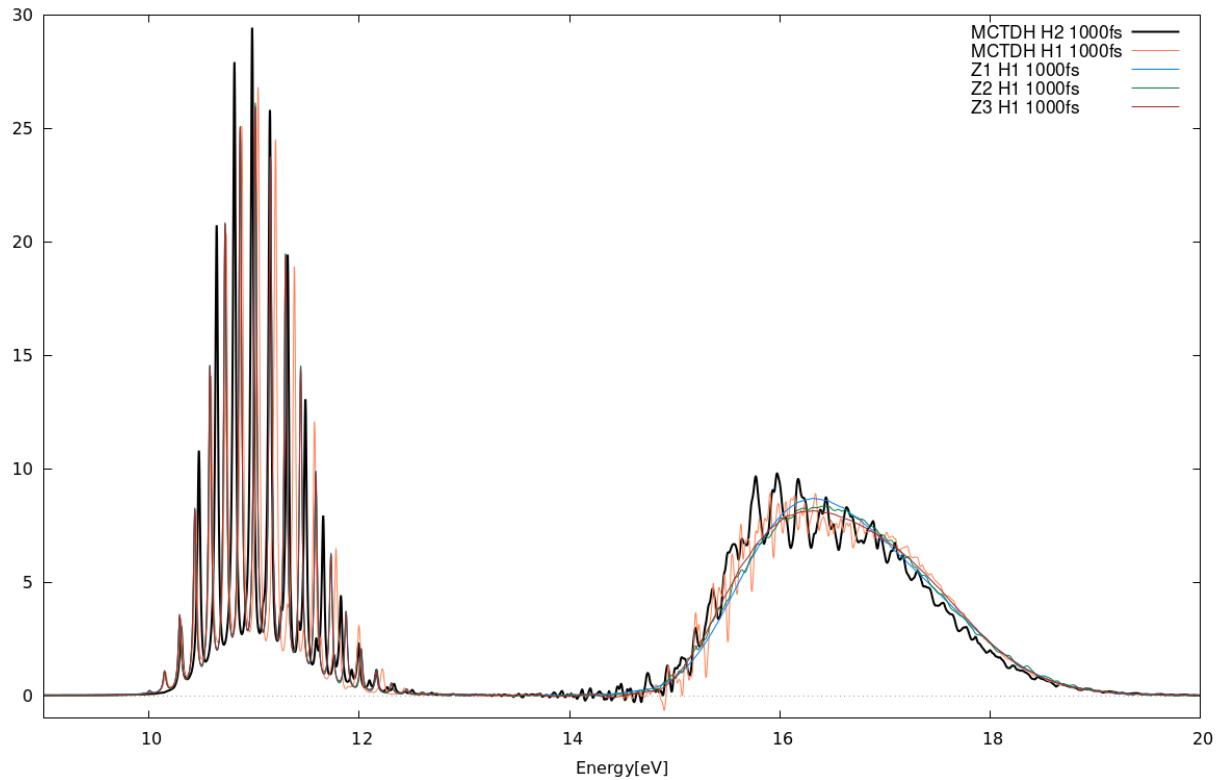


add short description of file show table with Hamiltonian parameters show image of molecule?
(probably unnnecessary) show spectra from MCTDH and VECC (maybe show constant, linear, quadratic, SOC?) discuss the results? are they identical, good this system was just top test xyz, and it behaved as we expected.

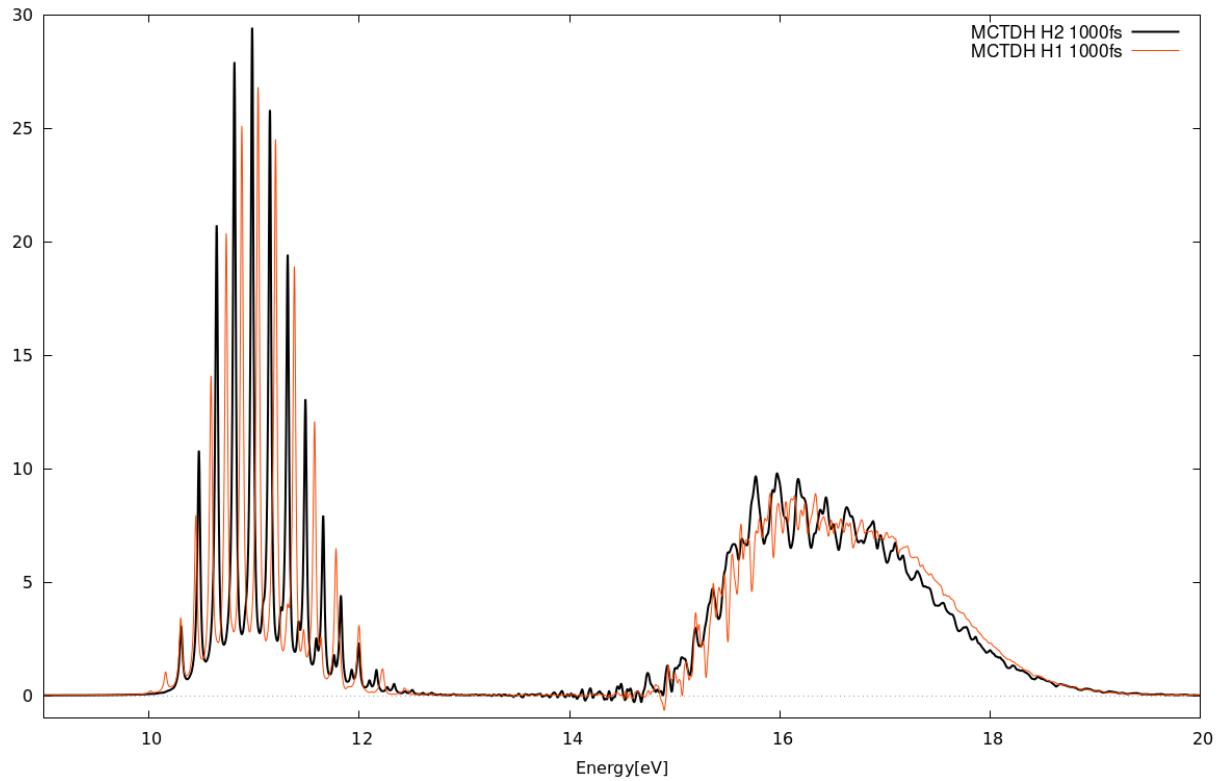
4.4.3 NH3

blahj

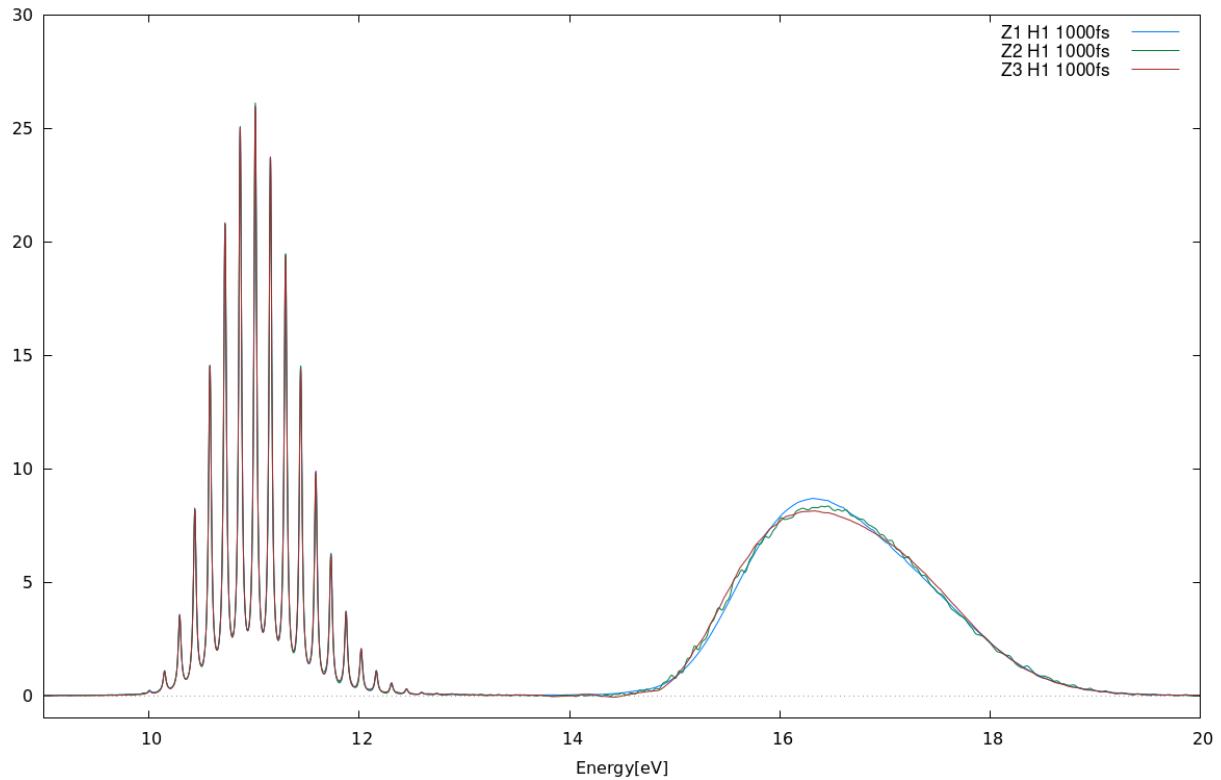
NH₃ Vibronic Composite Spectrum



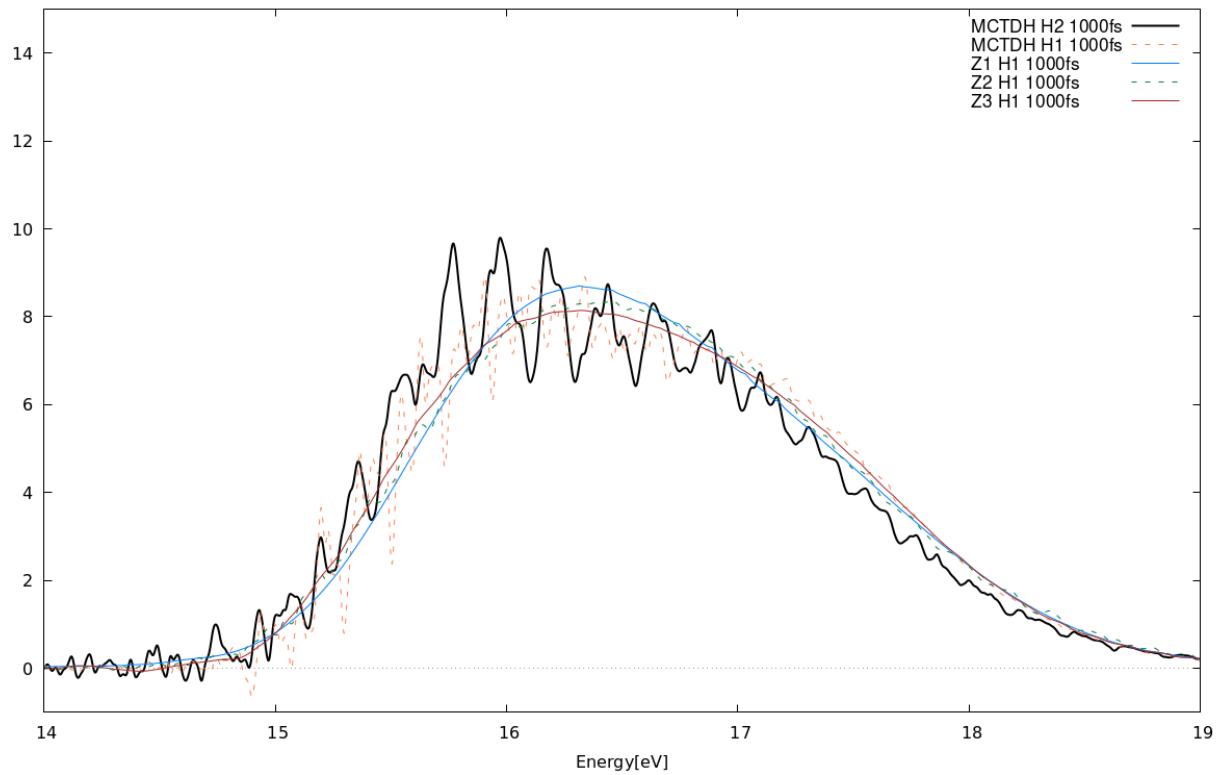
NH₃ Vibronic Spectrum - MCTDH only comparison



NH₃ Vibronic Spectrum - VECC only comparison



NH₃ Vibronic Spectrum Comparison - 14 to 19 eV

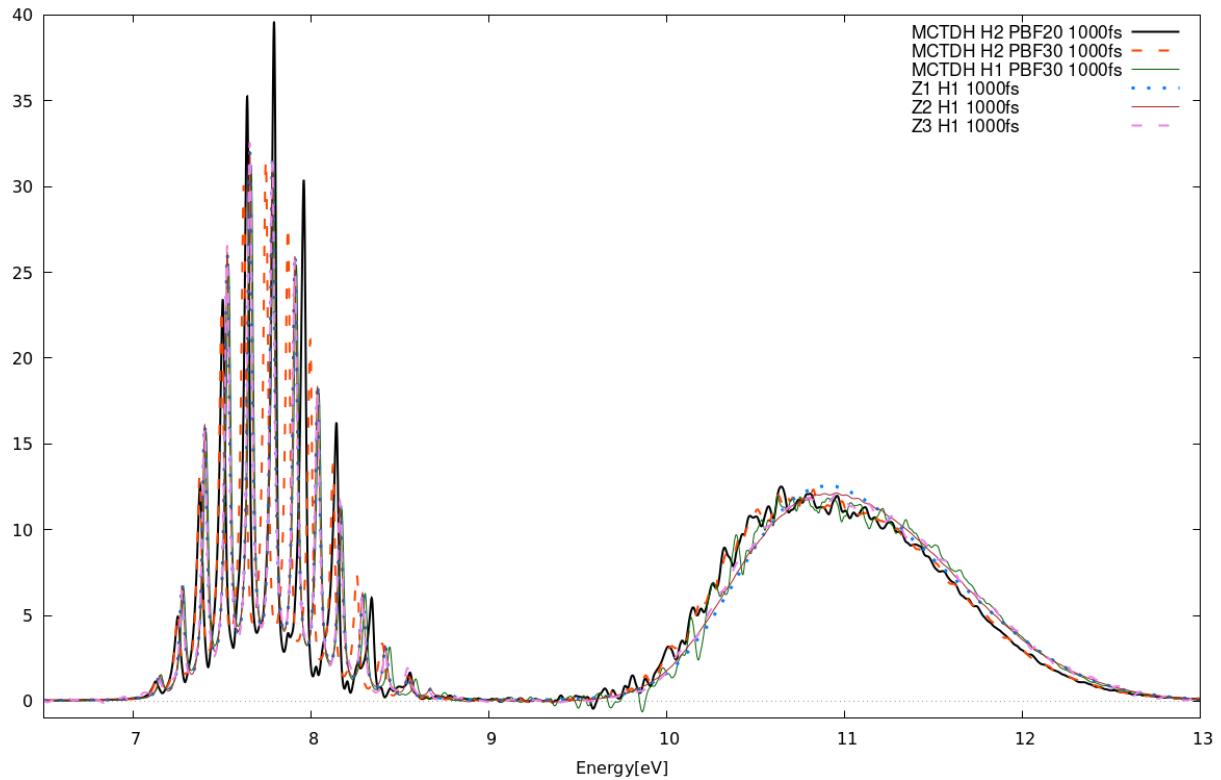


add short description of file show table with Hamiltonian parameters show image of molecule? (probably unnecessary) show spectra from MCTDH and VECC (maybe show constant, linear, quadratic, SOC?) discuss the results? are they identical, good this system was just top test xyz, and it behaved as we expected.

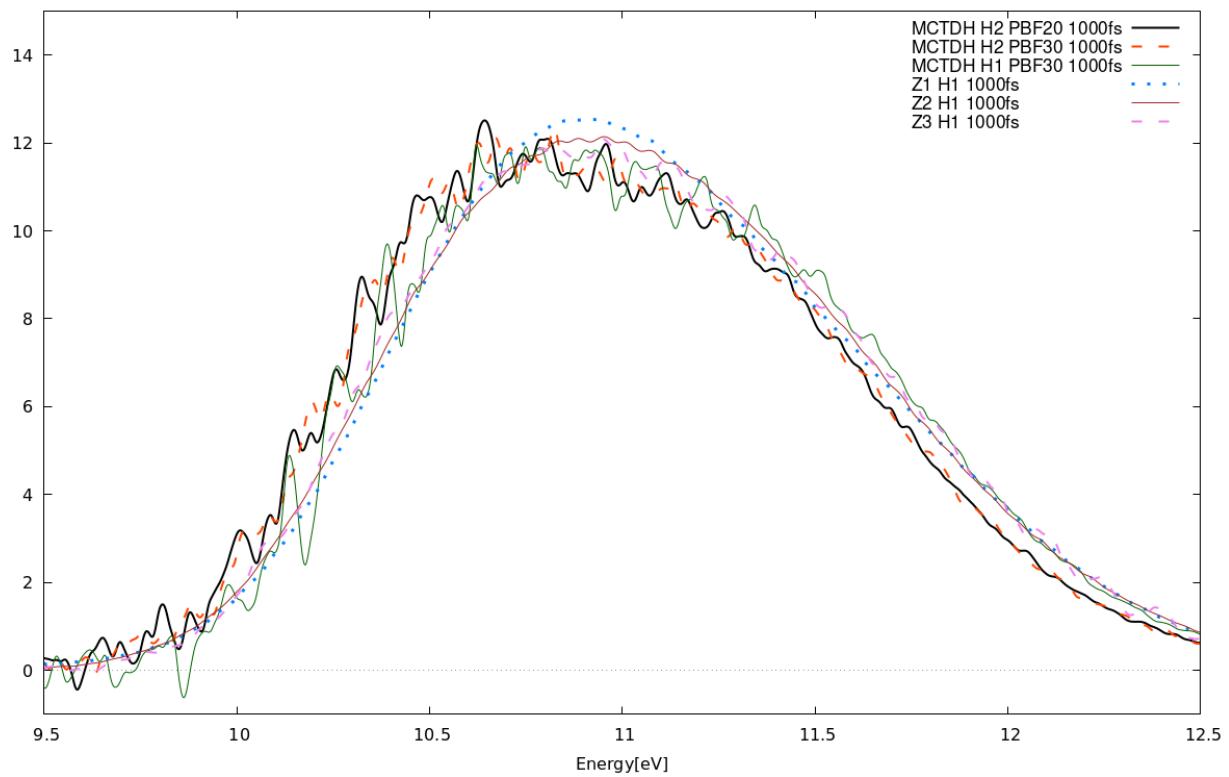
4.4.4 PH3

blah

PH₃ Vibronic Spectrum



PH₃ Vibronic Spectrum - High Energy Peak



4.5 Complex systems

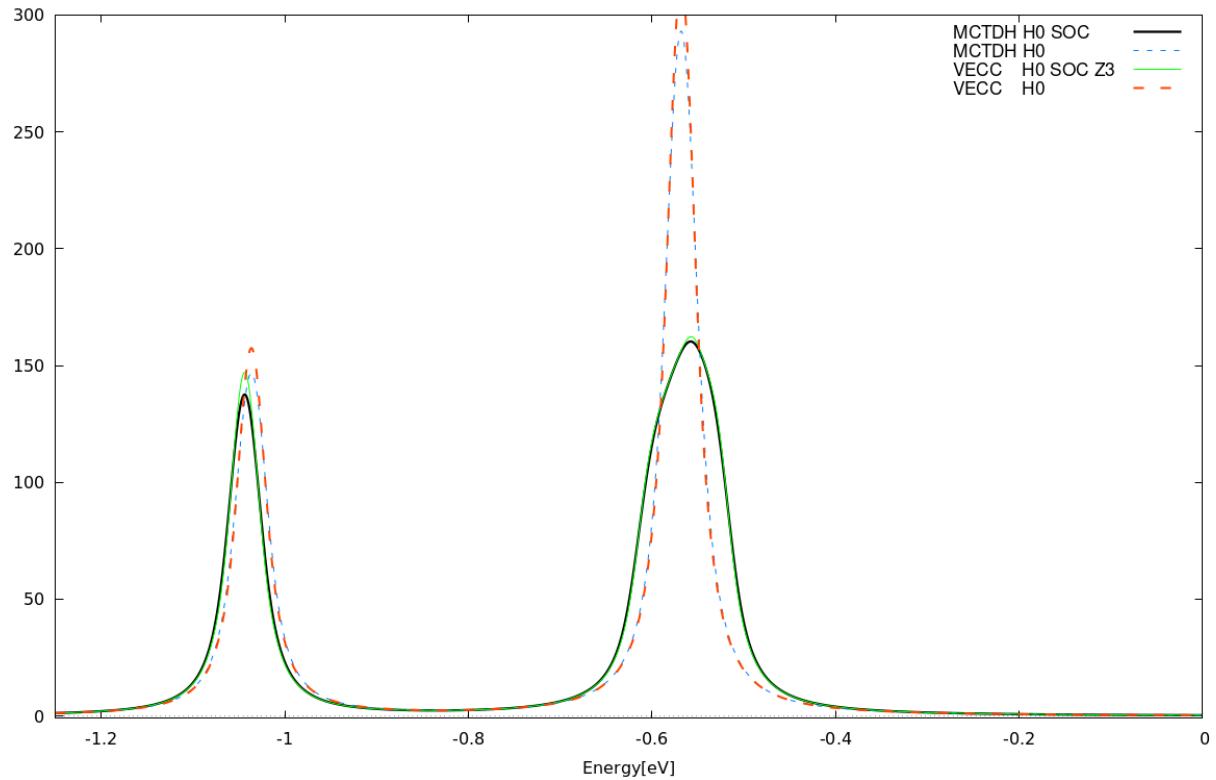
As we explained in (section intro) transition metal complexes are of high importance because for some reason. We primarily focused on CoF₃, RhF₃ and Fe(CO)₅ because of some reason. The two smaller systems (CoF₃, RhF₃) are good for benchmarking the model generation as well as fine-tuning the calculation parameters for MCTDH and VECC. The large system (Fe(CO)₅) is expected to be very difficult and has only some citation of a paper and demonstrate how it's really tough and scary.

To align VECC and MCTDH, it was a monumental task to get things right. We had to consult with Dieter Meyer! Finetuning. Hurdles as Marcel would say... Once they did align, it was a wonderful moment, almost breathtaking. IMAGINARY CHANGES EVERYTHING! Concerted effort to figure it all out. Full capacity to do SOC

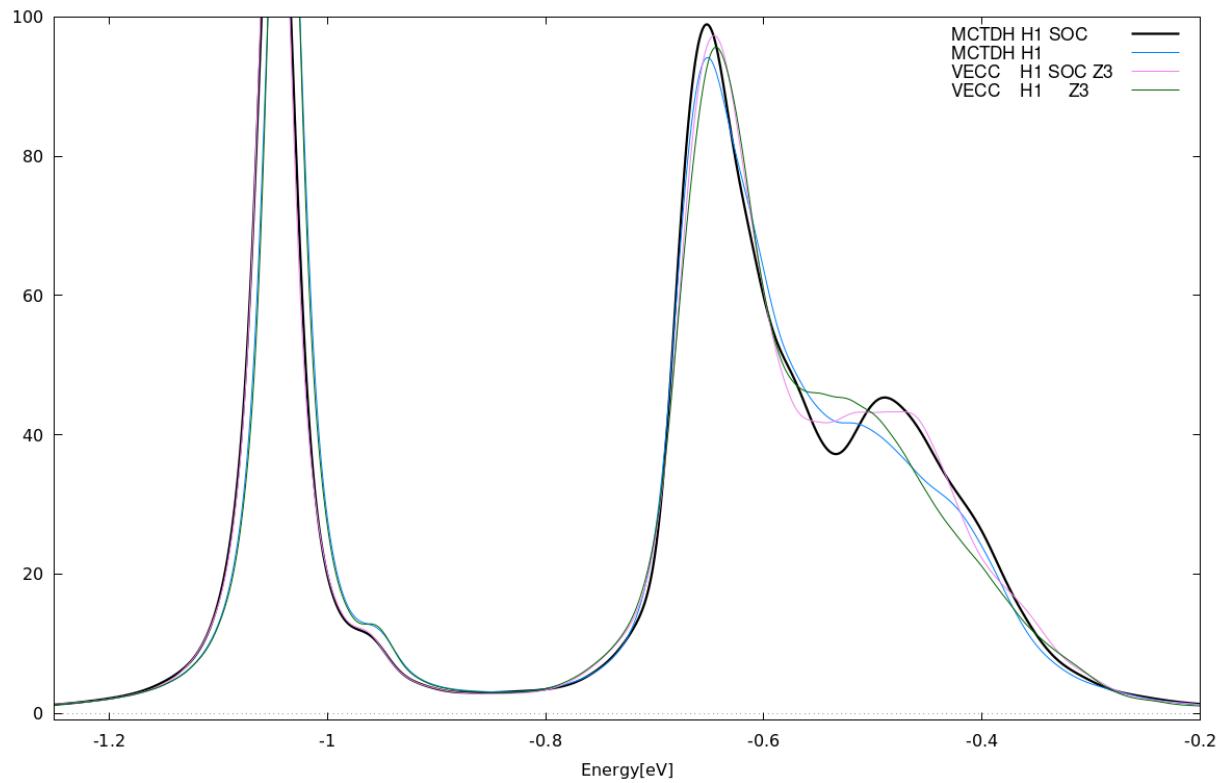
4.5.1 CoF₃

Tetraatomic

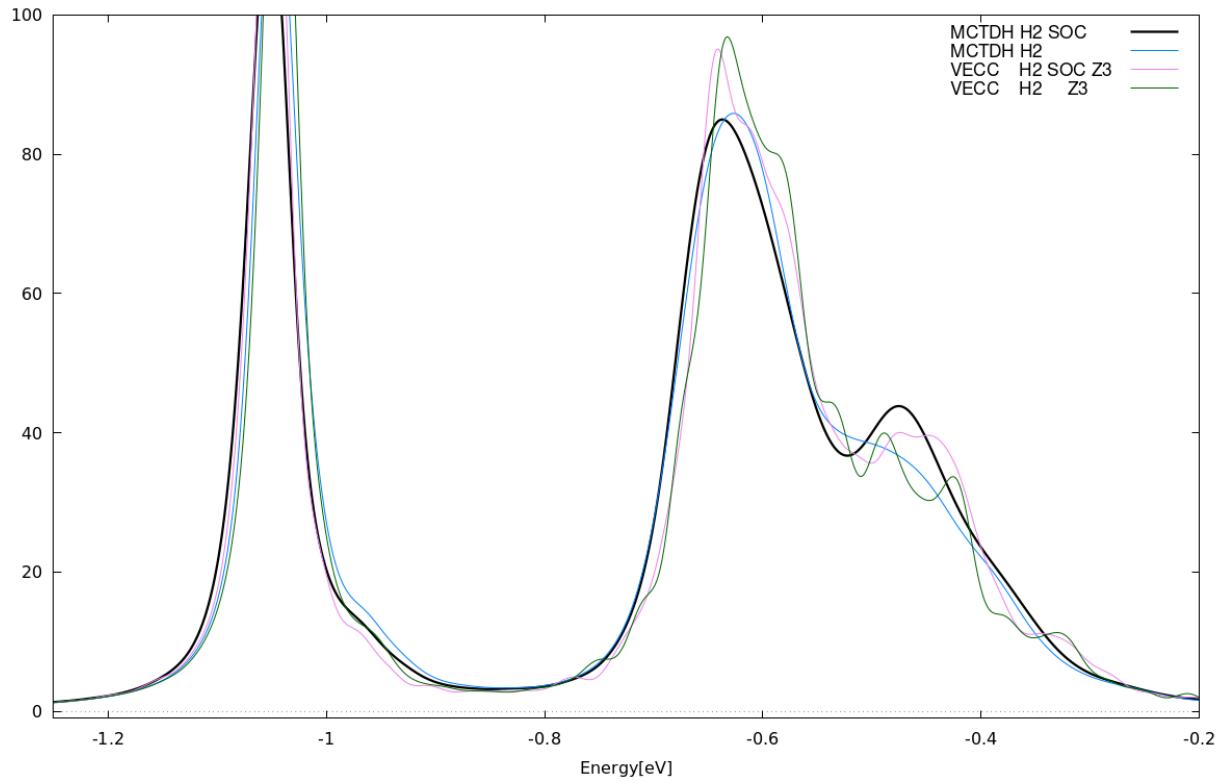
CoF₃ Vibronic Spectrum Composite - Constants



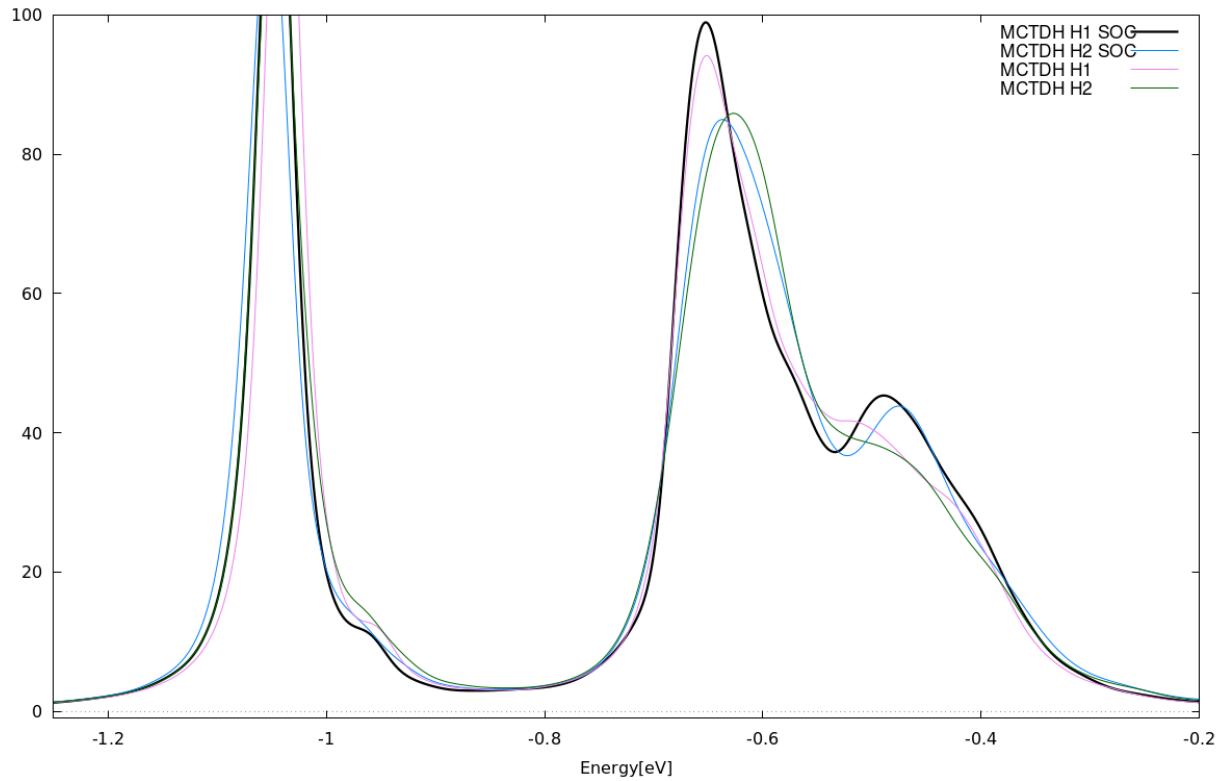
CoF₃ Vibronic Spectrum - Linear Model Comparison



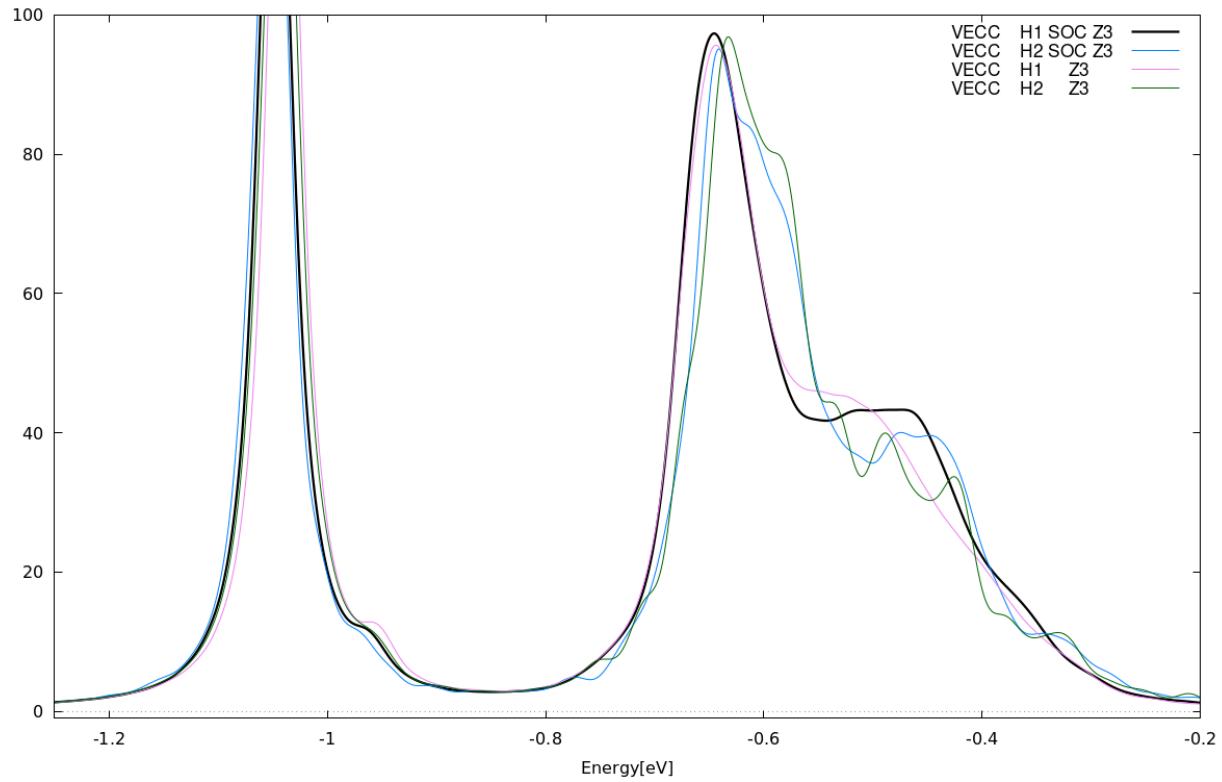
CoF₃ Vibronic Spectrum - Quadratic Model Comparison



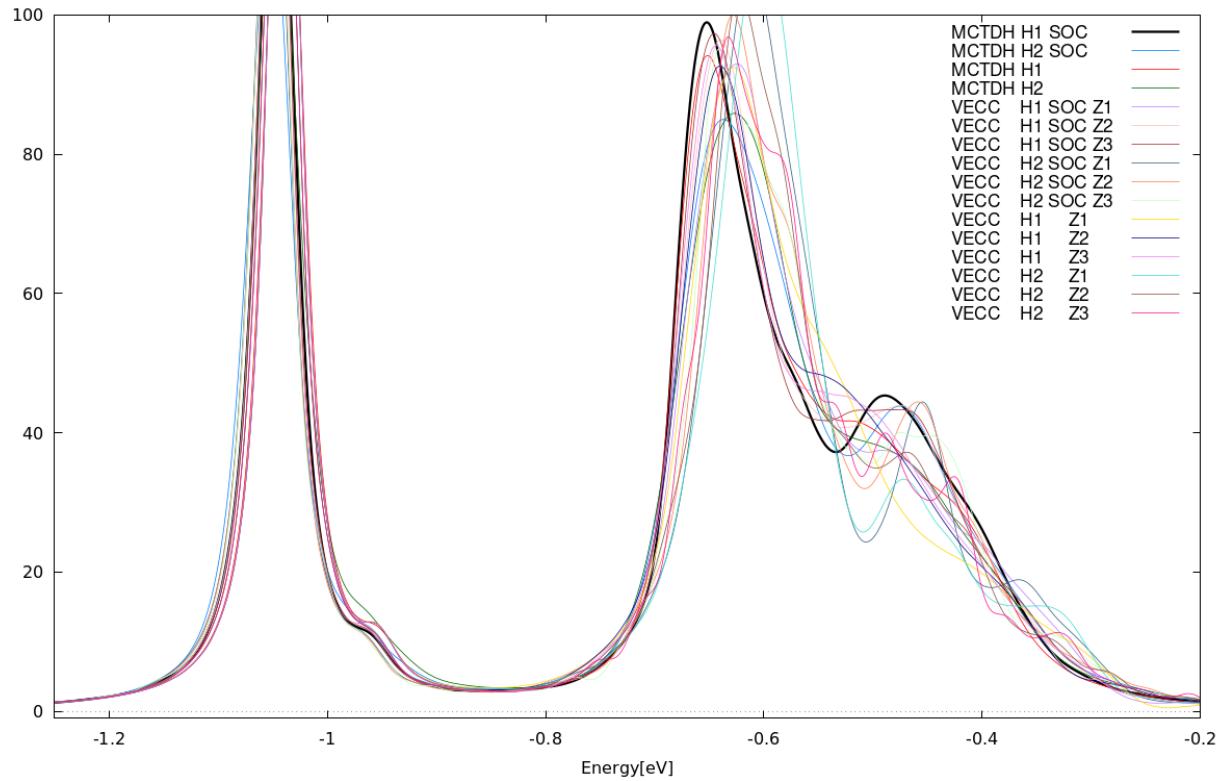
CoF₃ Vibronic Spectrum - MCTDH SOC Effect Comparison



CoF₃ Vibronic Spectrum - VECC SOC Effect Comparison



CoF₃ Vibronic Spectrum Composite



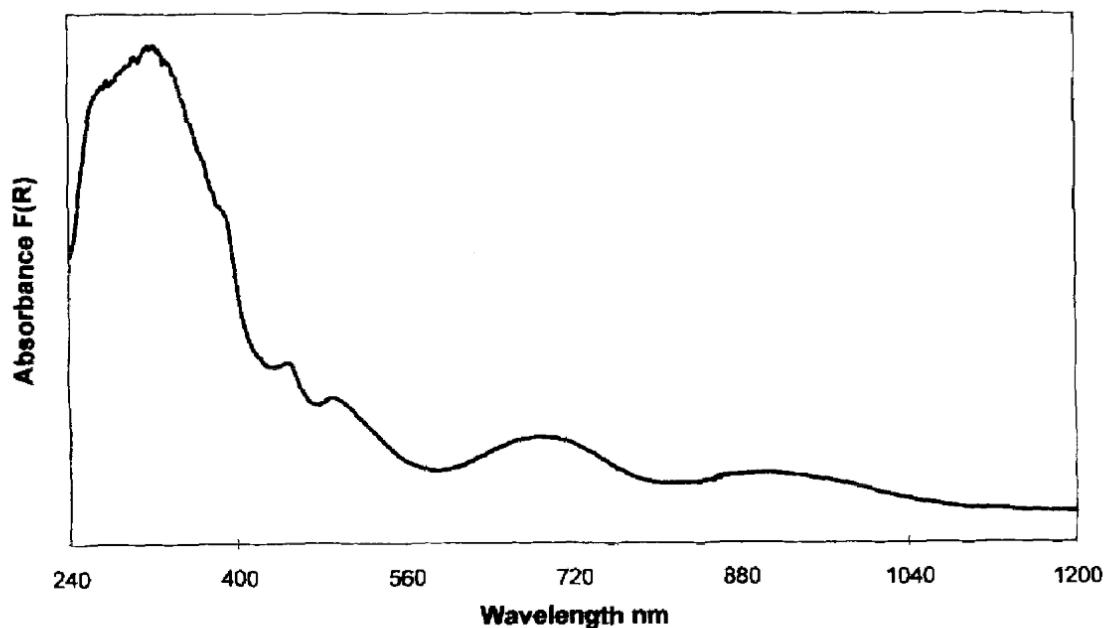
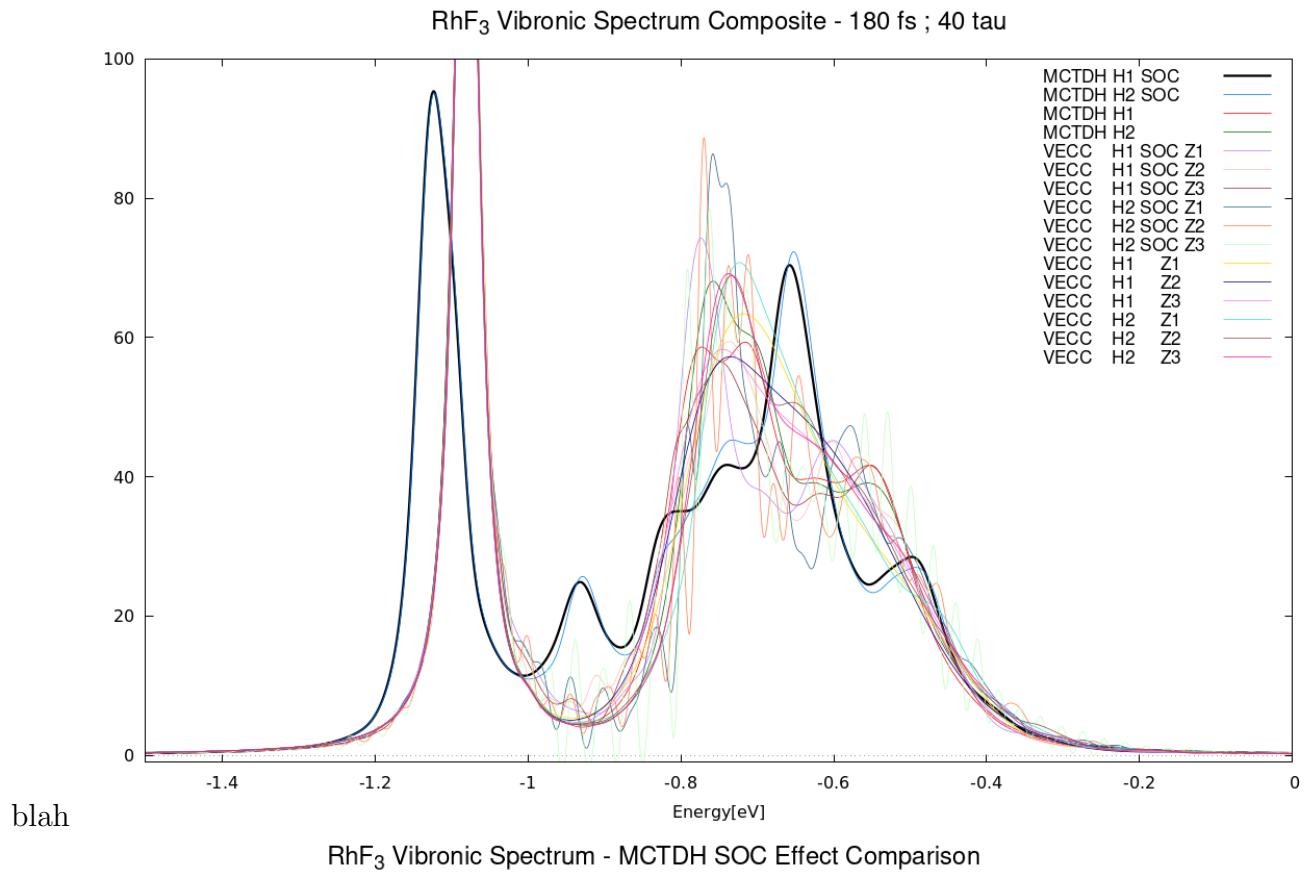


Fig. 3. Diffuse reflectance spectrum of CoF_3 .

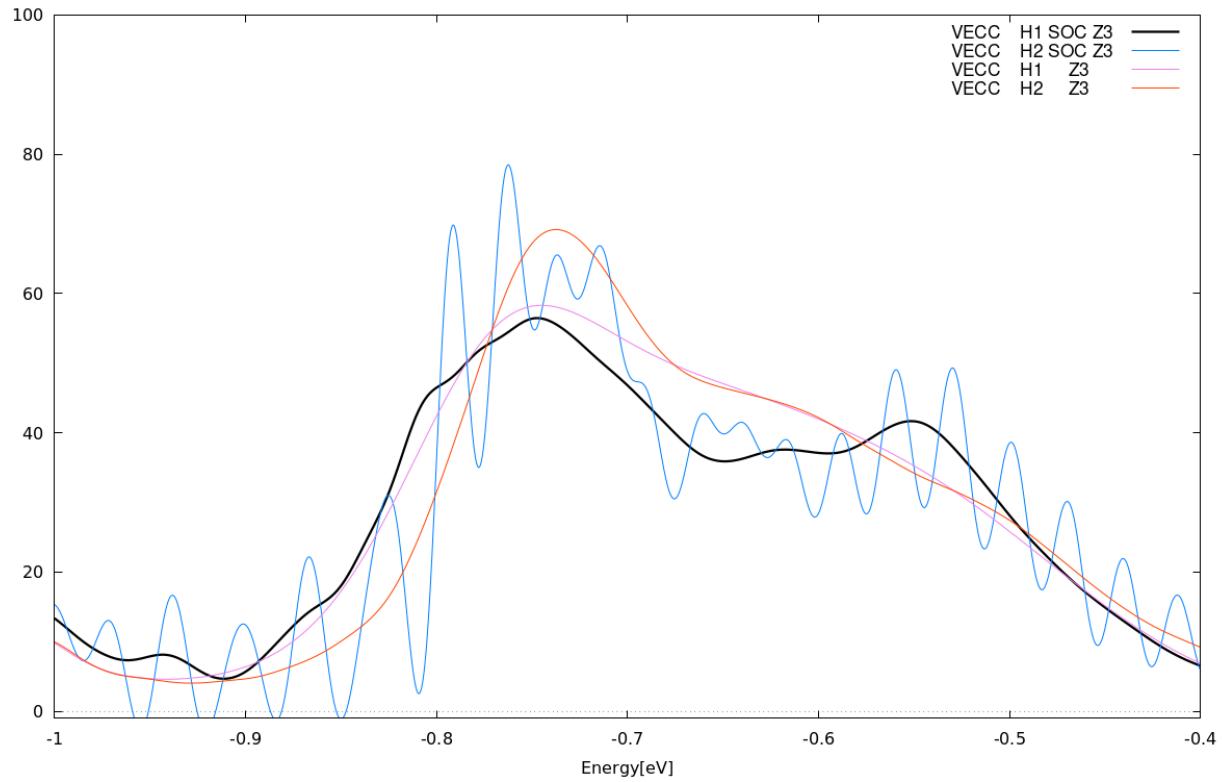
taken from "UV-visible spectroscopic studies of group 8-10 metal trifluorides" //

add short description of file show table with Hamiltonian parameters show image of molecule? (probably unnecessary) show spectra from MCTDH and VECC (maybe show constant, linear, quadratic, SOC?) discuss the results? are they identical, good this system was just top test xyz, and it behaved as we expected.

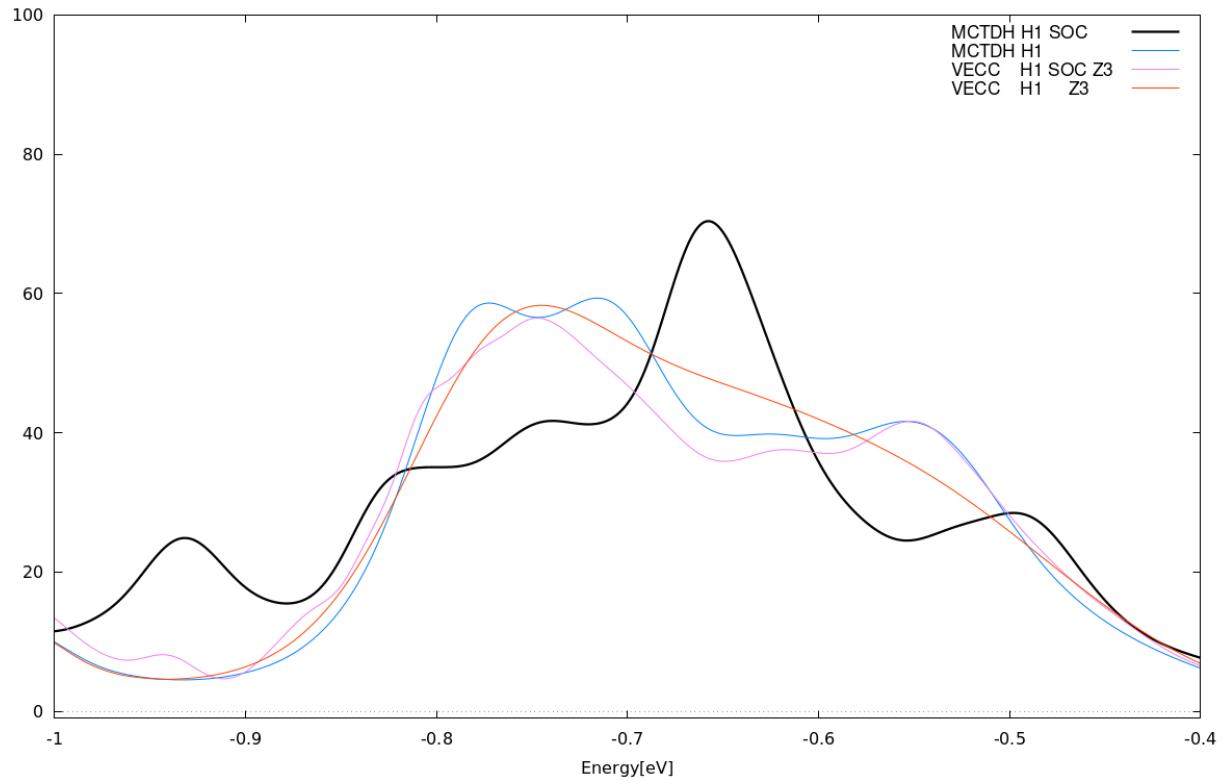
4.5.2 RhF₃



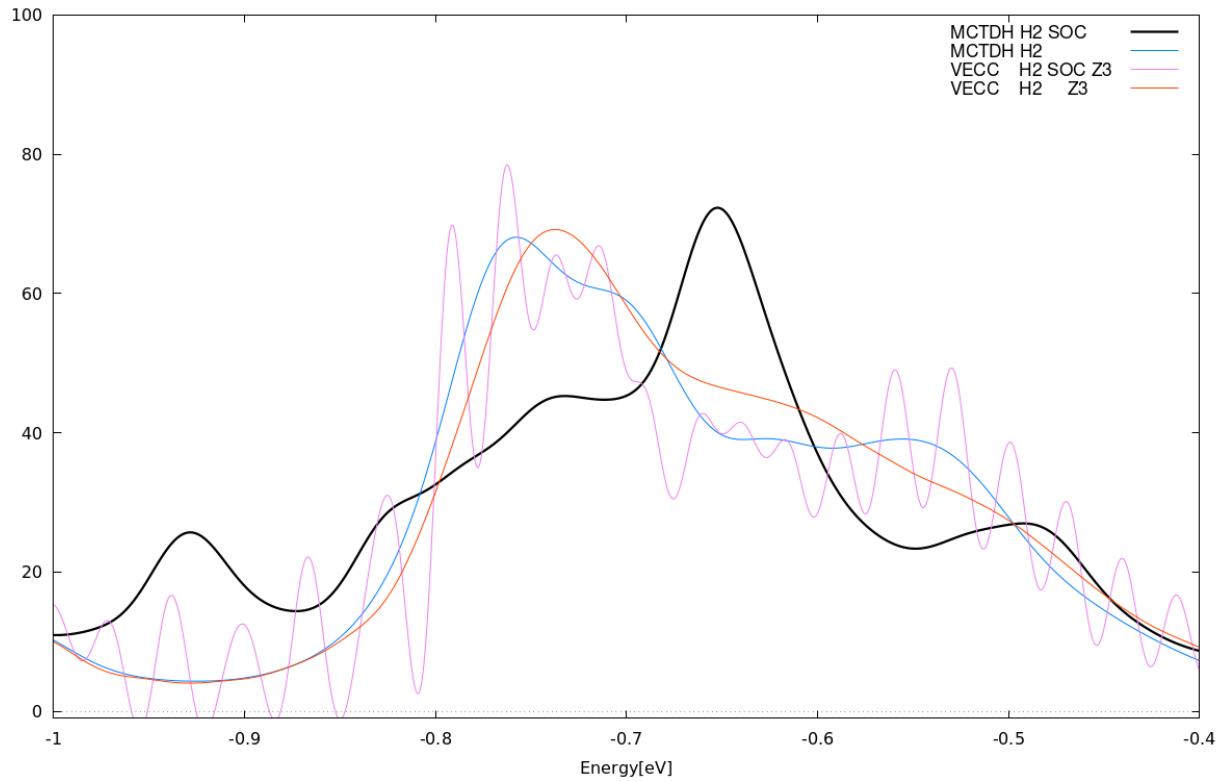
RhF₃ Vibronic Spectrum - VECC SOC Effect Comparison



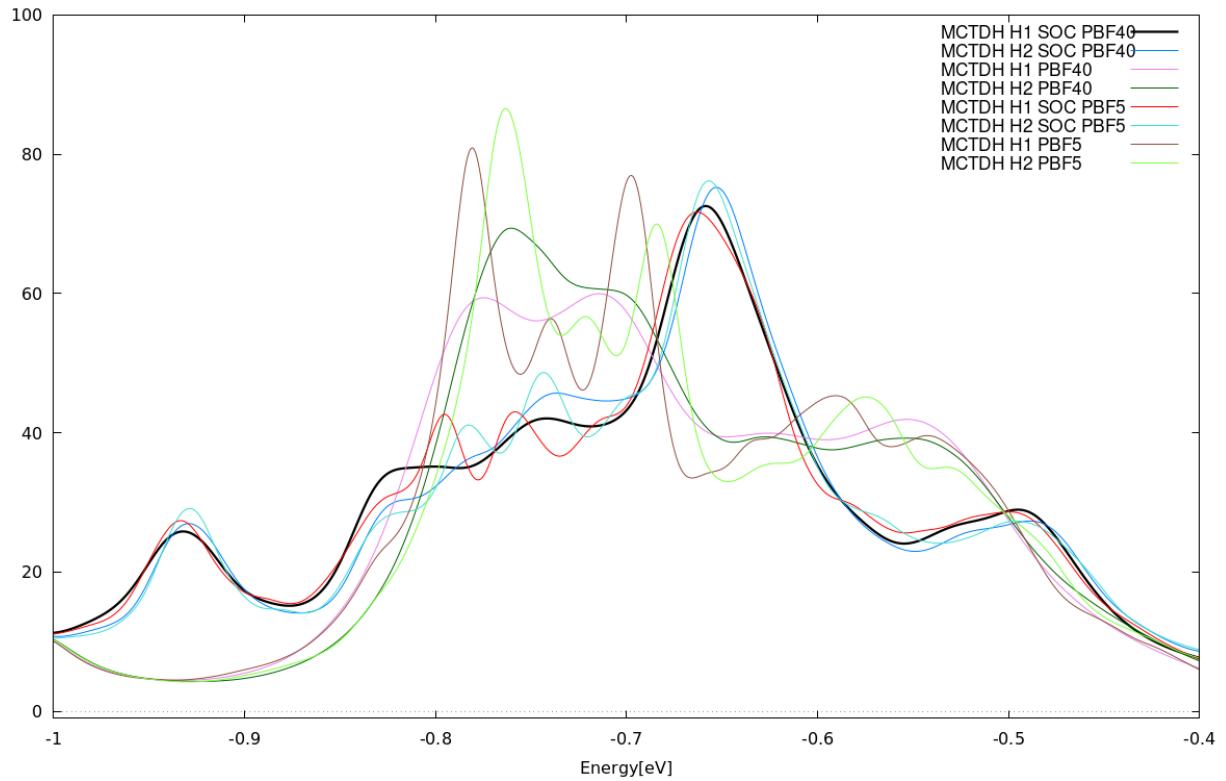
RhF₃ Vibronic Spectrum - Linear Model Comparison



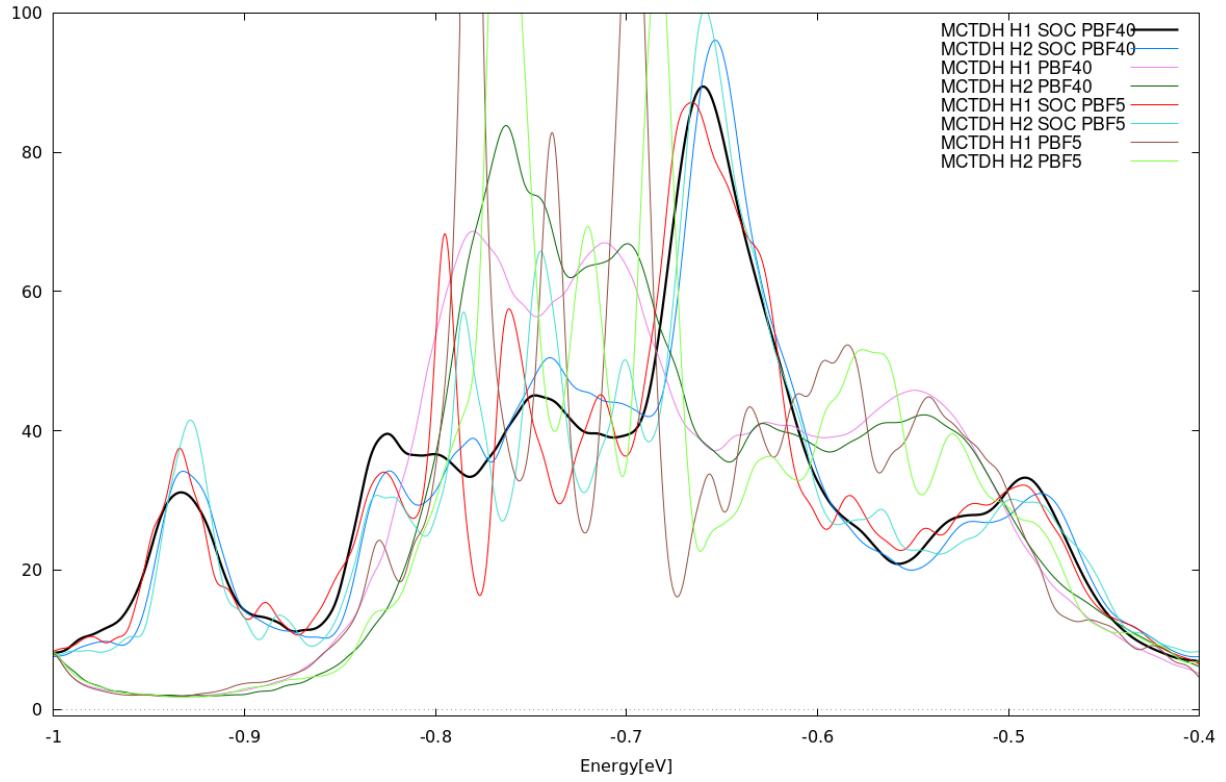
RhF₃ Vibronic Spectrum - Quadratic Model Comparison



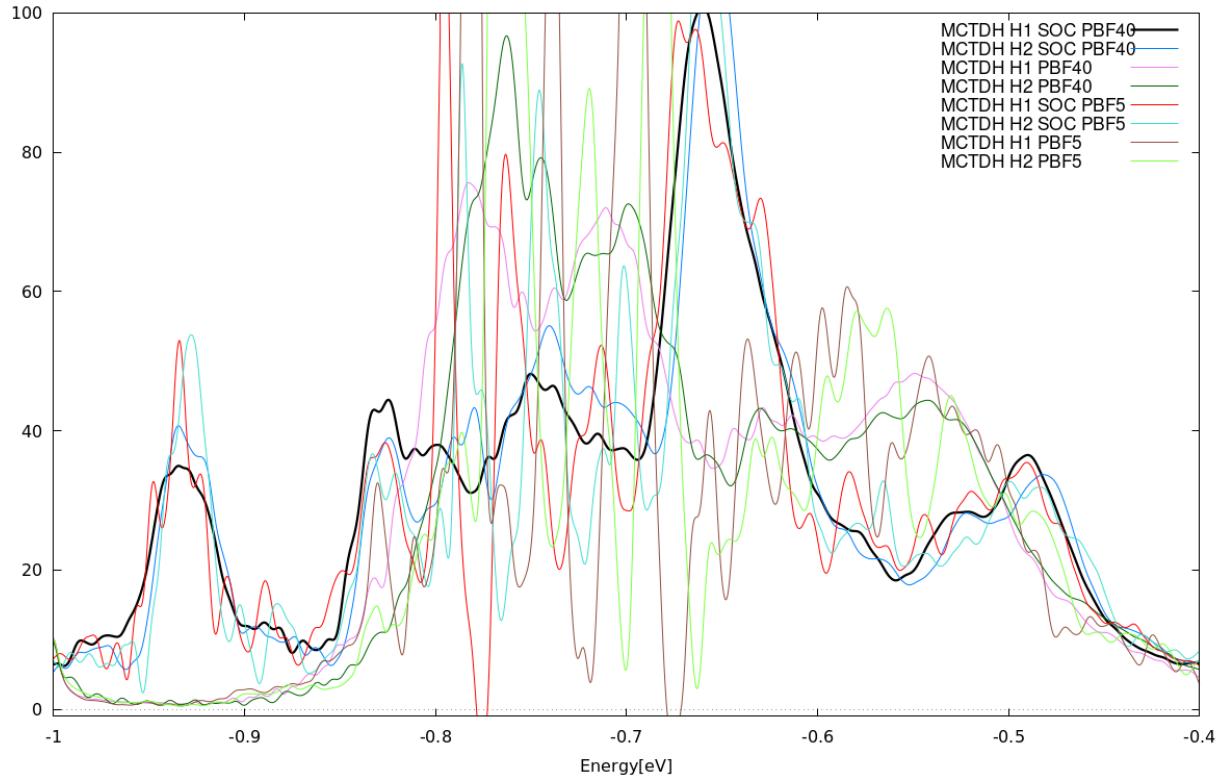
RhF₃ Vibronic Spectrum Composite - 1000 fs ; 40 tau



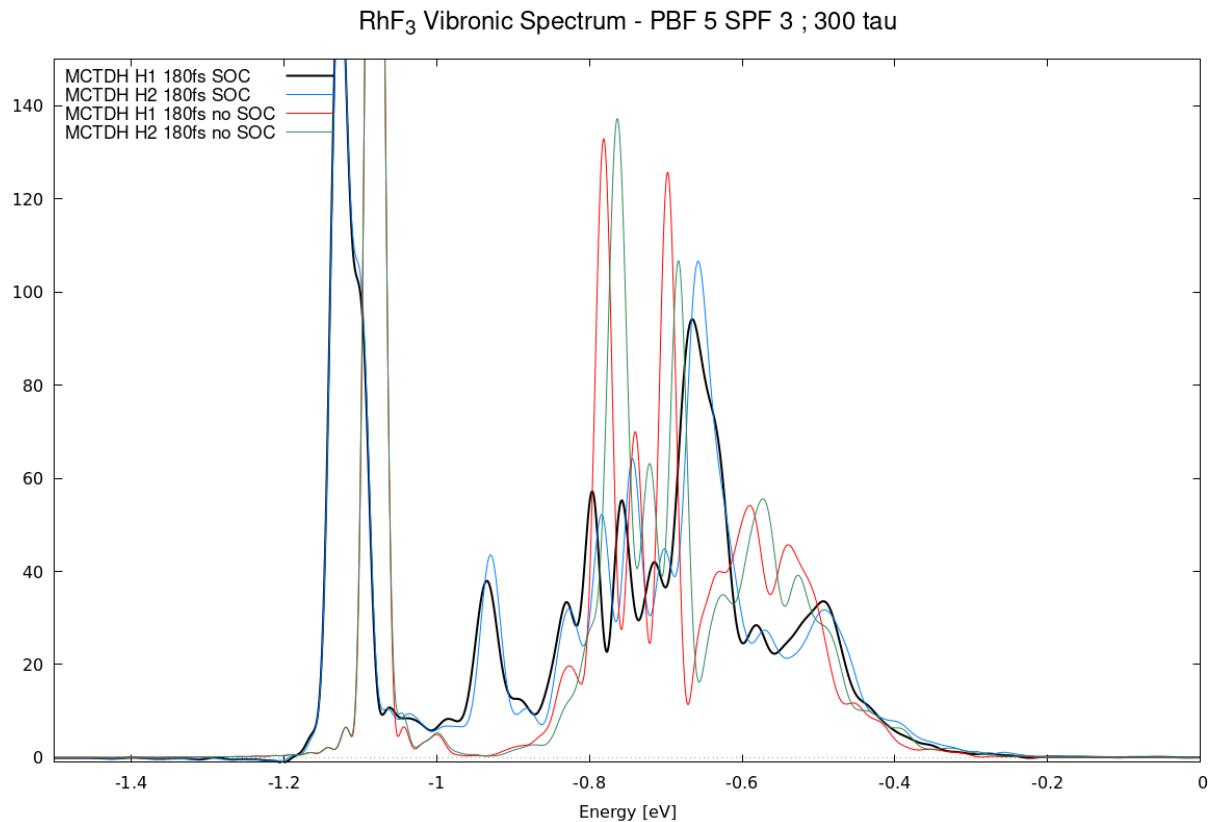
RhF₃ Vibronic Spectrum Composite - 1000 fs ; 90 tau



RhF₃ Vibronic Spectrum Composite - 1000 fs ; 200 tau



TAU MAKES A HUGE DIFFERENCE FOR RHF3! ESPECIALLY IF IT IS LONG PROPAGATION MCTDH.



HOWEVER THE PROBLEM IS THAT VECC CANNOT HANDLE LARGE TAU AND CANNOT HANDLE LARGE FS PROPAGATION ELSE DIVERGES SO 180FS 40 TAU IS IDEAL FOR COMPARISON!!

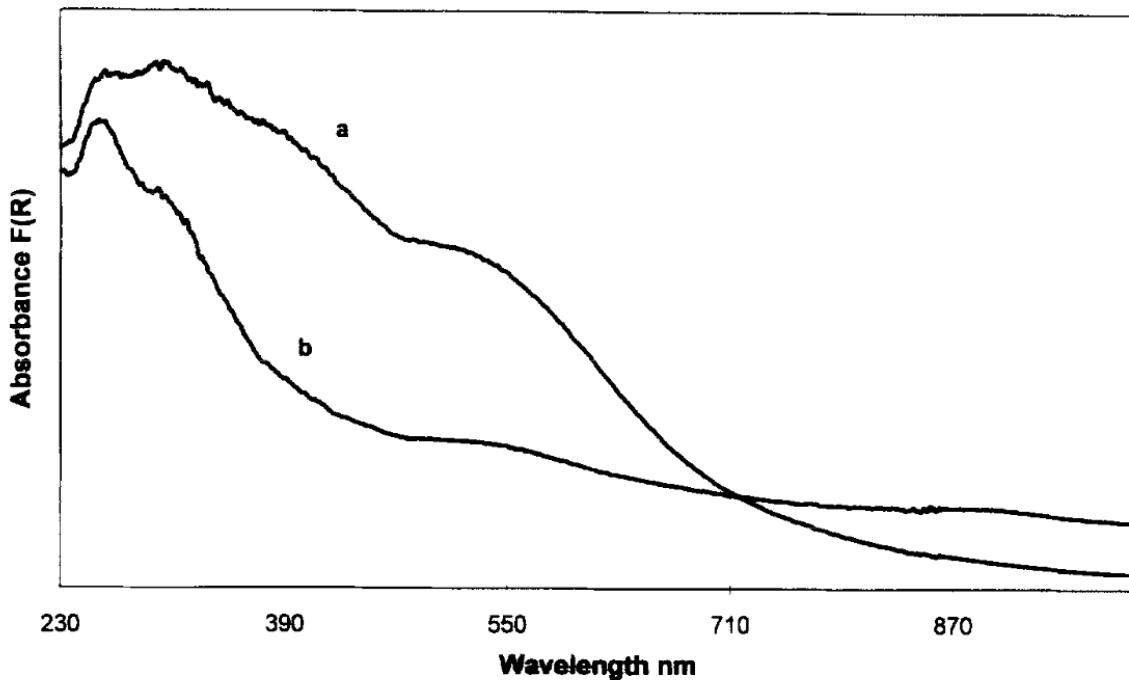


Fig. 4. Diffuse reflectance spectra of (a) RhF_3 and (b) IrF_3 .

taken from "UV-visible spectroscopic studies of group 8-10 metal trifluorides" //

add short description of file show table with Hamiltonian parameters show image of molecule? (probably unnecessary) show spectra from MCTDH and VECC (maybe show constant, linear, quadratic, SOC?) discuss the results? are they identical, good this system was just top test xyz, and it behaved as we expected.

4.5.3 $\text{Fe}(\text{CO})_5$

Postulated, a priori

characterized by

case study of a non-tetratomic transition metal complex system

add short description of file show table with Hamiltonian parameters show image of molecule? (probably unnecessary) show spectra from MCTDH and VECC (maybe show constant, linear, quadratic, SOC?) discuss the results? are they identical, good this system was just top test xyz, and it behaved as we expected.

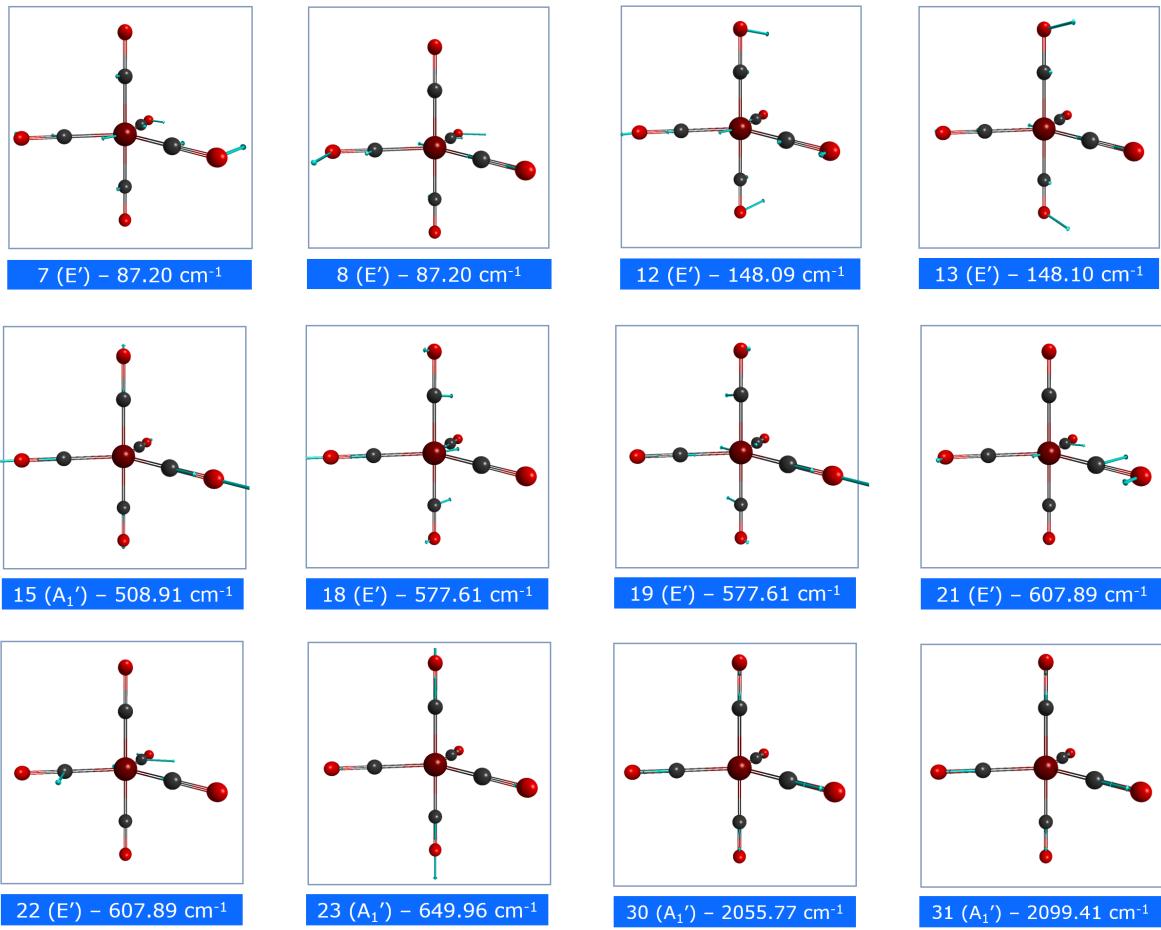


Fig . Screened 12 out of 27 modes.

4.5.4 Vibronic spectra of test case pnictogen hydrides

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

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

carbonyl ligand, due to carbonyls' known stabilizing effect on metals

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 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. Selected screen modes: [7, 8, 12, 13, 15, 18, 19, 21, 22, 23, 30, 31]. 12 out of 27. TZ: show a spreadsheet of screenshots here, displaying the ones we selected, does it make sense according to our analysis?

(see email with Dieter) We have good news. Prof. Meyer advised that the MCTDH-84 version I was running was outdated and that upgrading to the newest 86 version would do the trick. Fortunately Toby had a copy of 86 too on hand. Not only has the parameter limitation seemed to be resolved, but it similarly resolves a secondary issue in 84 where parallelized runs were erroring out (but works now in 86!).

Stitching together various functionalities, tuning

May 29th: we have done plotting of diabatic energies today and saw that mode 15 was equatorial dissociation and mode 23 was axial dissociation. Equatorial dissociation leads to one of the COs on the horizontal plane leaving, resulting in the axial COs to bend down and resulting in Td symmetry. Whereas, axial dissociation results in one CO on principal axis leaving and resulting in C3v symmetry.

An optimal and desirable system for exhibiting both JT and SOC effects is a large iron complex with low-lying degenerate electronic states and carbonyl ligand, due to carbonyls' known stabilizing effect on metals (Nooijen discussion reference)

<https://stackoverflow.com/questions/19412382/gnuplot-line-types#19420678>

We are using ORMAS+GMCPT! <https://ccl.scc.kyushu-u.ac.jp/~nakano/gmcpt.html>

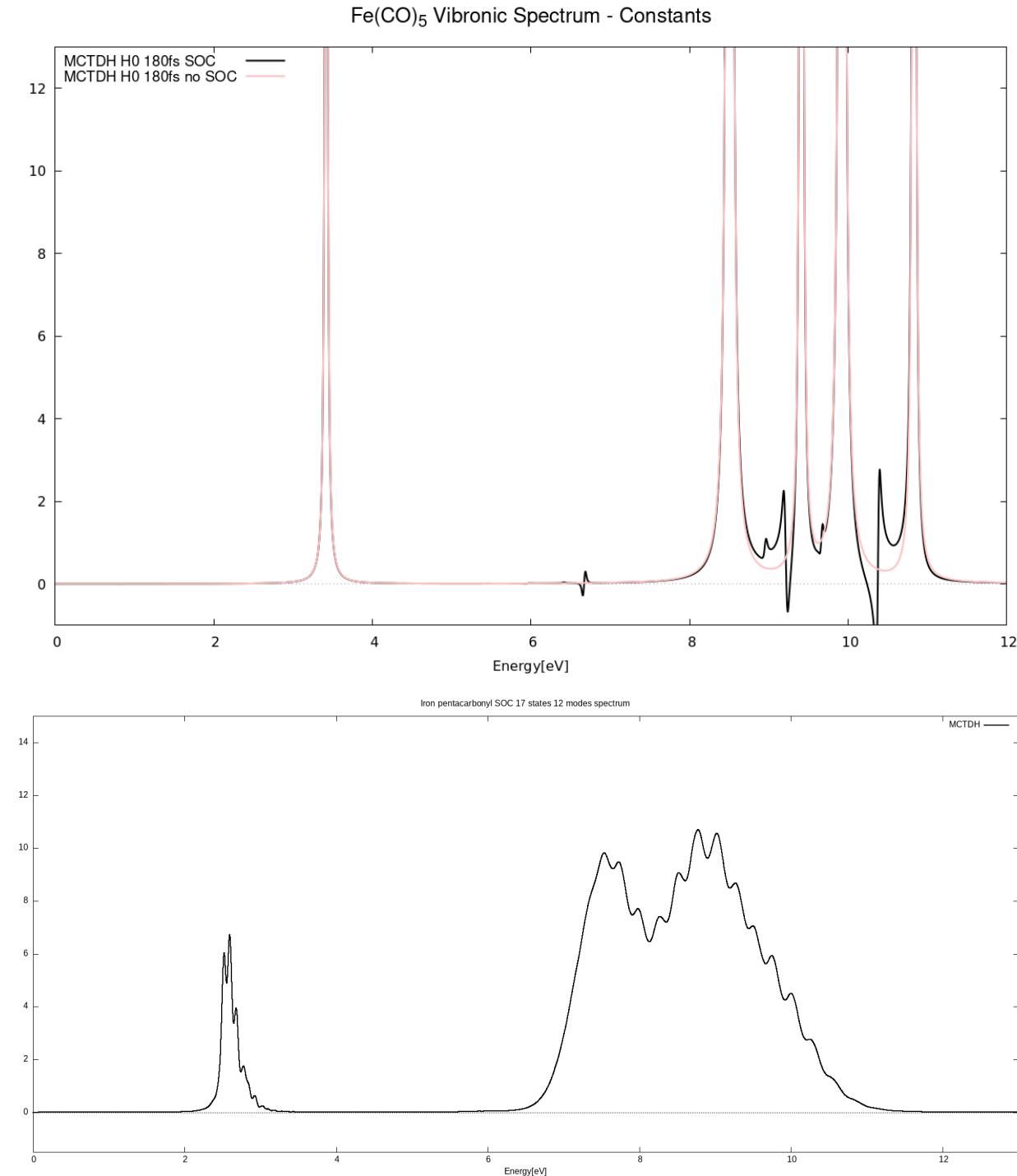
Normally we use Douglas-Kroll (DK) in the `$relfwn` group, however Sapporo basis sets like RELWFN=LUT-IOTC instead.

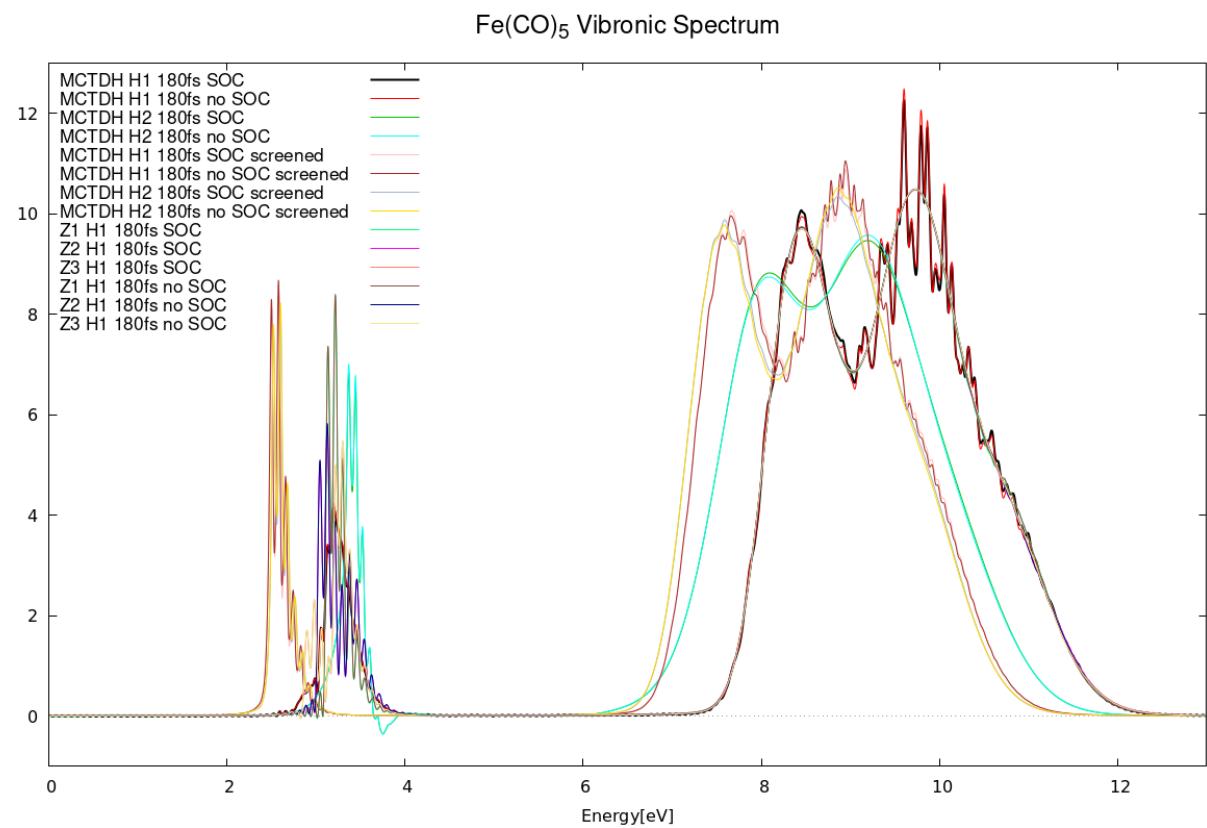
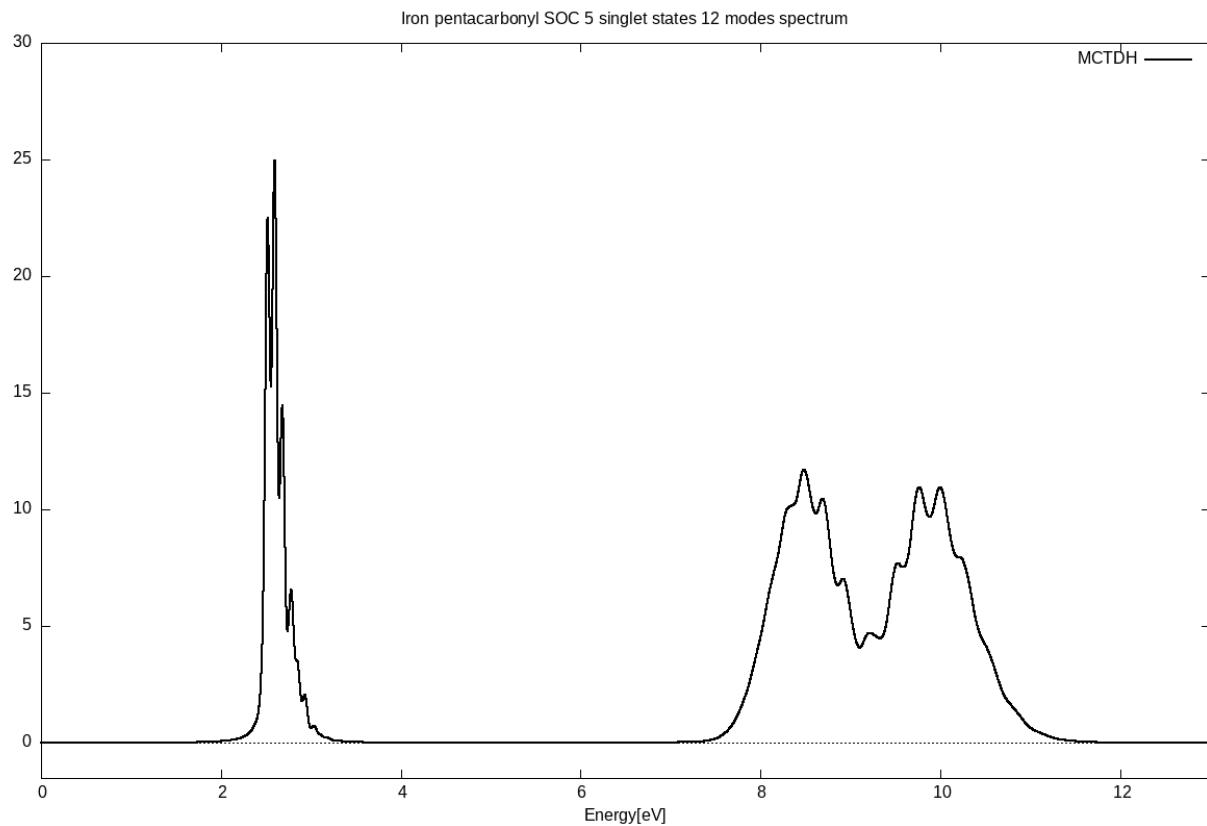
Scale breaks things. We have to fix things, redesign, change this, change that... Emphasize the technical hiccups, airplane dashboard ... It is fickle. On a macro level VECC gets it right, however does not get the fine structure peaks.

We cannot simplify everything down to a one-size fits all simple model. The sheer amount of parameters and caveats each spectra run takes translates into it being incomplete in some way if we do not alter the parameters specific for the case we are looking at. Overall you

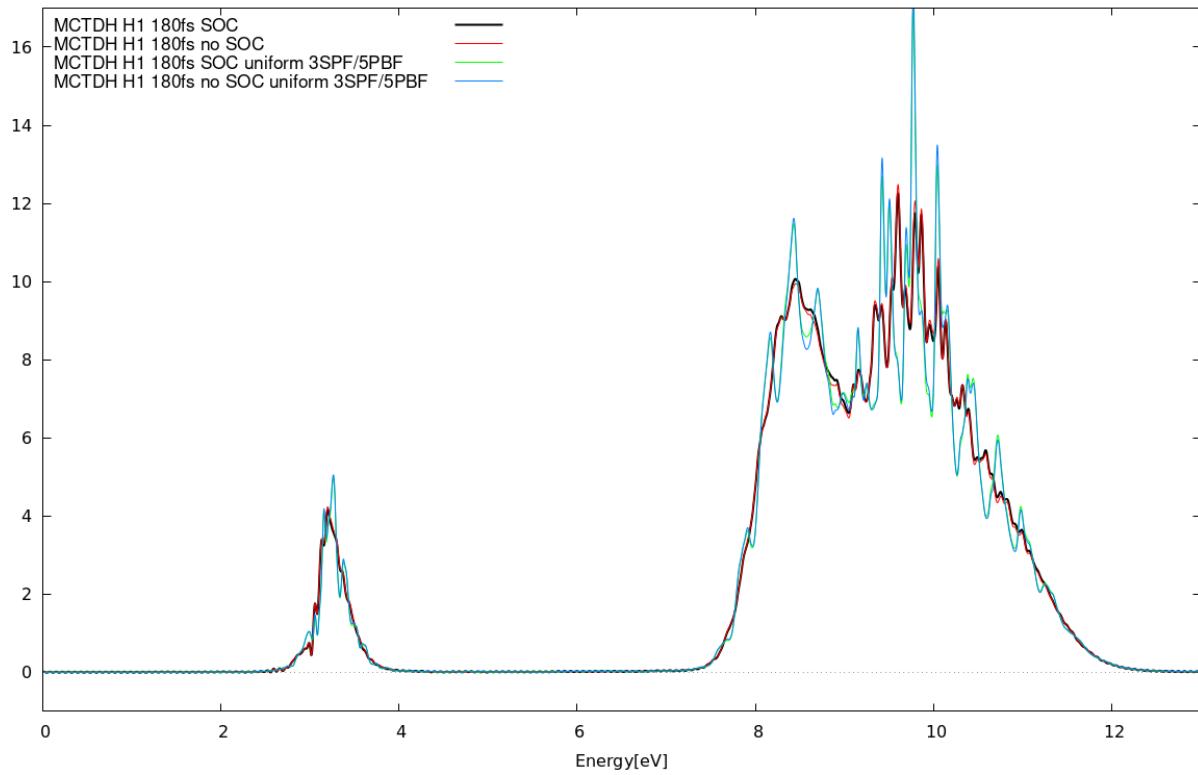
have to make decisions and cleverly choose an array of compatible parameters. Having reservations about a job is needed, oftentimes a double or even triple check is prudent.

FeCO SOC effect not observed probably because it is too subtle and weak.

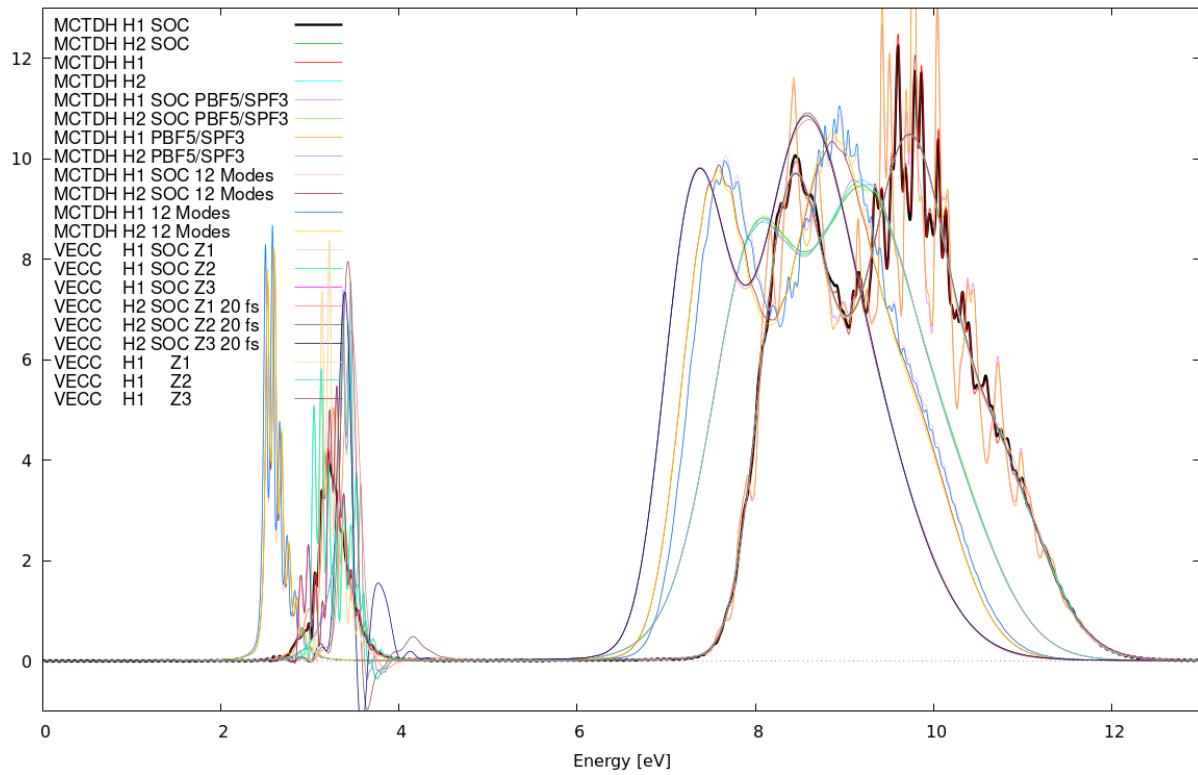




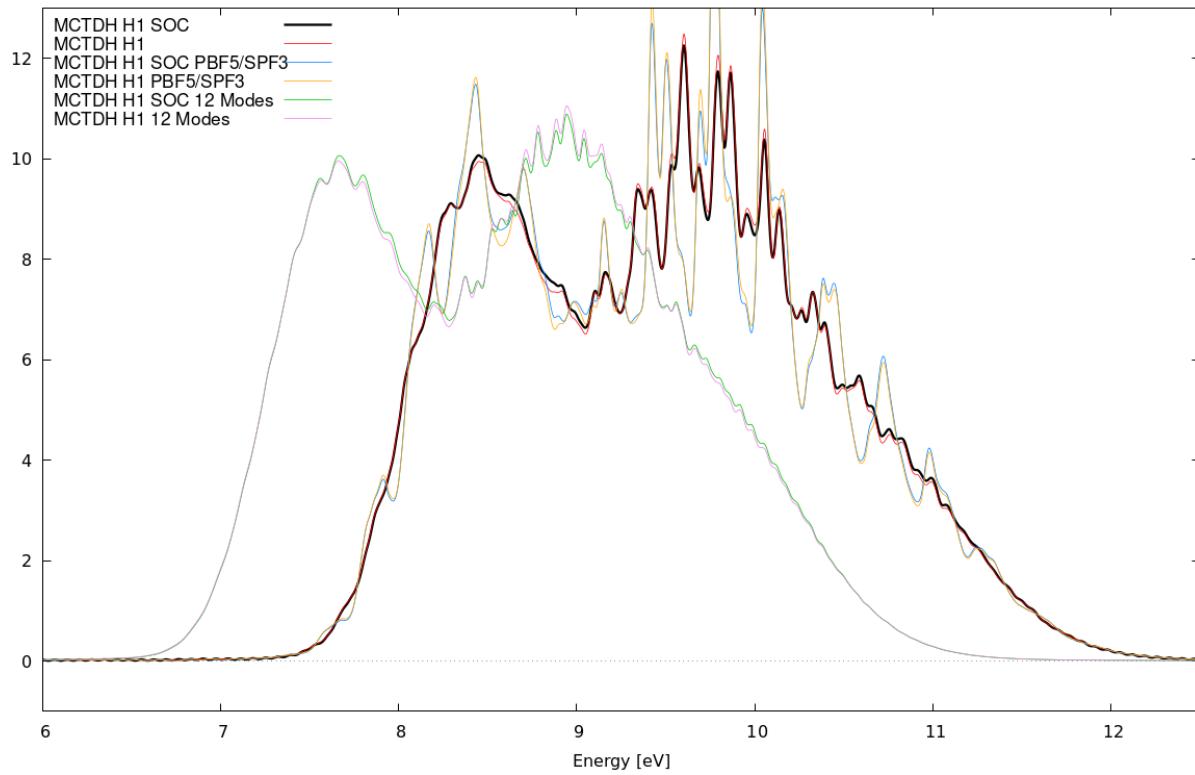
Fe(CO)₅ Vibronic Spectrum



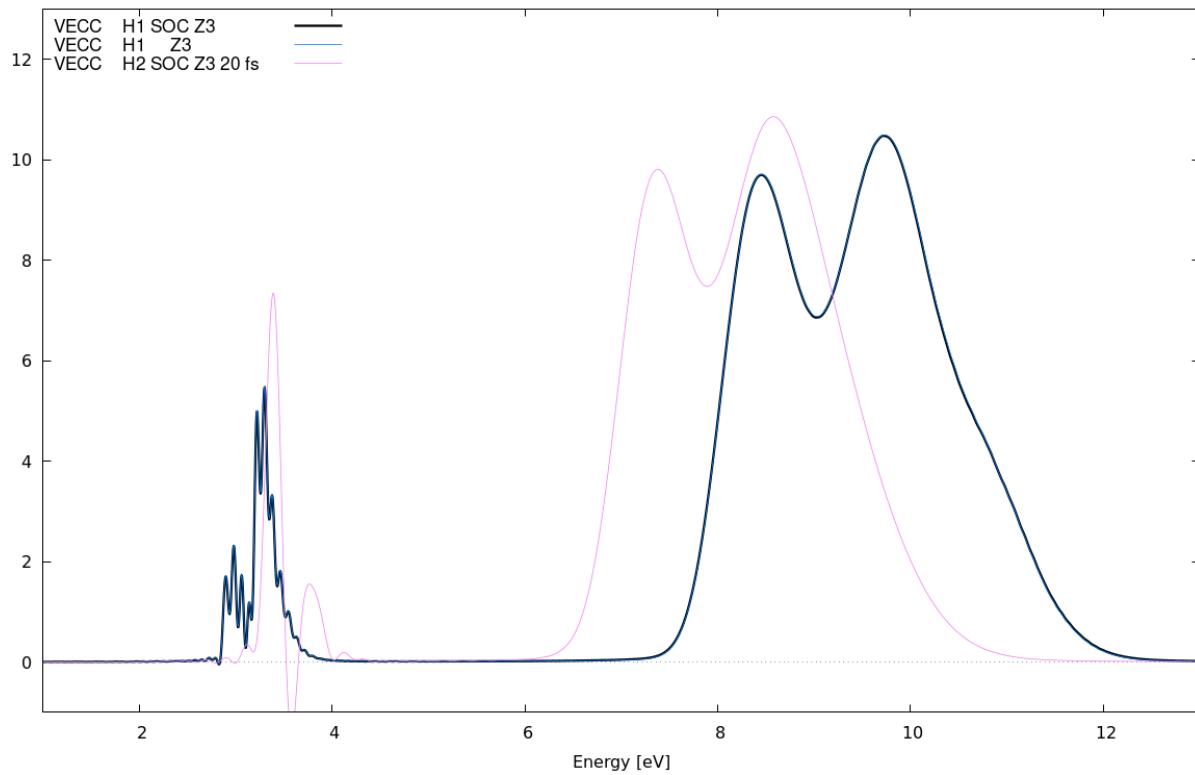
Fe(CO)₅ Vibronic Spectrum Composite - 180 fs ; 40 tau



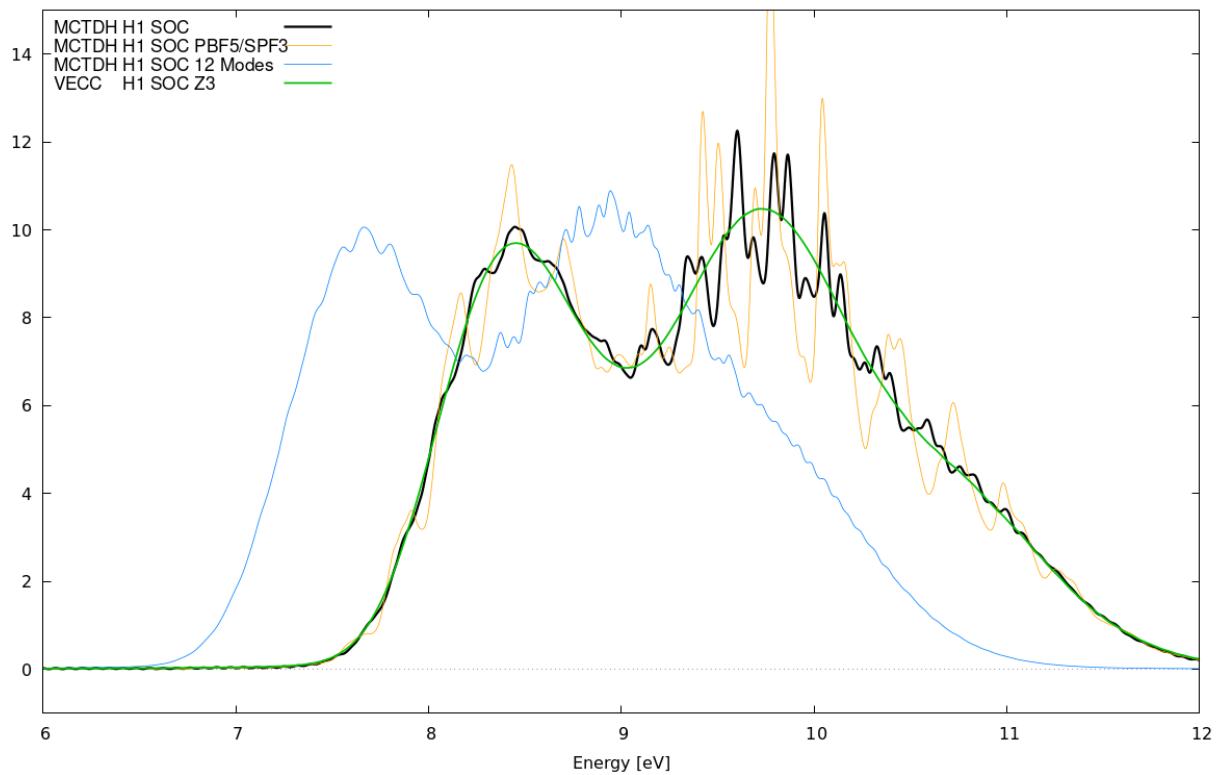
Fe(CO)₅ Vibronic Spectrum - MCTDH SOC Effect Comparison



Fe(CO)₅ Vibronic Spectrum - VECC SOC Effect Comparison

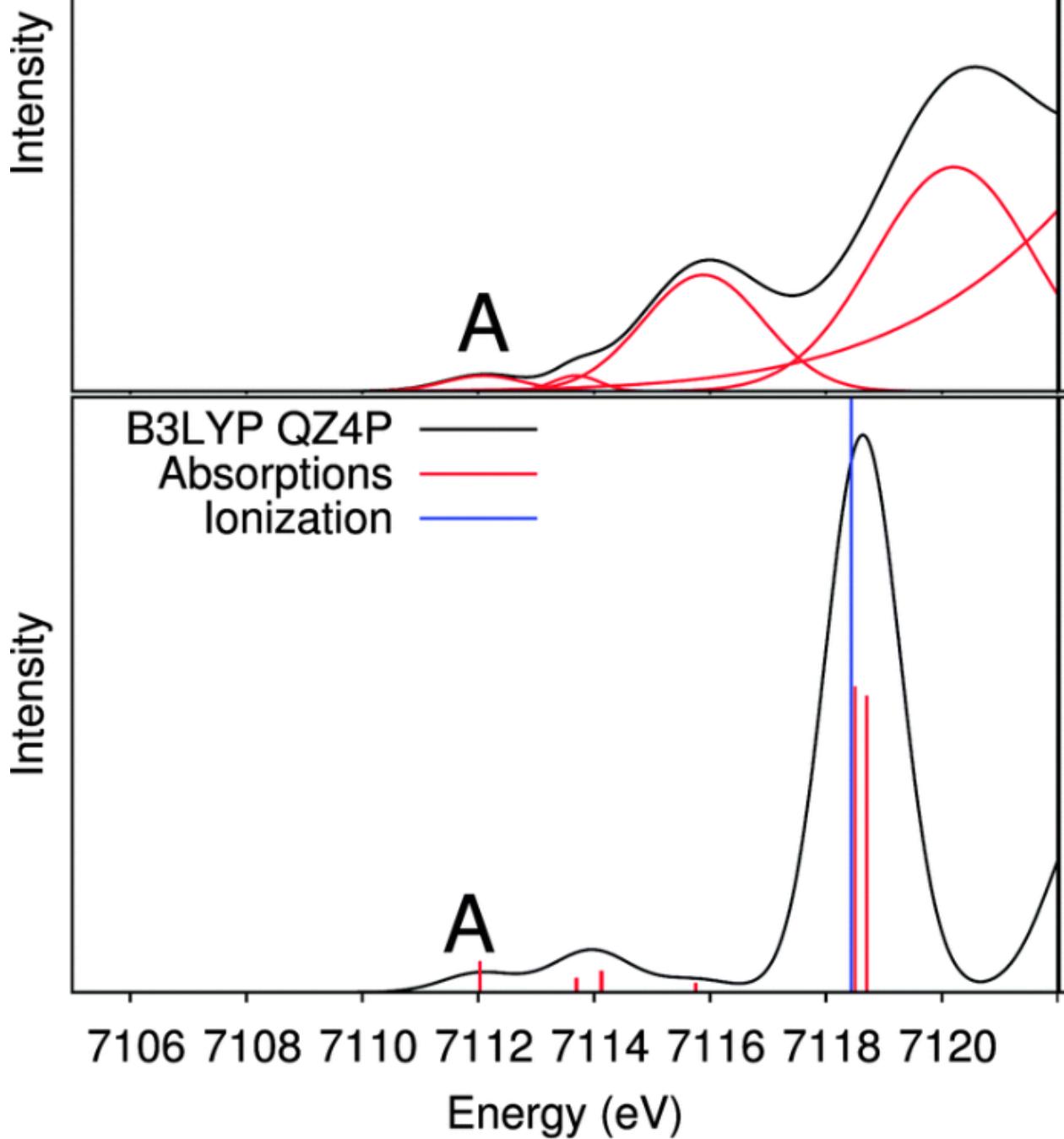


Fe(CO)₅ Vibronic Spectrum - 6 to 12 eV



Lack of tremble and jitters?

Experiment
 $\text{Fe}(\text{CO})_5$



taken from "High-resolution X-ray absorption spectroscopy of iron carbonyl complexes" //

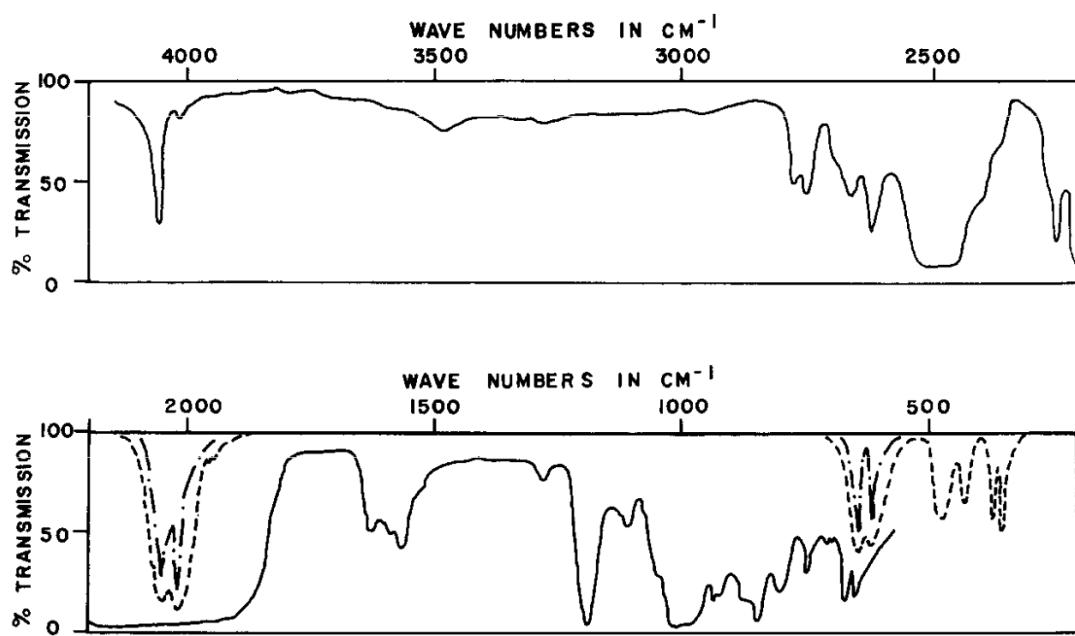
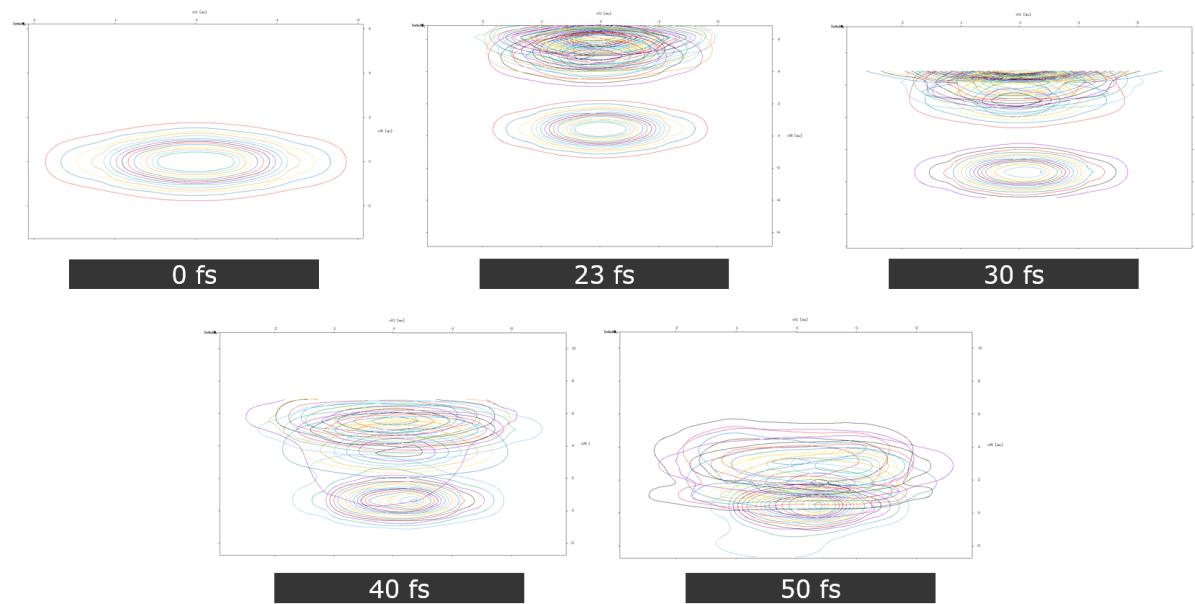


FIG. 1. Infrared absorption spectrum of iron pentacarbonyl. — Liquid at 25°C in 0.5-mm cell. - - Gas at 25°C in 10-cm cell (25-mm partial pressure of iron pentacarbonyl in 735 mm of nitrogen). - · - Gas at 25°C in 10-cm cell (5-mm partial pressure of iron pentacarbonyl in 755 mm of nitrogen).

taken from "Infrared Spectra and Normal Coordinate Analysis of Iron Pentacarbonyl" //



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

No matter how much we tried to increase the primitive basis grid size, the end state is always highly populated, meaning that the potentials are unbound.

AVOID DOING TDH AT ALL COSTS!!! IT EXPLODED MY SCRATCH

```
1 [bjb2chen@narval3 ~]$ diskusage_report
2
3          Description           Space      #
4          of files
5
6          /home (project bjb2chen)    636M/50G
7          12k/500k
8
9          /scratch (project bjb2chen)   19T/20T
10         18k/1000k
11
12         /project (project def-mnooijen) 138G/1000G
13         14k/500k
14
15         /nearline (project def-mnooijen) 98k/1000G
16         4/5000
17
18
19 [bjb2chen@narval3 Apr26_redo_SOC]$ du -ah SOC_CORRECTION/ | sort -hr |
20   head -n 20
21 19T      SOC_CORRECTION/TDH/mctdh/op_FeC027Q_17st_fullmodes
```

Haruyuki Nakano. Hisao Nakamura. See conversation with Neil on June 3rd.

SOC: <https://pubs.aip.org/aip/jcp/article/146/14/144103/195065> Metal trifluorides: <https://www.sciencedirect.com/scientific-topics/metal-trifluorides>

Diabatic states: G. J. Atchity and K. Ruedenberg, Theor. Chem. Acc. 97, 47 (1997).<https://doi.org/10.1007/s001020050001>

<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/cua.10724>

vibronic models; <https://pubs.acs.org/doi/10.1021/acs.jctc.1c00022> (September).

We are using ORIMAS | GMET 1: Student guide that clearly crystallized my knowledge of

GMO-QD1 T. <https://ccr.scc.kyushu-u.ac.jp/~nakano/gmcpf.htm>

Iron pentacarbonyl, haruyuki nakano and mark s gordon https://pubs.rsc.org/en/content/articlepdf/199

This is a very good diagram showing interdependencies: https://en.wikipedia.org/wiki/Complete_active_immutability

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://en.wikipedia.org/wiki/Dung_beetle

```
1 # This is a comment
2 def hello_world():
3     print("Hello, world!")
```

Listing 4.1: Example Python Code

4.7 Various things i need to remember / 12 billion caveats

1. VECC doesn't support off-diagonal E matrix terms; when reading from a .op file it only fills in the diagonal elements of the electronic matrix (E)

```
# save the reference Hamiltonian into the energies array
energies = np.zeros((A, A))
for a in range(A):
    list_of_words = lines[a].split()
    assert list_of_words[0] == f"EH_s{a+1:02}_s{a+1:02}"
    assert list_of_words[-1] == "ev"
    energies[a, a] = list_of_words[2]
```

2. VECC doesn't support more than 1D electron TDM
3. VECC diverges on quadratic after 20fs. You can salvage the run with high autospec damping, but it is fickle.
4. 4 thing
5. 5 thing

Chapter 5

Conclusion

Diabatization is at the heart of current vibronics research. Although it provides a numerical result, it can be applied to generalized situations. Perhaps one day there will be analytical models that can solve vibronics entirely. We cannot make sweeping conclusions about ...

<https://www.sci.uwaterloo.ca/~nooijen/website/research.html>

<https://yorkspace.library.yorku.ca/server/api/core/bitstreams/302f290d-535e-462e-8ca5-1e>
content

<https://pubs.acs.org/doi/10.1021/jp0456990>

<https://www.sci.uwaterloo.ca/~nooijen/website/vibron/VC/Manual.pdf>

https://www.sci.uwaterloo.ca/~nooijen/website/summary_papers.pdf

<https://onlinelibrary.wiley.com/doi/full/10.1002/qua.10724>

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

5.1 MCSCF Pipeline

Inside the repository are other accompanying videos with commentary running through the full fledged protocol from input geometry to spectra. RhF₃ molecule will be used as an example here.

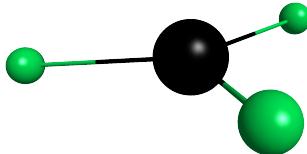


Figure 5.1: RhF₃

5.1.1 Step 1: Geometry optimization + frequency (Hessian)

Readers can follow along on: <https://youtu.be/KJBxNBxqhHA>.

Following standard practice for GAMESS users, input geometry coordinates are first set using the wxMacMolPlt GUI builder tool. Typically the \$data group appears as follows, and GAMESS will duplicate the single fluorine atom given by the appropriate D_{3h} geometry. In the appendices, I have provided a standardized template that can be used as a starting point for all GAMESS calculations in this protocol.

```
1 $data
2 DNH 3
3
4 Rh     45.0      0.0   0.0   0.0
5 F      9.0       2.07  0.0   0.0
6 $end
```

Listing 5.1: Example D_{3h} RhF₃ Input Geometry

Important additional flags that are needed to be modified from the default template:

```
1 $contrl runtyp=optimize                      # 'geometry optimization'
2 $contrl scftyp=rohf    mult=5 icharg=0      # ROHF, multiplicity of 5, neutral
```

```

3 $basis gbasis=SPKrTZP          # Sapporo triple zeta basis set
4 $contrl relwfn=LUT-IOTC       # relativistic effects for heavy
      atom
5 $scf diis=.t                 # turn on DIIS SCF converger

```

Listing 5.2: Additional flags for Step 1.

Table 5.1: Step 1 information.

Input	Output
rungms RhF3_SPK_mp2_rohf_D3h_mult5_diis_gh.inp ncpu ngb nhour	RhF3_*_gh.out

After the input file is successfully setup, one can submit the job to the cluster via the following input command. The resulting output file should locate the important 'EQUILIBRIUM GEOMETRY', and list out the $3N-6 = 6$ vibrational normal modes at the end of calculation.

5.1.2 Step 2: GMCPT

This is the crucial MCSCF (ORMAS) active space setup step.

```

1 $contrl runtyp=energy nosym=1 scftyp=mcscf relwfn=LUT-IOTC
2 $guess norb=168
3 $mcscf cistep=gmcci fullnr=.t. finci=mos
      accuracy=1.d-6 maxit=200 $end
4 $mrmp mrpt=gmcpt rdvecs=.f. $end
5 $gmcpt nmofzc=21 nmodoc=12 nmoact=5 nelact=6 stsym(1)=A
      reftyp=ormas nspace=1 mstart(1)=34 mine(1)=6 maxe(1)=6
6      kstate(1)=1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
7      wstate(1)=1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
8      iwgt=0 thrwgt=-1.0 krot=.f. ksdoe=.f. knospn=.t.
9      thrde=-1.0 edshft=0.02 $end
10
11

```

```

12 $transt $end
13 $data 'EQUILIBRIUM GEOMETRY' coordinates
14 $VEC group of ROHF orbitals

```

Listing 5.3: Additional flags for Step 2.

Table 5.2: Step 2 information.

Command Line	Input	Output
python3 toolkit.py python3 toolkit.py	RhF3_gh.out RhF3_gh.dat	ref_structure vec.dat (ROHF orbs)
rungms .inp ncpu ngb nhour	RhF3_gmcpt.inp	RhF3_gmcpt.out

Note: Talk about GMCPT characteristics

GMCPT is tedious to get right. The recommendation is to consult the students' guide for GMCPT/GMC-QDPT calculations prepared by Nakano and frequently run many experimental jobs. One must ensure that they get the combo of frozen core, doubly occupied, and active orbitals/electrons right. Likewise, if the orbital ranking you input is unphysical, it can potentially result in a poor calculation. If you desire to introduce SOC to your system, attention to add the *\$transt \$end* flag. As it turns out, if you add SOC, and you are running on *nlogn*, the memory mapping fails! This is due to a bug on compiling GMCPT module on nlogn, causing memory corrupted bytes to appear near the termination of the calculation. However, do not be alarmed, as all the meaningful SOC information will be printed out before the memory corruption happens in the output file. If you run the exact same calculations on *ComputeCanada* servers, there will be no problem.

5.1.3 Step 3: DMO Calculation

Discuss semi canonical orbitals to DMO

Here, you will simply use the toolkit script to extract out the Semi-canonical orbitals and \$DMO group from the second step's *.dat* file. Be sure you know the active space's *nmoact* and *mstart* values, as the toolkit script will require these inputs to extract out the correct portion of diabatic molecular orbitals from the full Semi-canonical group. Step 3's input will be identical to Step 2, however the prior orbitals shall be deleted and in its place concatenate the Semi-canonical orbitals along with \$DMO group to the end of the file.

```
1 ~ Step 2 Input ~  
2 $VEC group of Semi-canonical orbitals (replacing old ROHF orbitals)  
3 $DMO group
```

Listing 5.4: Additional flags for Step 3.

Table 5.3: Step 3 information.

Command Line	Input	Output
python3 toolkit.py	RhF3_gmcpt.dat	active_space_orbs (Semi-canonical MOs)
python3 toolkit.py	RhF3_gmcpt.dat	dmo.dat (DMO group)
rungms .inp ncpu ngb nhour	RhF3_dmo.inp	RhF3_dmo.out

5.1.4 Step 4:

Rerun the previous Step 3's process on the new \$DMO outputs to obtain the Semi-canonical MOs and \$DMO group again. After, you will run toolkit on the *.out* file from Step 3, and from it can extract the \$REFDET group automatically. Note that one can actually handcraft a \$REFDET group for better accuracy, using knowledge of dominant Slater determinants and

state configurations by reviewing Step 3's output. Once more, concatenate it to the exact same Step 3 input, and this will be the Step 4 input file.

```

1 ~ Step 3 Input ~
2 $VEC group of Semi-canonical orbitals (fresh from Step 3 outputs)
3 $DMO group (fresh from Step 3 outputs)
4 $REFDET group

```

Listing 5.5: Additional flags for Step 4.

Table 5.4: Step 4 information.

Command Line		Input	Output
python3 toolkit.py		RhF3_dmo.dat	active_space_orbs (Fresh Semi-canonical MOs)
python3 toolkit.py		RhF3_dmo.dat	dmo.dat (DMO group)
rungms .inp ncpu ngb nhour		RhF3_refdet.inp	RhF3_refdet.out

Cleanup: temp.inp

After Step 4 terminates successfully, there should be full vibronic coupling results. If this is the case, then the diabatization template `temp.inp` shall be prepared from Step 4's input. The only difference is that the `$data` group must be deleted, and a new `$data` section appended to the end of the template.

```

1 ~ Step 4 Input ~
2 ~ early $data group deleted ~
3 $data
4 title of diabatization
5 C1

```

Listing 5.6: Additional flags for Step 5.