A *gentle* introduction to MCTDH

Ramón L. Panadés-Barrueta

Computational Chemical Physics Group.
University of Twente.

November 5, 2020

UNIVERSITY
OF TWENTE.

1. Nuclear Quantum Dynamics
   - The Standard method
   - The Time-Dependent Hartree method
2. The Multiconfiguration Time-Dependent Hartree method
   - EOMs
   - Relaxation and block-improved relaxation
3. Potential energy surface representations
   - Tensor decomposition algorithms
4. Code structure and example applications

1. Nuclear Quantum Dynamics
   - The Standard method
   - The Time-Dependent Hartree method
2. The Multiconfiguration Time-Dependent Hartree method
   - EOMs
   - Relaxation and block-improved relaxation
3. Potential energy surface representations
   - Tensor decomposition algorithms
4. Code structure and example applications

Nuclear Quantum Dynamics

What

The subfield of Theoretical Chemistry in which both the **electrons** and the **nuclei** of a molecular system are treated in a **quantum-mechanical** manner.

Nuclear Quantum Dynamics

What

The subfield of Theoretical Chemistry in which both the **electrons** and the **nuclei** of a molecular system are treated in a **quantum-mechanical** manner.

When

- Spectroscopy (e.g. IR transitions)
- Quantum tunneling
- Vibronic coupling
- ZPE determination

Nuclear Quantum Dynamics

Find the numerical solution of the TDSE truncating the Hilbert space to a finite dimension (Galerkin's method):

$$i\hbar\frac{\partial\Psi}{\partial t} = \hat{H}\Psi \tag{1}$$

Given a parametric representation of the WF (Ψ), the optimal solution can be found using the Dirac-Frenkel Variational Principle (DF-VP):

$$\langle\delta\Psi|\hat{H} - i\frac{\partial}{\partial t}|\Psi\rangle = 0 \tag{2}$$

The Standard method

Most direct representation of the WF:

$$\Psi(q_i, \ldots, q_f, t) = \sum_{j_1=1}^{N_1} \cdots \sum_{j_f=1}^{N_f} C_{j_1 \ldots j_f}(t) \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)}(q_\kappa) \tag{3}$$

After plugging this WF into the DF-VP, and performing the corresponding algebra we obtain the following EOMs:

$$\begin{aligned} i\dot{C}_L &= \sum_J \langle \varphi_L | \hat{H} | \varphi_J \rangle \, C_J \\ \boldsymbol{C}(t) &= e^{-i\boldsymbol{H}t} \boldsymbol{C}(0) \end{aligned} \tag{4}$$

where we have introduced the composite indexes $J = (j_1, \ldots, j_f)$.

The Time-Dependent Hartree method

If we now consider time-dependent single-particle functions (SPFs):

$$\Psi(q_1,\ldots,q_f,t) = A(t)\prod_{\kappa=1}^{f}\underbrace{\sum_{\mu=1}^{N_\kappa} c_\mu^{(\kappa,j_\kappa)}(t)\cdot\chi_\mu^{(\kappa)}(q_\kappa)}_{\varphi_\kappa(q_\kappa,t)} \tag{5}$$

and use the DF-VP with arbitrary real constraints $g_\kappa = i\left\langle\varphi_\kappa(t)|\dot{\varphi}_\kappa(t)\right\rangle$, we get the EOMs:

$$\begin{aligned} A(t) &= A(0)\cdot e^{-i\int_0^t E(t')dt'} \\ i\dot{\varphi}_\kappa &= (\mathcal{H}^{(\kappa)} - E)\varphi_\kappa \end{aligned} \tag{6}$$

with $\mathcal{H}^{(\kappa)} = \left\langle\Phi^{(\kappa)}|H|\Phi^{(\kappa)}\right\rangle$.

Limitations of SM and TDH

Standard method

Its application is largely limited due to the **curse of dimensionality**. Only systems up to four atoms (6D) can be addressed in practice.

Time-Dependent Hartree

A better approach, but physically inaccurate. The **nuclear correlation** is harder to retrieve than the electronic correlation due to the nuclei's larger mass. The character of the nuclear WF is inherently **multiconfigurational**.

1 Nuclear Quantum Dynamics
- The Standard method
- The Time-Dependent Hartree method

2 The Multiconfiguration Time-Dependent Hartree method
- EOMs
- Relaxation and block-improved relaxation

3 Potential energy surface representations
- Tensor decomposition algorithms

4 Code structure and example applications

Ansätze comparison

Standard Method (CI)

$$\Psi(q_i, \ldots, q_f, t) = \sum_{j_1=1}^{N_1} \cdots \sum_{j_f=1}^{N_f} C_{j_1 \ldots j_f}(t) \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)}(q_\kappa) \qquad (7)$$

Ansätze comparison

Standard Method (CI)

$$\Psi(q_i, \ldots, q_f, t) = \sum_{j_1=1}^{N_1} \cdots \sum_{j_f=1}^{N_f} C_{j_1 \ldots j_f}(t) \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)}(q_\kappa) \tag{7}$$

Time-Dependent Hartree (HM)

$$\Psi(q_1, \ldots, q_f, t) = A(t) \prod_{\kappa=1}^{f} \varphi_{\kappa}^{(\kappa)}(q_\kappa, t) \tag{8}$$

Ansätze comparison

Standard Method (CI)

$$\Psi(q_i, \ldots, q_f, t) = \sum_{j_1=1}^{N_1} \cdots \sum_{j_f=1}^{N_f} C_{j_1 \ldots j_f}(t) \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)}(q_\kappa) \tag{7}$$

Time-Dependent Hartree (HM)

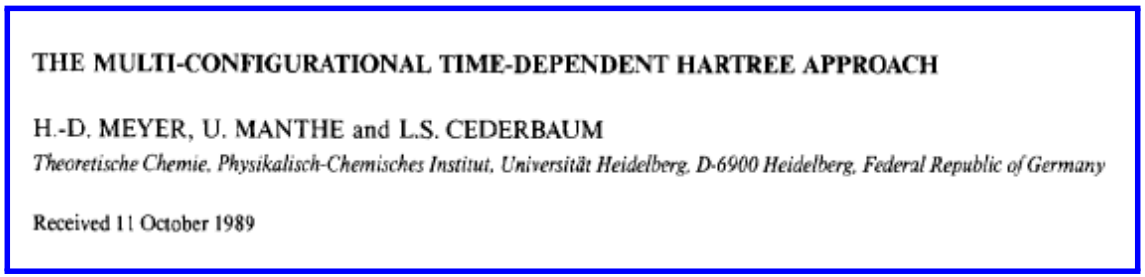
$$\Psi(q_1, \ldots, q_f, t) = A(t) \prod_{\kappa=1}^{f} \varphi_{\kappa}^{(\kappa)}(q_\kappa, t) \tag{8}$$

Multiconfiguration Time-Dependent Hartree (MCSCF)

$$\Psi(q_1, \ldots, q_f, t) = \sum_{j_1=1}^{n_1} \cdots \sum_{j_f=1}^{n_f} A_{j_1, \ldots, j_f}(t) \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)}(q_\kappa, t) \tag{9}$$

MCTDH origins and distribution

MCTDH was originally developed by Meyer and coworkers from the University of Heidelberg, in the early nineties:



THE MULTI-CONFIGURATIONAL TIME-DEPENDENT HARTREE APPROACH

H.-D. MEYER, U. MANTHE and L.S. CEDERBAUM
Theoretische Chemie, Physikalisch-Chemisches Institut, Universität Heidelberg, D-6900 Heidelberg, Federal Republic of Germany

Received 11 October 1989

There are currently three major implementations of the algorithm:

UNIVERSITÄT BIELEFELD

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

UCL

The MCTDH EOMs

The MCTDH ansatz has a very flexible Sum-of-Products (SOP) form:

$$\Psi(q_1,\ldots,q_f,t)=\sum_{j_1=1}^{n_1}\cdots\sum_{j_f=1}^{n_f}A_{j_1,\ldots,j_f}(t)\prod_{\kappa=1}^{f}\varphi_{j_\kappa}^{(\kappa)}(q_\kappa,t) \qquad (10)$$

with time dependent SPFs

$$\varphi_{j_\kappa}^{(\kappa)}(q_\kappa,t)=\sum_{\mu=1}^{N_\kappa}c_\mu^{(\kappa,j_\kappa)}(t)\cdot\chi_\mu^{(\kappa)}(q_\kappa) \qquad (11)$$

The $\chi_\mu^{(\kappa)}(q_\kappa)$ are typically DVR functions.

The MCTDH EOMs

The *ansatz* WF is determined up to a multiplicative constant. To derive the EOMs arbitrary constraint operators ($\hat{g}^{(\kappa)}$) are introduced:

$$i \langle \varphi_l^{(\kappa)} | \dot{\varphi}_j^{(\kappa)} \rangle = \langle \varphi_l^{(\kappa)} | \hat{g}^{(\kappa)} | \varphi_l^{(\kappa)} \rangle \tag{12}$$

Using once again the DF-VP we get (for $\hat{g}^{(\kappa)} \equiv 0$):

$$\begin{aligned} i\dot{A}_J &= \sum_L \langle \Phi_J | \hat{H} | \Phi_L \rangle A_L \\ i\dot{\varphi}_j^{(\kappa)} &= (1 - \hat{P}^{(\kappa)}) \sum_{k,l=1}^{n_\kappa} (\boldsymbol{\rho}^{(\kappa)^{-1}})_{jk} \langle \hat{H} \rangle_{kl}^{(\kappa)} \varphi_l^{(\kappa)} \end{aligned} \tag{13}$$

The MCTDH EOMs

$$i\dot{A}_J = \sum_L \langle \Phi_J | \hat{H} | \Phi_L \rangle A_L$$
$$i\dot{\varphi}_j^{(\kappa)} = (1 - \hat{P}^{(\kappa)}) \sum_{k,l=1}^{n_\kappa} (\rho^{(\kappa)^{-1}})_{jk} \langle \hat{H} \rangle_{kl}^{(\kappa)} \varphi_l^{(\kappa)} \tag{14}$$

$$\Phi_J = \prod_{\kappa=1}^{f} \varphi_{j_\kappa}^{(\kappa)} \tag{15}$$

$$\rho_{kl}^{(\kappa)} = \langle \psi_k^{(\kappa)} | \psi_l^{(\kappa)} \rangle = \sum_{J^\kappa} A_{J_k^\kappa}^* A_{J_l^\kappa} \quad \langle \hat{H} \rangle_{kl}^{(\kappa)} = \langle \psi_k^{(\kappa)} | \hat{H} | \psi_l^{(\kappa)} \rangle \tag{16}$$

$$\hat{P}^{(\kappa)} = \sum_{j=1}^{n_\kappa} |\varphi_j^{(\kappa)}\rangle \langle \varphi_j^{(\kappa)}| \tag{17}$$

MCTDH integration scheme

The MCTDH-EOMs solution is expensive due to the large amount of multidimensional integrals to solve. Since the **mean fields** are not strongly oscillating we can consider (CMF integration):

$$
\begin{aligned}
i\dot{A}_J &= \sum_L \bar{\mathcal{K}}_{JL} A_J \\
i\dot{\varphi}_j^{(1)} &= (1-\hat{P}^{(1)})\{\hat{h}^{(1)}\varphi_j^{(1)} + \sum_{k,l=1}^{n_1} (\boldsymbol{\rho}^{(1)^{-1}})_{jk} \langle \bar{H}_R \rangle_{kl}^{(1)} \varphi_l^{(1)}\} \\
&\vdots \\
i\dot{\varphi}_j^{(f)} &= (1-\hat{P}^{(f)})\{\hat{h}^{(f)}\varphi_j^{(f)} + \sum_{k,l=1}^{n_f} (\boldsymbol{\rho}^{(f)^{-1}})_{jk} \langle \bar{H}_R \rangle_{kl}^{(f)} \varphi_l^{(f)}\}
\end{aligned}
\tag{18}
$$

Mode combination

Nothing prevents us from grouping physical coordinates into logical particles:

$$\begin{aligned} Q_\kappa &\equiv (q_{\kappa,1}, q_{\kappa,1}, \ldots, q_{\kappa,d}) \\ \varphi_j^{(\kappa)}(Q_\kappa, t) &= \varphi_j^{(\kappa)}(q_{\kappa,1}, q_{\kappa,1}, \ldots, q_{\kappa,d}, t) \end{aligned} \tag{19}$$

Under these conditions, the MCTDH *ansatz* will take the form:

$$\begin{aligned} \Psi(Q_1, \ldots, Q_p, t) &= \sum_{j_1=1}^{n_1} \cdots \sum_{j_p=1}^{n_p} A_{j_1,\ldots,j_p}(t) \prod_{\kappa=1}^{p} \varphi_{j_\kappa}^{(\kappa)}(Q_\kappa, t) \\ \varphi_j^{(\kappa)}(Q_\kappa, t) &= \sum_{i_1 \ldots i_d} C_{i_1 \ldots i_d}^{(\kappa,j)}(t) \prod_{\nu=1}^{d} \chi^{(\kappa,\nu)}(q_{\kappa,\nu}) \end{aligned} \tag{20}$$

Multilayer MCTDH (3-layer case)

We can propagate the multidimensional SPFs with MCTDH itself!

$$\Psi(q_1, q_2, q_3, t) = \sum_{j_{12}=1}^{n_{12}} \sum_{j_3=1}^{n_3} A_{j_{12} j_3}(t) \varphi_{j_{12}}^{(12)}(q_1, q_2, t) \varphi_{j_3}^{(3)}(q_3, t) \quad (21)$$

where we have introduced:

$$\varphi_{j_{12}}^{(12)}(q_1, q_2, t) = \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} B_{k_1, k_2}^{(12, j_{12})} \prod_{\mu=1}^{2} \xi_{k_\mu}^{(\mu)}(q_\mu, t) \quad (22)$$

and:

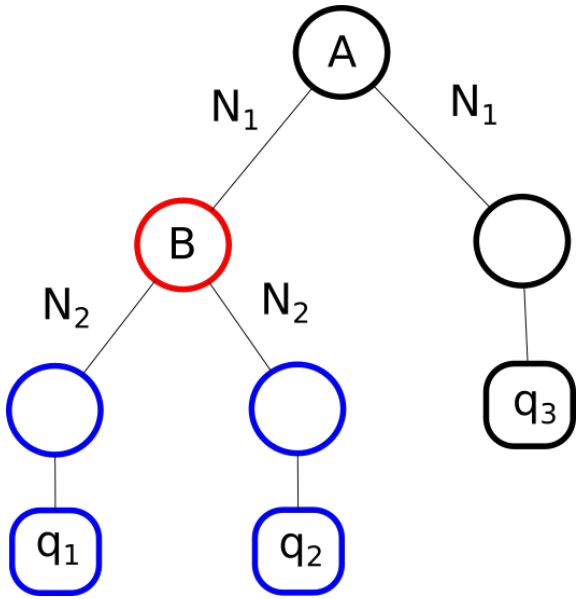
$$\xi_{k_\mu}^{(\mu)}(q_\mu, t) = \sum_{i_\mu=1}^{N_\mu} c_{i_\mu}^{(\mu, k_\mu)} \chi_{i_\mu}^{(\mu)}(q_\mu) \quad (23)$$

Multilayer MCTDH (3-layer case)

$$\Psi(q_1, q_2, q_3, t) = \sum_{j_{12}=1}^{n_{12}} \sum_{j_3=1}^{n_3} A_{j_{12} j_3}(t) \varphi_{j_{12}}^{(12)}(q_1, q_2, t) \varphi_{j_3}^{(3)}(q_3, t)$$

$$\varphi_{j_{12}}^{(12)}(q_1, q_2, t) = \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} B_{k_1, k_2}^{(12, j_{12})} \prod_{\mu=1}^{2} \xi_{k_\mu}^{(\mu)}(q_\mu, t)$$

$$\xi_{k_\mu}^{(\mu)}(q_\mu, t) = \sum_{i_\mu=1}^{N_\mu} c_{i_\mu}^{(\mu, k_\mu)} \chi_{i_\mu}^{(\mu)}(q_\mu)$$



Obtaining vibrational orbitals

MCTDH can be also used to solve the TISE. The GS distribution of the system can be obtained by propagation in negative imaginary time $\tau = -it$:

$$\dot{\Psi} = -\hat{H}\Psi \tag{24}$$

The new algorithm can be derived by applying the time-independent variational principle with Lagrange multipliers:

$$\delta\{\langle\Psi|\hat{H}|\Psi\rangle - E(\sum_{J} A_J^* A_J - 1) - \sum_{\kappa=1}^{f}\sum_{j,l=1}^{n_\kappa} \epsilon_{jl}^{(\kappa)}[\langle\varphi_j^{(\kappa)}|\varphi_l^{(\kappa)}\rangle - \delta_{jl}]\} = 0 \tag{25}$$

Obtaining vibrational orbitals

Taking the variations with respect to the complex conjugate of both the A-vector and the SPFs independently we get:

$$\sum_K H_{JK} A_K = E A_J$$

$$\frac{\partial \varphi_j^{(\kappa)}}{\partial \tau} = -(1 - \hat{P}^{(\kappa)}) \sum_{k,l} (\rho^{(\kappa)^{-1}})_{jk} \langle \hat{H} \rangle_{kl}^{(\kappa)} \varphi_l^{(\kappa)} = 0 \tag{26}$$

The second of these equations implies that we can obtain the updated SPFs simply by relaxation. The A-vector in the first equation can be obtained by Davidson diagonalization algorithm.

1. Nuclear Quantum Dynamics
   - The Standard method
   - The Time-Dependent Hartree method
2. The Multiconfiguration Time-Dependent Hartree method
   - EOMs
   - Relaxation and block-improved relaxation
3. Potential energy surface representations
   - Tensor decomposition algorithms
4. Code structure and example applications

The importance of the SOP form

The multidimensional integrals arising from the MCTDH-EOMs are the bottleneck of the propagation. To address this issue, we impose SOP form to **all quantities**:

$$\hat{O} = \sum_{\alpha=1}^{S} c_\alpha \prod_{\kappa=1}^{f} \hat{o}_\alpha^{(\kappa)}$$
$$\langle \Phi_J | \hat{O} | \Phi_L \rangle = \sum_{\alpha=1}^{S} c_\alpha \prod_{\kappa=1}^{f} \langle \varphi_{j_\kappa}^{(\kappa)} | \hat{o}_\alpha^{(\kappa)} | \varphi_{l_\kappa}^{(\kappa)} \rangle \tag{27}$$

- KEO already in the desired form (`TANA` and `TNUM` software)
- PES might be challenging to transform

Transforming the PES

Usually the PES needs to be refitted (**tensor decomposed**) before using it. The POTFIT algorithm is an elegant way of achieving this in **Tucker form**:

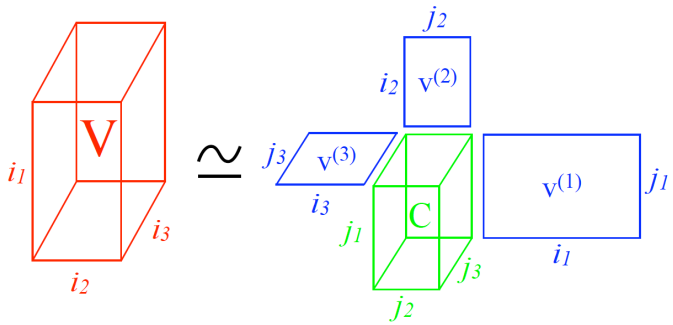
$$V_{i_1,\ldots,i_f} = V(q^{(1)}_{i_1},\ldots,q^{(f)}_{i_f})$$
$$V_{i_1,\ldots,i_f} = \sum_{j_1=1}^{m_1}\cdots\sum_{j_f=1}^{m_f} C_{j_1\cdots j_f}\prod_{\kappa=1}^{f} u^{(\kappa)}_{i_\kappa j_\kappa} \tag{28}$$

with the core tensor coefficients given by the overlap with the potential:

$$C_{j_1\ldots j_f} = \sum_{i_1\ldots i_f} V_{i_1\ldots i_f}\, u^{(1)}_{i_1\ j_1}\cdots u^{(f)}_{i_f\ j_f} \tag{29}$$

The Tucker form

The tucker decomposition of a 3D tensor can be represented graphically as[1]



which can be contrasted with the algebraic and tensor forms:

$$V_{i_1,\ldots,i_f} = \sum_{j_1=1}^{m_1} \cdots \sum_{j_f=1}^{m_f} C_{j_1 \cdots j_f} \prod_{\kappa=1}^{f} u_{i_\kappa j_\kappa}^{(\kappa)} \tag{30}$$

$$\mathcal{V} = \mathcal{C} \times_1 \mathbf{U}_1 \cdots \times_n \mathbf{U}_n$$

[1] Panadés-Barrueta R., and Peláez D. JCP (in review)

Tensor decomposition algorithms

There is a number of tensor decomposition algorithms currently in use (e.g. POTFIT, MGPF, MCPF, MLPF), however, they are all limited by the size of the grids. The **SOP-FBR** method was developed as an alternative to the former:

$$V(q_1,\ldots,q_f) = \sum_{j_1=1}^{m_1}\cdots\sum_{j_f=1}^{m_f} C_{j_1\cdots j_f}\prod_{\kappa=1}^{f}\Phi_{j_\kappa}^{(\kappa)}(q_\kappa)$$
$$\Phi_{j_\kappa}(q_\kappa) = \sum_{\nu_\kappa=1}^{t_k} B_{\nu_\kappa j_\kappa}^{(\kappa)}\, T_{\nu_\kappa}(q_\kappa) \tag{31}$$

This is a fully analytical SOP form, differentiable *ad infinitum*, and that can be directly interfaced with MCTDH.

The POTFIT and HOOI algorithms

Algorithm 1: POTFIT

Result: $\mathcal{C}, \mathrm{U}_1, \ldots, \mathrm{U}_n$

Input: $\mathcal{V}$;

for $k \leftarrow 1$ **to** n **do**

  $\mathrm{U}_k \leftarrow EVD(\mathrm{V}_{(k)}^{\dagger} \cdot \mathrm{V}_{(k)})$

end

$\mathcal{C} \leftarrow \mathcal{V} \times_1 \mathrm{U}_1{}^{-1} \cdots \times_n \mathrm{U}_n{}^{-1}$

Algorithm 2: HOSVD HOOI

Result: $\mathcal{C}, \mathrm{U}_1, \ldots, \mathrm{U}_n$

Input: $\mathcal{V}$;

repeat

  for $k \leftarrow 1$ **to** n **do**

    $\mathcal{Y} \leftarrow \mathcal{V} \times_1 \mathrm{U}_1^{-1} \cdots \times_{k-1} \mathrm{U}_{k-1}^{-1} \times_{k+1} \mathrm{U}_{k+1}^{-1} \cdots \times_n \mathrm{U}_n^{-1}$;

    $\mathrm{U}_k \leftarrow SVD(\mathrm{V}_{(k)})$ $SVD(\mathrm{Y}_{(k)})$

  end

  $\mathcal{C} \leftarrow \mathcal{V} \times_1 \mathrm{U}_1{}^{-1} \cdots \times_n \mathrm{U}_n{}^{-1}$

until $\|\mathcal{V}_{app} - \mathcal{V}\| < \epsilon$;

- $EVD(\mathrm{V}_{(k)}^{\dagger} \cdot \mathrm{V}_{(k)}) \equiv SVD(\mathrm{V}_{(k)})$!
- POTFIT optimizes the factor matrices in a slightly different manner:

$$\tilde{\mathrm{v}}_j^{(\kappa)} = \mathrm{v}_j^{(\kappa)} + \sum_{l=n_\kappa+1}^{N_\kappa} \mu_{jl}^{(\kappa)} \mathrm{v}_l^{(\kappa)}$$

The SOP-FBR algorithm

Algorithm 3: SOP-FBR

Result: x_{opt}

Input: x_{guess} guess parameters, D dimensionality, M number of basis functions, T degree of Chebyshev series, N_g number of geometries, ϵ threshold ;

$k \leftarrow 0$;

$x_0 \leftarrow x_{guess}$;

$G_{ab}, E_{ab} \leftarrow \texttt{geogen}(N_g)$;

Function `target` (B, C)**:**

  $E_{sop} \leftarrow \texttt{sopfbr}(B, C)$;

  $\rho \leftarrow \|E_{ab} - E_{sop}\|_{L_2}$;

return ρ

repeat

  $B, C \leftarrow \texttt{split}(x_k, T \times M)$;

  $B \leftarrow \text{BFGS}(\texttt{target}(B, \bar{C}))$;

  $\rho, C \leftarrow \text{Powell}(\texttt{target}(\bar{B}, C))$;

  $x_{k+1} \leftarrow \texttt{concatenate}(B, C)$;

  $k \leftarrow k + 1$;

until $\rho < \epsilon \vee k < N$;

$x_{opt} \leftarrow x_k$

1 Nuclear Quantum Dynamics
- The Standard method
- The Time-Dependent Hartree method

2 The Multiconfiguration Time-Dependent Hartree method
- EOMs
- Relaxation and block-improved relaxation

3 Potential energy surface representations
- Tensor decomposition algorithms

4 Code structure and example applications

The Heidelberg implementation of MCTDH

The actual implementation is written mainly in *f*(or)tr[an] , with some small C and Python contributions. The program has a modular structure with a very intuitive and consistent input syntax. Some sections of a POTFIT input file:

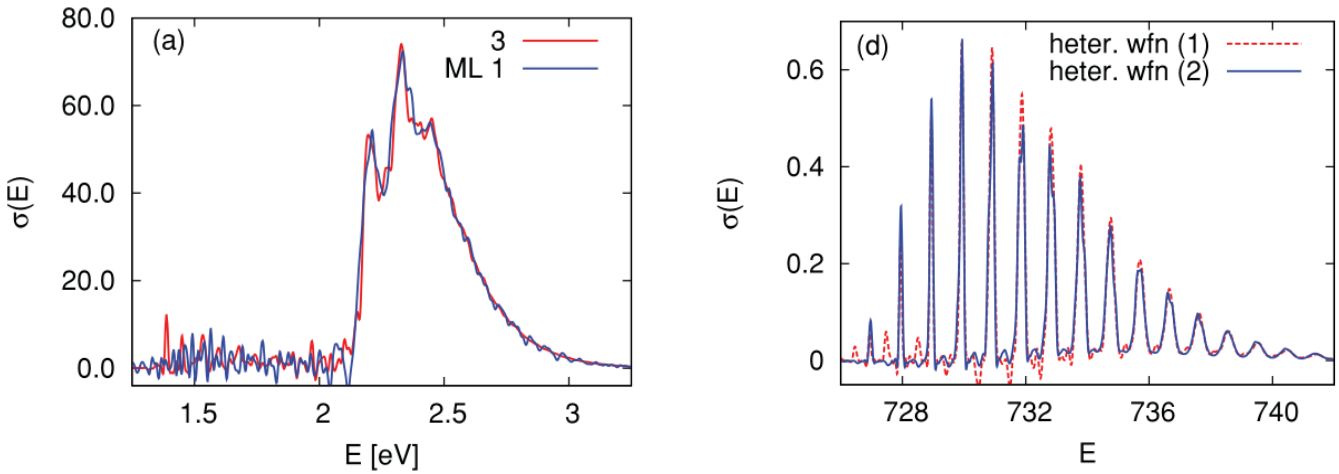
```
RUN-SECTION                                    # System declaration
    name = h2o.pfit                            # The file extension only
end-run-section                                # suggests a POTFIT calculation

OPERATOR-SECTION
    pes = pjt2{binding}
    vcut < 0.5                                 # Define Hamiltonian
end-operator-section

PRIMITIVE-BASIS-SECTION
    r1     sin     34   1.0    3.475
    r2     sin     34   1.0    3.475           # Define coordinates
    theta  Leg/R   50    0     all   0.5  3.2   # and basis functions
end-primitive-basis-section
```

Applications

Some interesting applications that showcase the power of MCTDH are [2]:



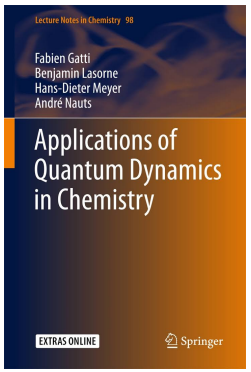
(a) 24D (b) 1458D

Figure: Power spectrum obtained with ML-MCTDH for (a) pyrazine (b) the Henon-Heiles Hamiltonian

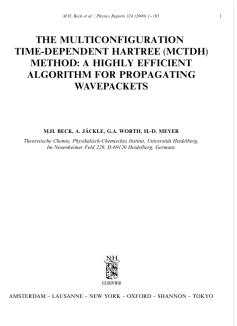
[2] Vendrell, O., and Meyer, H.D., JCP 134.4 (2011): 044135.

Bibliography

Gatti, F., *et al.* Applications of quantum dynamics in chemistry. Vol. 98. Springer, 2017.



Beck, M.H., *et al.* The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. Physics reports 324.1 (2000): 1–105.



Thanks for your attention!

Questions?