# The Quantics Package: Using and Developing

Graham Worth

Dept. of Chemistry, University College London, U.K.

# Install Quantics

Download from Gitlab (ask for access) to `$QUANTICS_DIR`

```
cd $QUANTICS_DIR/install
./install_quantics
```

Accept all questions - i.e. answer y. Or use

```
./install_quantics -y
```

```
source ~/.bashrc
```

Ready to Roll!
Documentation is found in `$QUANTICS_DIR/doc/index.html`

Can switch between versions with `minstall VER_NAME`.

See which you are using with `menv`.

# Steps for quantum dynamics simulations

1. Get the Hamiltonian
   - Specify the system coordinates
   - Work out the Kinetic energy operator
   - Work out the potential energy operator

# Steps for quantum dynamics simulations

1. Get the Hamiltonian
   - Specify the system coordinates
   - Work out the Kinetic energy operator
   - Work out the potential energy operator
2. Define a *primitive basis set* to represent the Hamiltonian and wavepacket

# Steps for quantum dynamics simulations

1. Get the Hamiltonian
   - Specify the system coordinates
   - Work out the Kinetic energy operator
   - Work out the potential energy operator
2. Define a *primitive basis set* to represent the Hamiltonian and wavepacket
3. Define an initial wavepacket

# Steps for quantum dynamics simulations

1. Get the Hamiltonian
    - Specify the system coordinates
    - Work out the Kinetic energy operator
    - Work out the potential energy operator
2. Define a *primitive basis set* to represent the Hamiltonian and wavepacket
3. Define an initial wavepacket
4. Select a method to propagate the wavepacket (integrate the TDSE)

# Steps for quantum dynamics simulations

1. Get the Hamiltonian
   - Specify the system coordinates
   - Work out the Kinetic energy operator
   - Work out the potential energy operator
2. Define a *primitive basis set* to represent the Hamiltonian and wavepacket
3. Define an initial wavepacket
4. Select a method to propagate the wavepacket (integrate the TDSE)
5. Analyse wavepacket to obtain (time-evolution) data.

# Methods Available

**Table 1**
Main methods available in the QUANTICS Program.

| Method | Equation solved | Basis sets |
|---|---|---|
| Standard WP | TDSE | DVRs or FFT |
| MCTDH | TDSE | DVRs or FFT |
| ML-MCTDH | TDSE | DVRs or FFT |
| GMCTDH | TDSE | DVRs or FFT or GWPs |
| ML-GMCTDH | TDSE | DVRs or FFT or GWPs |
| $\rho$MCTDH(I) | LvN | DVRs or FFT |
| $\rho$MCTDH(II) | LvN | DVRs or FFT |
| ML-$\rho$MCTDH(I) | LvN | DVRs or FFT |
| ML-$\rho$MCTDH(II) | LvN | DVRs or FFT |
| $\rho$GMCTDH(II) | LvN | DVRs or FFT or GWPs |
| ML-$\rho$GMCTDH(II) | LvN | DVRs or FFT or GWPs |
| Standard WP | TISE | DVRs or FFT |
| vMCG | TDSE | GWPs |
| clMCG | TDSE | GWPs |
| iMCG | TDSE | GWPs |
| DD-vMCG | TDSE | GWPs |
| TSH | TDSE | trajectories |
| DD-TSH | TDSE | trajectories |

| Integration schemes | Method |
|---|---|
| Chebyshev | Standard WP |
| Second Order Differencing | Standard WP |
| Split Operator | Standard WP |
| Constant Mean Field (CMF) | MCTDH (not ML-) |
| Adams–Bashforth–Moulton | all MCTDH and MCG |
| Burlisch–Stoer | all MCTDH and MCG |
| Runge–Kutta | all MCTDH and MCG |

DVRs:
Harmonic Oscillator, Sine, Cosine, Exponential, Legendre, Laguerre

2D spherical harmonic basis sets:
Spherical Harmonic FBR, Extended Legendre, 2D Legendre

# Butatriene photoelectron spectrum

18 modes $\mathrm{D_{2h}}$

$X^2\mathrm{B_{2g}}$ ; $A^2\mathrm{B_{2u}}$ $\qquad \mathrm{B_{2g}} \times \mathrm{B_{2u}} = \mathrm{A_u}$

$$\mathbf{H} = \sum_i \frac{\omega_i}{2}\left(-\frac{\partial^2}{\partial Q_i^2} + Q_i^2\right)\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} \epsilon_1 & 0 \\ 0 & \epsilon_2 \end{pmatrix}$$

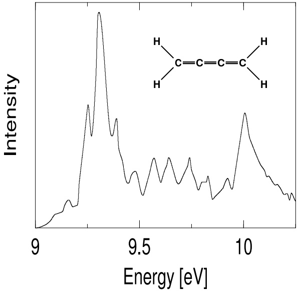
$$+ \sum_{i \in \mathrm{A_g}}^{4} \begin{pmatrix} \kappa_i^{(1)} & 0 \\ 0 & \kappa_i^{(2)} \end{pmatrix} Q_i + \begin{pmatrix} 0 & \lambda \\ \lambda & 0 \end{pmatrix} Q_{\mathrm{A_u}}$$

$$+ \dots$$

$$\langle \psi_\alpha | \frac{\partial H}{\partial Q_i} | \psi_\beta \rangle$$

linear: 5 modes, 16 parameters
bilinear: 18 modes, 79 parameters

$$I(\omega) \sim \int_{-\infty}^{\infty} dt \langle \Psi(0) \mid \Psi(t) \rangle e^{i\omega t}$$



# KE for a torsion

Transform the “torsional” normal mode to an angle

$$\frac{\omega_5}{2}\frac{\partial^2}{\partial Q_5^2} \longrightarrow \frac{1}{2I}\frac{\partial^2}{\partial \alpha^2}$$

$$\frac{\omega_5}{2}Q_5^2 \longrightarrow A\sin^2(2\alpha) + B\sin^2(4\alpha) + C\sin^2(6\alpha) + D\sin^2(8\alpha)$$

$$\lambda_5 Q_5 \longrightarrow \lambda_5 r_0\sqrt{\omega_5 m_H}\sin(2\alpha)$$

# Different Representations

5D: $Q_5, Q_8, Q_{12}, Q_{14}, Q_{15}$

1. MCTDH multi-set
   $\Psi(\mathbf{Q},t) = \sum_s \sum_{i_1 i_2 i_3} A^{(s)}_{i_1 i_2 i_3} \varphi^{(s)}_{i_1}(Q_5)\varphi^{(s)}_{i_2}(Q_{14},Q_8)\varphi^{(s)}_{i_3}(Q_{12},Q_{15})$
   $\varphi^{(s)}_{i_1}(Q_5,t) = \sum_{a_1} c_{a_1,i_1}(t)\chi_{a_1}(Q_5)$ etc.
2. MCTDH single-set
   $\Psi(\mathbf{Q},t) = \sum_{i_1 i_2 i_3 s} A_{i_1 i_2 i_3,s} \varphi^{(s)}_{i_1}(Q_5)\varphi^{(s)}_{i_2}(Q_{14},Q_8)\varphi^{(s)}_{i_3}(Q_{12},Q_{15})|s>$
3. ML-MCTDH $\Psi(\mathbf{Q},t) = \sum_s A_s \varphi^1_s(Q_5,Q_8,Q_{12},Q_{14},Q_{15})|s>$
   $\varphi^1_s(Q_5,Q_8,Q_{12},Q_{14},Q_{15}) = \sum_{i_1 i_2} A^2_{s,i_1 i_2}\varphi^2_{s,i_1}(Q_5,Q_{14})\varphi^2_{s,i_2}(Q_8,Q_{12},Q_{15})$
   $\varphi^2_{si_1}(Q_5,Q_{14}) = \sum_{j_1 j_2} A^3_{i_1,j_1 j_2}\varphi^3_{i_1,j_1}(Q_5)\varphi^3_{i_1,j_2}(Q_{14})$ etc.
4. vMCG (G-MCTDH)
   $\Psi(\mathbf{Q},t) = \sum_{i_1 i_2 s} A_{i_1 i_2,s} G^{(1)}_{i_1}(Q_5,Q_{14}) G^{(2)}_{i_2}(Q_8,Q_{12}Q_{15})|s>$
5. TSH $\{T_i(\mathbf{Q}_i,\mathbf{P}_i)\}$ ; $\rho(\mathbf{Q},\mathbf{P})\, d\mathbf{Q}d\mathbf{P} = \sum_j T_j\, d\mathbf{Q}d\mathbf{P}$
6. DD-vMCG
7. DD-TSH

# Recipe for Direct Dynamics

DD-vMCG

$$\Psi(\mathbf{x}, t) = \sum_J A_J g_J(\mathbf{x}, t)$$

$$\begin{aligned}
\dot{q}_j &= \frac{p_j}{m} + \frac{1}{2\alpha_j} \mathrm{Im} \sum_m C_{jm}^{-1} \tilde{Y}_m \\
\dot{p}_j &= -V_j' + \mathrm{Re} \sum_m C_{jm}^{-1} \tilde{Y}_m
\end{aligned}$$

- Gradients and Hessians directly from quantum chemistry. (Hessian update).
- Store results in a database (energy, gradient, Hessian)
- Shepard Interpolate between points
- Diabatise using *Propagation Diabatisation*

# Recipe for Direct Dynamics

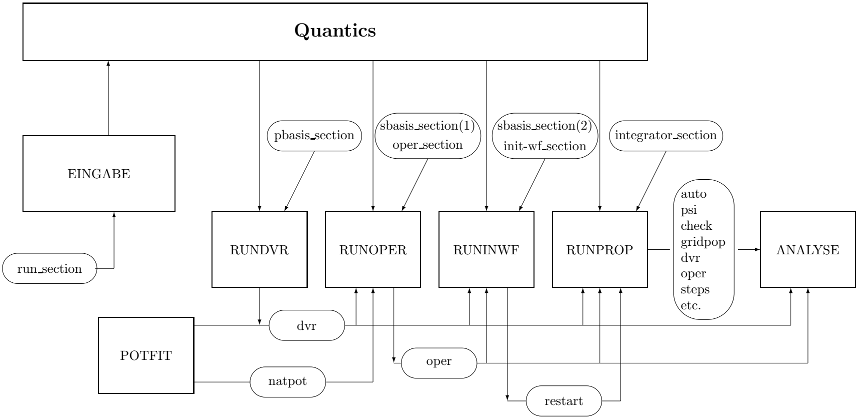
DD-TSH

Swarm of trajectories sampled from initial distribution

$$\begin{aligned} \dot{q}_j &= \frac{p_j}{m} \\ \dot{p}_j &= -V'_j \end{aligned}$$

- Gradients and Hessians directly from quantum chemistry. (Hessian update).
- Store results in a database (energy, gradient, Hessian)
- Shepard Interpolate between points
- Diabatise using *Propagation Diabatisation*

# Code Structure



# DB Structure

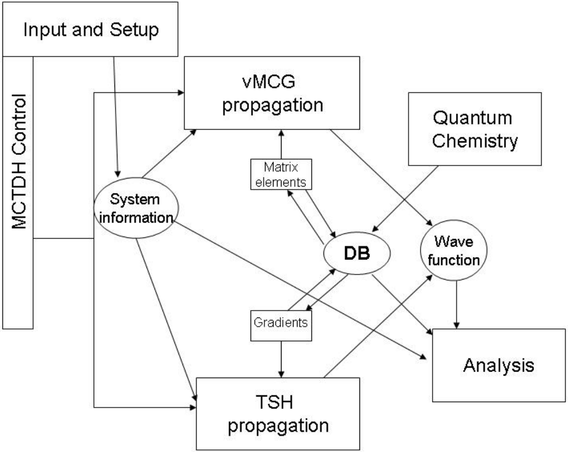



Fig. 1. Information flow of MCTDH propagation showing database (DB) as central element linking alternative propagation schemes (vMCG or TSH) to the Quantum Chemistry programs and the Analysis programs

# GIT Usage

- git clone https://gitlab.com/quantics/quantics.git QUANTICS_DIR
- cd QUANTICS_DIR
- git branch newbranch
- git checkout newbranch

After making changes

- git add .
- git commit

You will be asked to give a log message on committing changes.

- git push

On gitlab make a merge request

Document, add tests, .....