Excited State Molecular Dynamics Simulations with pyUNIxMD

Seung Kyu Min

Department of Chemistry,
Ulsan National Institute of Science and Technology (UNIST),
S. Korea

Important references for pyUNIxMD

- **The first paper about pyUNIxMD – structures, algorithms, and features**

Received: 7 May 2021 | Revised: 10 June 2021 | Accepted: 16 June 2021
DOI: 10.1002/jcc.26711

SOFTWARE NOTE

Journal of COMPUTATIONAL CHEMISTRY WILEY

PyUNIxMD: A Python-based excited state molecular dynamics package

In Seong Lee | Jong-Kwon Ha | Daeho Han | Tae In Kim | Sung Wook Moon | Seung Kyu Min

Department of Chemistry, Ulsan National Institute of Science and Technology (UNIST), Ulsan, South Korea

Correspondence
Seung Kyu Min, Department of Chemistry, Ulsan National Institute of Science and Technology (UNIST), 50 UNIST-gil, Ulju-gun, Ulsan 44919, South Korea.

Abstract
Theoretical/computational description of excited state molecular dynamics is nowadays a crucial tool for understanding light-matter interactions in many materials. Here we present an open-source Python-based nonadiabatic molecular dynamics program package, namely PyUNIxMD, to deal with mixed quantum-classical dynamics for correlated electron-nuclear propagation. The PyUNIxMD provides many interfaces for

J. Comp. Chem., **42**, 1755 (2021)

- **The second paper about pyUNIxMD – interfacing with arbitrary programs such as machine learned potentials**

Topics in Current Chemistry (2022) 380:8
https://doi.org/10.1007/s41061-021-00361-7

REVIEW

Check for updates

Coupled- and Independent-Trajectory Approaches Based on the Exact Factorization Using the PyUNIxMD Package

Tae In Kim[1] · Jong-Kwon Ha[1] · Seung Kyu Min[1]

Received: 27 August 2021 / Accepted: 11 November 2021 / Published online: 27 January 2022

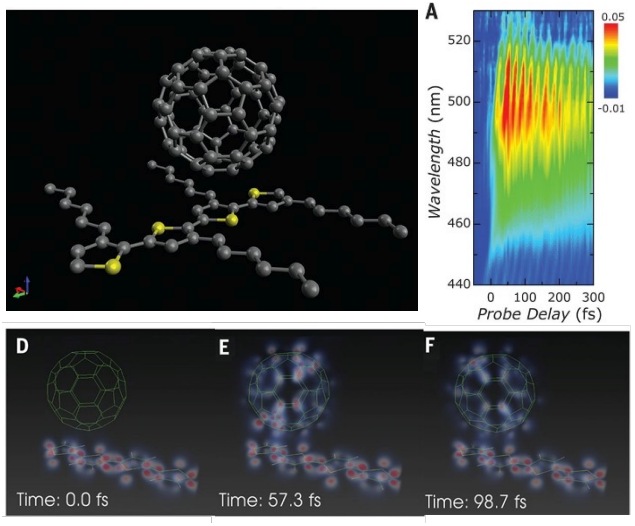
Abstract
We present mixed quantum-classical approaches based on the exact factorization framework. The electron–nuclear correlation term in the exact factorization enables us to deal with quantum coherences by accounting for electronic and nuclear nona-

Top. Cur. Chem., **380**, 8 (2022)

Outline

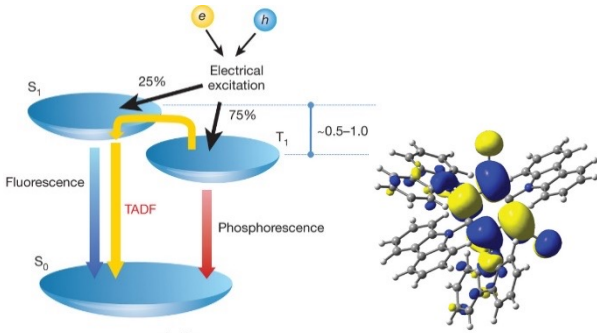
- Introduction – Excited state phenomena
- Electron-nuclear correlation from exact factorization
- Mixed quantum-classical approaches based on exact factorization
- PyUNIxMD program – Structures, Flows, etc.
- Incoming features
- Summary

What we want to simulate…



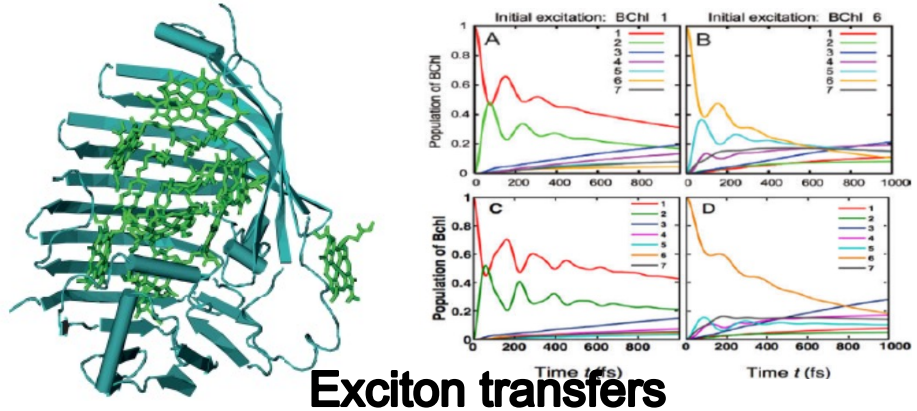
Science, 344, 1001 (2014)

Electron transfers



Nature, 492, 234 (2012)

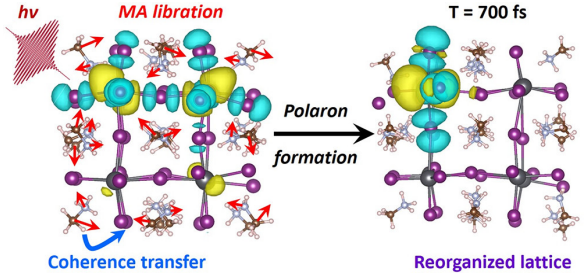
Thermally activated delayed fluorescence



Exciton transfers

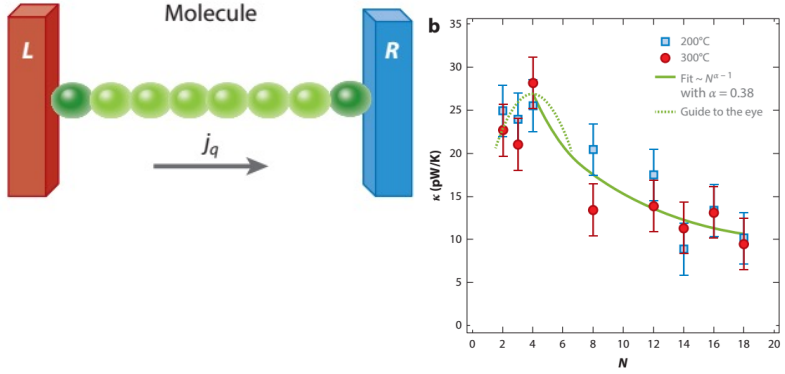
UNIxMD

pyUNIxMD



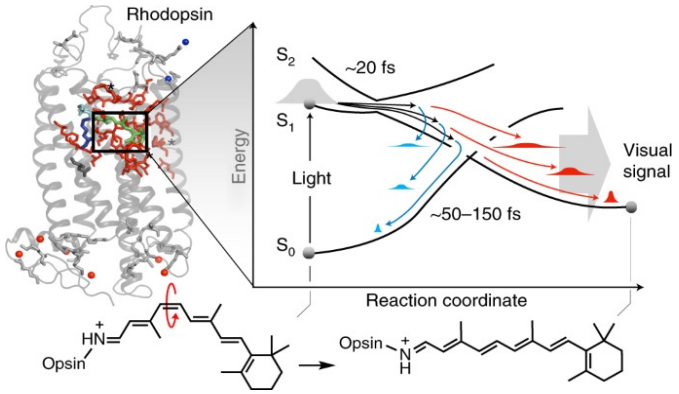
J. Am. Chem. Soc., 142, 16569 (2020)

Polaron formation



Phys. Rev. Lett., 113, 060801 (2014)

Nanoscale “polaron” transport phenomena



Nat. Chem., 14, 441 (2022)

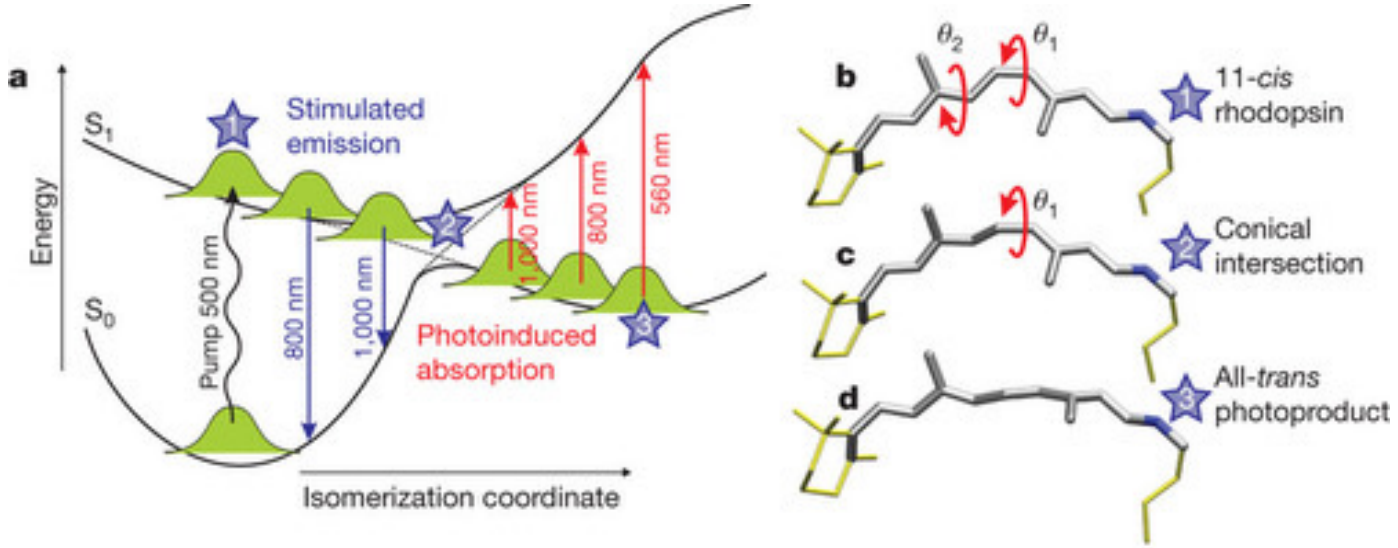
Vision process

Features of pyUNIxMD

- Electron-nuclear correlation based on the exact factorization
  - A very natural handling of electron-nuclear correlation beyond classical nonadiabatic couplings
  - It can provide a correct description of decoherence
- Modularized & easy-to-use (with a simple python knowledge)
  - Can be coupled to any python-based program
- Various mixed quantum-classical (MQC) dynamics are implemented

Dynamics with electronic excited states

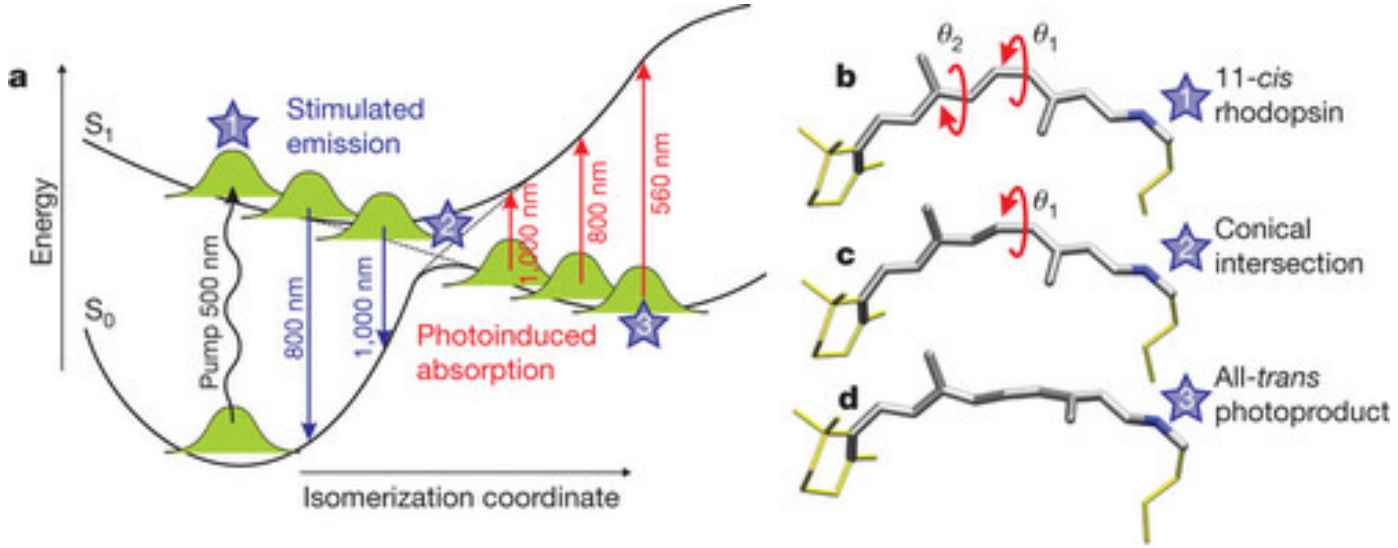
- **Key ingredients**



- Multiple Born-Oppenheimer potential energy surfaces (PESs)
- Nonadiabatic electronic transitions (internal conversion, …)
- Quantum nuclear dynamics (nuclear wave packet splitting)

Dynamics with electronic excited states

- Key ingredients



- Multiple Born-Oppenheimer potential energy surfaces (PESs)

  QC, ML, …

- Nonadiabatic electronic transitions (internal conversion, …)

- Quantum nuclear dynamics (nuclear wave packet splitting) CTMQC, DISH-XF, PyUNIxMD

Beyond classical nuclei: exact factorization

- **Molecular wave function as a single product**

$$\Psi(\underline{\underline{\mathbf{r}}},\underline{\underline{\mathbf{R}}},t) = \chi(\underline{\underline{\mathbf{R}}},t)\Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\underline{\mathbf{r}}},t),$$

the solution of the full molecular TDSE

where $\int d\underline{\underline{\mathbf{r}}}|\Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\underline{\mathbf{r}}},t)|^2 = 1 \forall \underline{\underline{\mathbf{R}}}$

- **Nuclear wave function χ → exact time-dependent nuclear density**

- **Nuclear EOM**

$$i\hbar\partial_t\chi(\underline{\underline{\mathbf{R}}},t) = \left(\sum_{\nu=1}^{N_n}\frac{\left[-i\hbar\nabla_\nu + \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t)\right]^2}{2M_\nu} + \epsilon(\underline{\underline{\mathbf{R}}},t)\right)\chi(\underline{\underline{\mathbf{R}}},t)$$

- **Electronic wave function Φ → a time-dep. (TD) PES ε, a TD vector potential *A***

$$\epsilon(\underline{\underline{\mathbf{R}}},t) = \left\langle \Phi_{\underline{\underline{\mathbf{R}}}}(t)\right| \hat{H}_{BO} + \hat{U}_{en}^{coup} - i\hbar\partial_t \left|\Phi_{\underline{\underline{\mathbf{R}}}}(t)\right\rangle_{\underline{\underline{\mathbf{r}}}} \quad \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t) = \left\langle \Phi_{\underline{\underline{\mathbf{R}}}}(t)\right| - i\hbar\nabla_\nu\Phi_{\underline{\underline{\mathbf{R}}}}(t)\right\rangle_{\underline{\underline{\mathbf{r}}}}$$

Phys. Rev. Lett., **105**, 123002 (2010), *J. Chem. Phys.*, **137**, 22A530 (2012)

Beyond classical nuclei: exact factorization

- **Electronic equation of motion**

$$i\hbar\partial_t \Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\underline{\mathbf{r}}},t) = \left(\hat{H}_{BO}(\underline{\underline{\mathbf{r}}},\underline{\underline{\mathbf{R}}}) + \hat{U}_{en}^{coup}[\Phi_{\underline{\underline{\mathbf{R}}}},\chi] - \epsilon(\underline{\underline{\mathbf{R}}},t)\right)\Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\underline{\mathbf{r}}},t)$$

where the electron-nuclear correlation operator is

$$\hat{U}_{en}^{coup}[\Phi_{\underline{\underline{\mathbf{R}}}},\chi] = \sum_{\nu=1}^{N_n}\frac{1}{M_\nu}\left[\frac{\left[-i\hbar\nabla_\nu - \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t)\right]^2}{2} + \left(\frac{-i\hbar\nabla_\nu\chi}{\chi} + \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t)\right)\cdot\left(-i\hbar\nabla_\nu - \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t)\right)\right]$$

- **Analysis of electron-nuclear correlation operator**

$$\sum_{\nu=1}^{N_n}\frac{1}{M_\nu}\left[\frac{\left[-i\hbar\nabla_\nu - \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t)\right]^2}{2}\right]$$

generate a potential corresponding to diagonal BO correction (DBOC)

$$\frac{-i\hbar\nabla_\nu\chi}{\chi} + \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t) = \mathbf{P}_{cl} - \frac{i\hbar\nabla_\nu|\chi|}{|\chi|}$$

classical momentum provides Ehrenfest equation with trajectories

quantum momentum provides additional correlation toward quantum (de)coherence

Phys. Rev. Lett., **105**, 123002 (2010), *J. Chem. Phys.*, **137**, 22A530 (2012)

Early members of the exact factorization

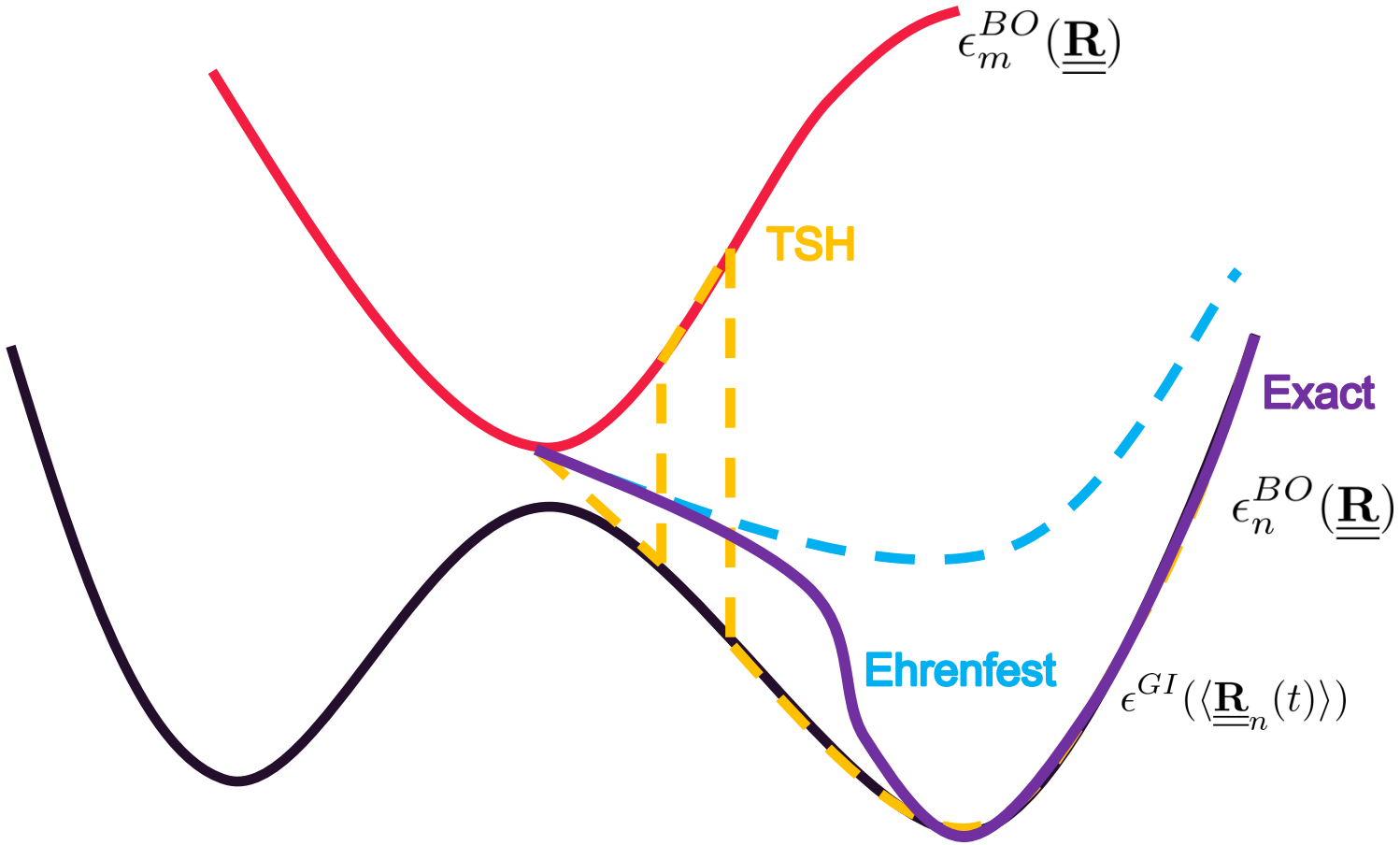
Neepa T. Maitra (Rutgers U., Newark)

Long time ago... @ CECAM Workshop in Lausanne

Phys. Rev. Lett., **105**, 123002 (2010), *J. Chem. Phys.*, **137**, 22A530 (2012)

Classical Nuclei: Effective potential energy surfaces

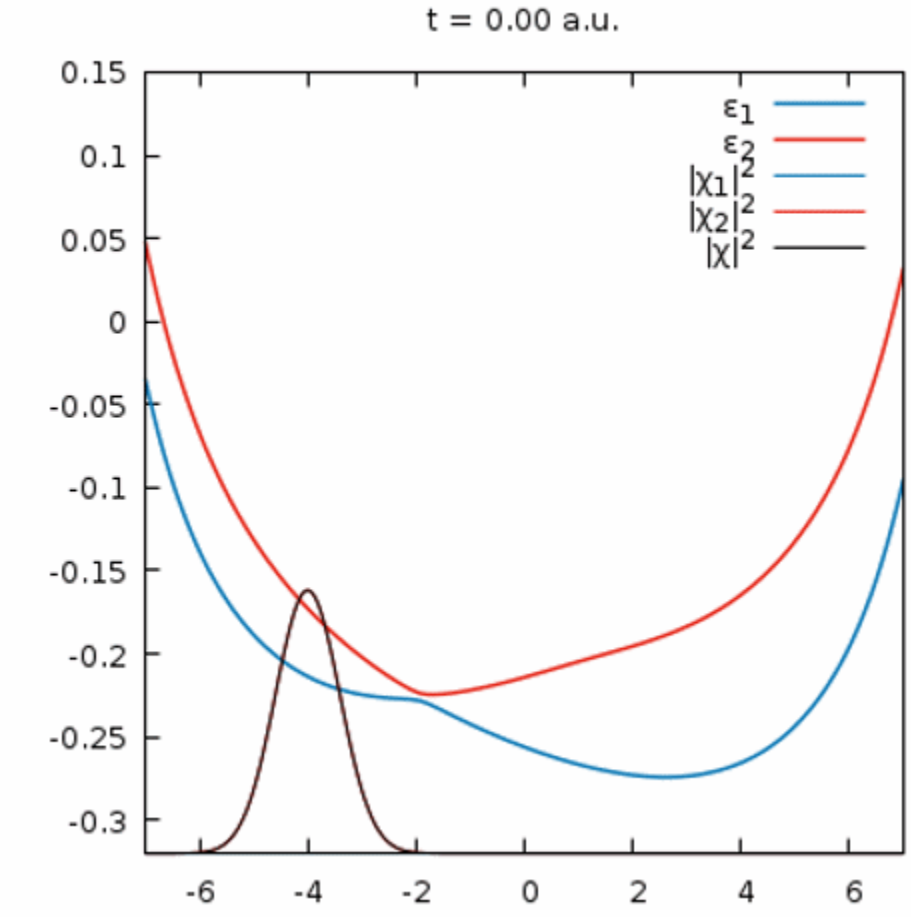
- Exact factorization can capture correct behavior in potential energy surfaces

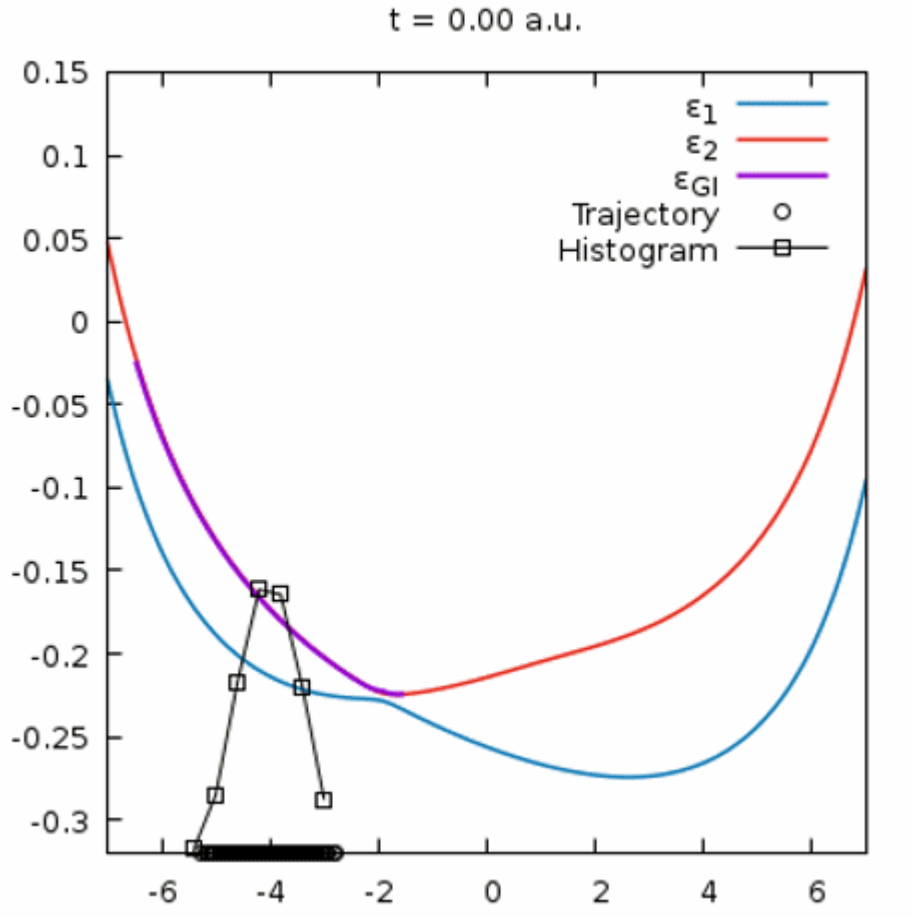


$$\langle\underline{\underline{\mathbf{R}}}_n(t)\rangle = \langle\chi_n(\underline{\underline{\mathbf{R}}},t)|\underline{\underline{\mathbf{R}}}|\chi_n(\underline{\underline{\mathbf{R}}},t)\rangle / \langle\chi_n(\underline{\underline{\mathbf{R}}},t)|\chi_n(\underline{\underline{\mathbf{R}}},t)\rangle$$

Toward mixed quantum-classical approach

Quantum dynamics vs. trajectories on potential from exact factorization





formally exact, practically useless…

Trajectory-based equations of motion

$$\epsilon^{apx}(\underline{\underline{\mathbf{R}}},t) = \left\langle \Phi_{\underline{\underline{\mathbf{R}}}}(t) \middle| \hat{H}_{BO} \middle| \Phi_{\underline{\underline{\mathbf{R}}}}(t) \right\rangle + \left\langle \Phi_{\underline{\underline{\mathbf{R}}}}(t) \middle| -i\hbar\partial_t \middle| \Phi_{\underline{\underline{\mathbf{R}}}}(t) \right\rangle_{\underline{\mathbf{r}}}$$

$$\hat{U}_{en}^{coup}[\Phi_{\underline{\underline{\mathbf{R}}}},\chi] \simeq \sum_{\nu=1}^{N_n} \frac{1}{M_\nu} \left[\left(\tilde{\mathbf{P}}_\nu(\underline{\underline{\mathbf{R}}},t) + \frac{-i\hbar\nabla_\nu|\chi|}{|\chi|} \right) \left(-i\hbar\nabla_\nu - \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t) \right) \right]$$

$$\epsilon(\underline{\underline{\mathbf{R}}},t) = \left\langle \Phi_{\underline{\underline{\mathbf{R}}}}(t) \middle| \hat{H}_{BO} + \hat{U}_{en}^{coup} - i\hbar\partial_t \middle| \Phi_{\underline{\underline{\mathbf{R}}}}(t) \right\rangle_{\underline{\underline{\mathbf{r}}}}$$

$$i\hbar\partial_t \Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\mathbf{r}},t) = \left(\hat{H}_{BO}(\underline{\mathbf{r}},\underline{\underline{\mathbf{R}}}) + \hat{U}_{en}^{coup}[\Phi_{\underline{\underline{\mathbf{R}}}},\chi] - \epsilon(\underline{\underline{\mathbf{R}}},t) \right) \Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\mathbf{r}},t)$$

$$i\hbar\partial_t \chi(\underline{\underline{\mathbf{R}}},t) = \left(\sum_{\nu=1}^{N_n} \frac{\left[-i\hbar\nabla_\nu + \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t) \right]^2}{2M_\nu} + \epsilon(\underline{\underline{\mathbf{R}}},t) \right) \chi(\underline{\underline{\mathbf{R}}},t)$$

$$\tilde{\mathbf{P}}_\nu = \nabla_\nu S_0(\underline{\underline{\mathbf{R}}},t) + \mathbf{A}_\nu(\underline{\underline{\mathbf{R}}},t) \qquad \chi(\underline{\underline{\mathbf{R}}},t) = |\chi(\underline{\underline{\mathbf{R}}},t)| e^{iS(\underline{\underline{\mathbf{R}}},t)/\hbar}$$

Quantum Hamilton-Jacobi eq. → Newtonian equation (trajectory)

Phys. Rev. Lett., **115**, 073001 (2015), *J. Chem. Theory Comput.*, **12**, 2127 (2016).

Trajectory-based equations of motion

- **Electronic equation of motion**

$$i\hbar\frac{d}{dt}\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t) = \hat{H}_{BO}(\underline{\mathbf{r}};\underline{\underline{\mathbf{R}}}^{(I)}(t))\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t)$$
$$+i\hbar\sum_{\nu}^{N_n}\left.\frac{\nabla_\nu|\chi|}{|\chi|}\right|_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}\cdot\left(\mathbf{A}_\nu^{(I)}\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t)+i\hbar\nabla_\nu\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t)\right)$$

- **Nuclear equation of motion**

$$\dot{\mathbf{P}}_\nu = -\left\langle\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(t)\left|\nabla_\nu\hat{H}_{BO}\right|\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(t)\right\rangle_{\underline{\mathbf{r}}}$$
$$+2\sum_{\nu'}^{N_n}\frac{1}{M_{\nu'}}\left(\left.\frac{\nabla_{\nu'}|\chi|}{|\chi|}\right|_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}\cdot\mathbf{A}_{\nu'}^{(I)}(t)\right)\mathbf{A}_\nu^{(I)}(t)$$
$$+2\,\mathrm{Re}\int d\underline{\mathbf{r}}\left[\sum_{\nu'}^{N_n}\frac{1}{M_{\nu'}}\left.\frac{\nabla_{\nu'}|\chi|}{|\chi|}\right|_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}\cdot\left(-i\hbar\nabla_{\nu'}\Phi^*_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t)\right)\right]\left(-i\hbar\nabla_\nu\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t)\right)$$

Phys. Rev. Lett., **115**, 073001 (2015), *J. Chem. Theory Comput.*, **12**, 2127 (2016).

Trajectory-based equations of motion

- **Electronic equation of motion**

$$\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}},t) = \sum_{l=1}^{N_{st}} C_l(\underline{\underline{\mathbf{R}}}^{(I)}(t),t)\varphi^{(l)}_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\mathbf{r}})$$

$$\dot{C}_l^{(I)}(t) = \frac{-i}{\hbar}\epsilon_{BO}^{(l)}\left(\underline{\underline{\mathbf{R}}}^{(I)}(t)\right)C_l^{(I)}(t) - \sum_{\nu=1}^{N_n}\mathbf{v}_\nu^{(I)}\cdot\sum_{k=1}^{N_{st}}C_k^{(I)}(t)\mathbf{d}^{(I)}_{\nu,lk}$$

$$+\sum_{\nu=1}^{N_n}\frac{1}{M_\nu}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{(I)}\cdot\Big(\sum_k \rho^{(I)}_{kk}(t)\mathbf{f}^{(I)}_{k,\nu}(t) - \mathbf{f}^{(I)}_{l,\nu}(t)\Big)C_l^{(I)}(t)$$

- **Nuclear equation of motion**

$$\dot{\tilde{P}}_\nu^{(I)}(t) = -\sum_k \rho^{(I)}_{kk}(t)\nabla_\nu\epsilon^{(k),(I)}_{BO} - \sum_{k,l}\rho^{(I)}_{lk}(t)\left(\epsilon^{(k),(I)}_{BO} - \epsilon^{(l),(I)}_{BO}\right)\mathbf{d}^{(I)}_{\nu,lk}$$

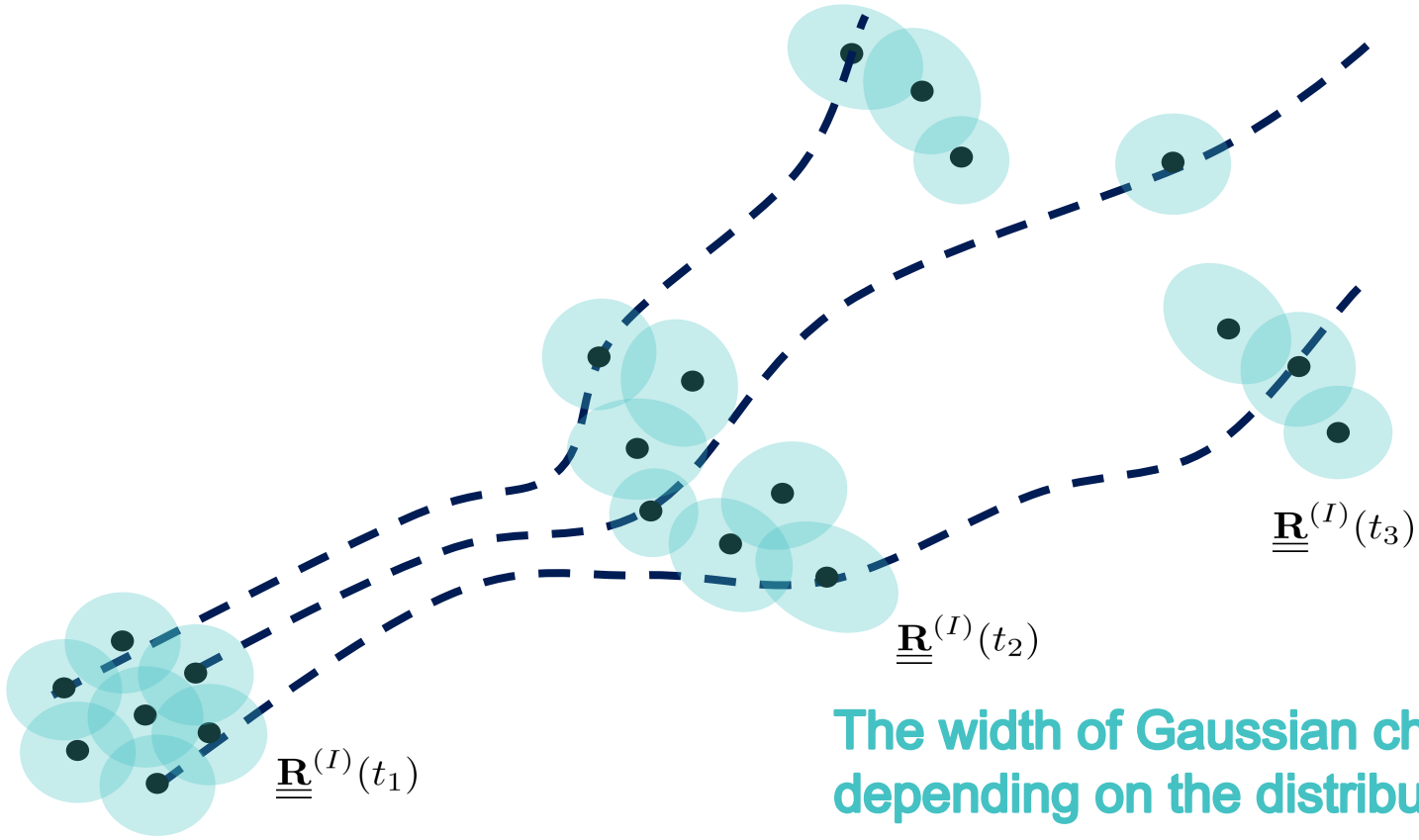
$$-\sum_l \rho^{(I)}_{ll}(t)\left(\sum_{\nu'=1}^{N_n}\frac{2\hbar}{M_{\nu'}}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{(I)}\cdot\mathbf{f}^{(I)}_{l,\nu'}(t)\right)\left[\sum_k \rho^{(I)}_{kk}(t)\mathbf{f}^{(I)}_{k,\nu}(t) - \mathbf{f}^{(I)}_{l,\nu}(t)\right]$$

$$\rho^{(I)}_{lk}(t) = C_l^{(I)*}(t)C_k^{(I)}(t) \qquad \mathbf{f}^{(I)}_{l,\nu}(t) = -\int_0^t dt'\nabla_\nu\epsilon^{(l)}_{BO}$$

Phys. Rev. Lett., **115**, 073001 (2015), *J. Chem. Theory Comput.*, **12**, 2127 (2016).

Coupled trajectory approach (CT-MQC)

- **Calculating quantum momentum from a nuclear density from multiple trajectories**
  - **Put Gaussians on all trajectories, and compute the quantum momentum**

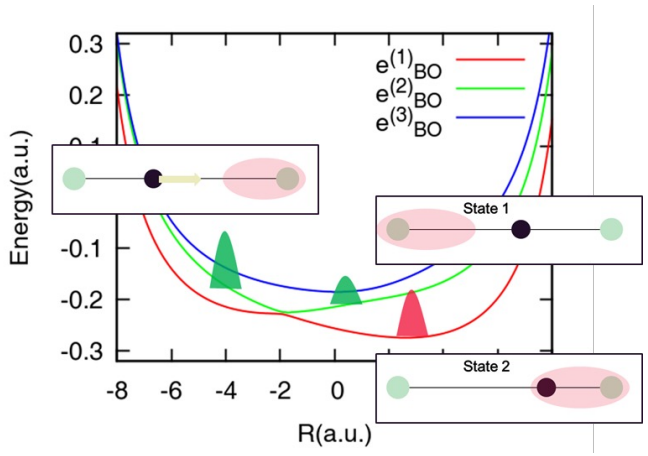


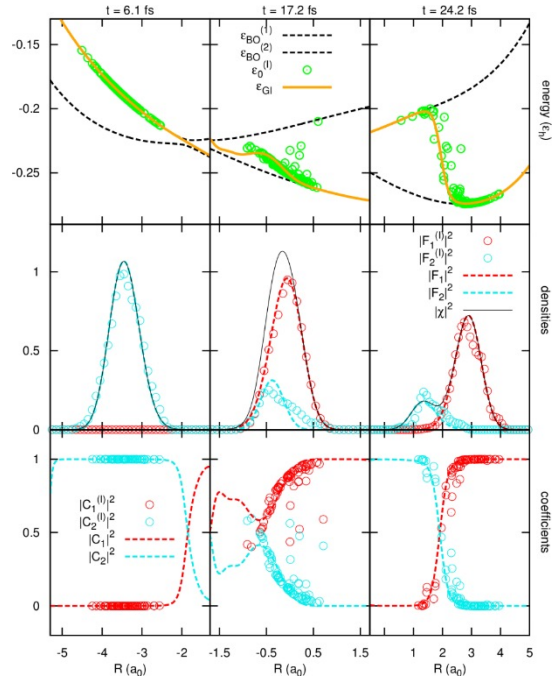
The width of Gaussian changes in time depending on the distribution of trajectories

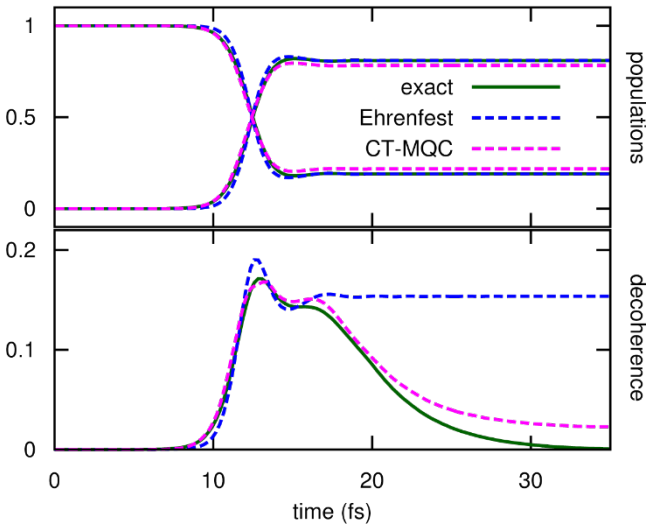
Phys. Rev. Lett., **115**, 073001 (2015), *J. Chem. Theory Comput.*, **12**, 2127 (2016).

Coupled trajectory approach (CT-MQC)

- **Additional ingredients compared to Ehrenfest/Surface hopping dynamics**
  - **Positions of all trajectories to construct nuclear quantum momentum**
  - **Phase factors from all BO forces** $\mathbf{f}_{l,\nu}^{(I)}(t) = -\int_0^t dt' \nabla_\nu \epsilon_{BO}^{(l)}$
  - **All trajectories should be run simultaneously!**





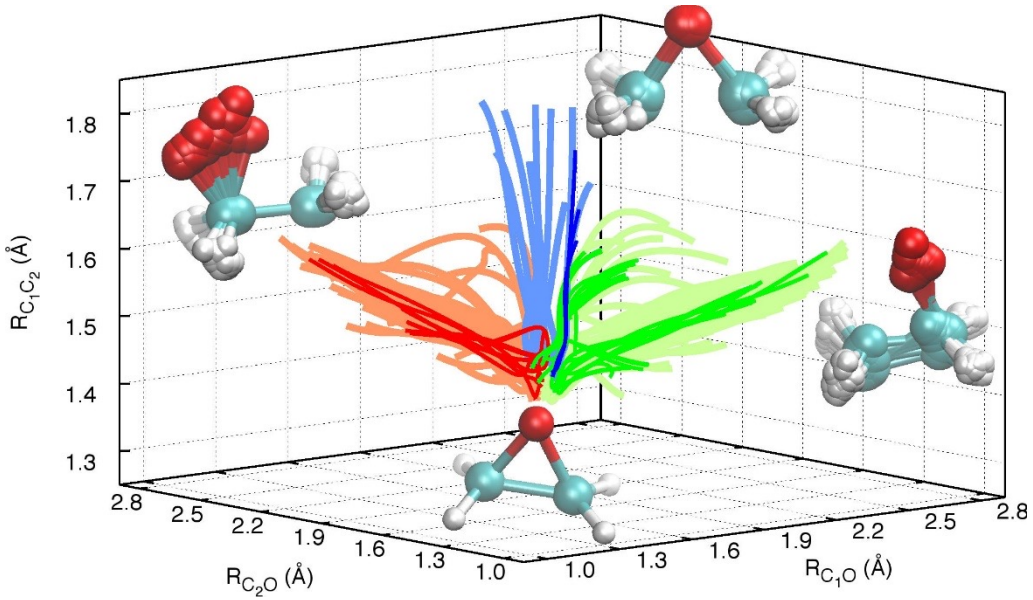


QM $\int dR |C_1(R,t)|^2 |C_2(R,t)|^2 |\chi(R,t)|^2$

CT $\sum_I |C_1^{(I)}(t)|^2 |C_2^{(I)}(t)|^2 / N_{traj}$

First principle-based implementation

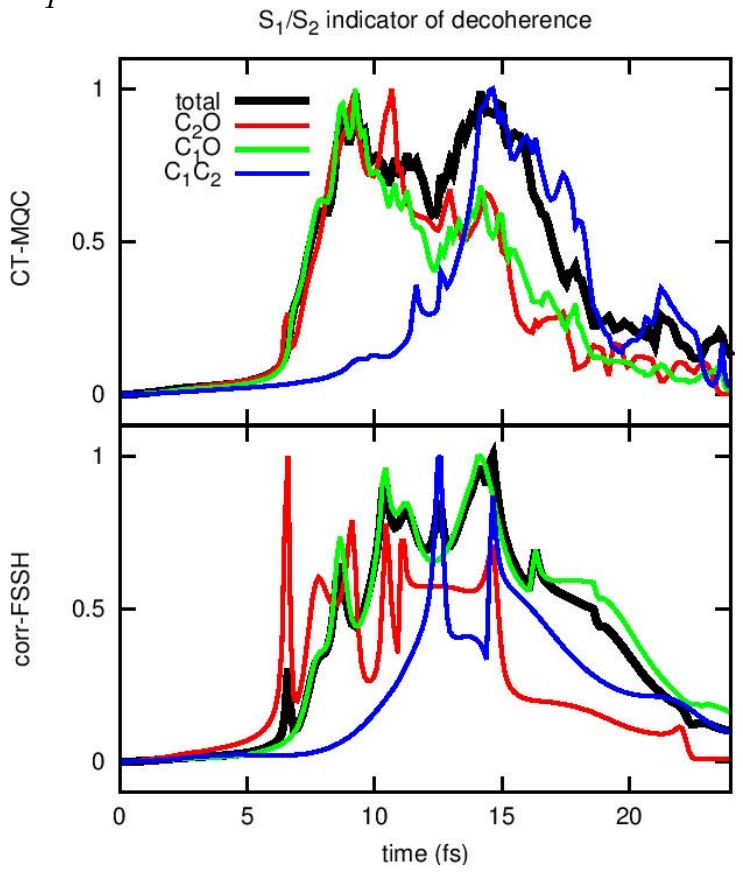
- Car-Parrinello Molecular Dynamics (CPMD) program
- Excited state calculation with LR-TDDFT
- Photodissociation dynamics of Oxirane ($S_2 \rightarrow S_1$)



100 trajectories
with initial Boltzmann distribution at 300K

Quantum-like trajectories for CT-MQC
(deep: (corr)-FSSH, pastel: CT-MQC)

$$\sum_I |C_1^{(I)}(t)|^2 |C_2^{(I)}(t)|^2 / N_{traj}$$



J. Phys. Chem. Lett., 8, 3048 (2017).

Independent (auxiliary)-trajectory method: Decoherence-induced surface hopping based on exact factorization (DISH-XF)

- Coupled-trajectory algorithm requires …
  - Nonadiabatic coupling vectors (NACVs) among all states
    c.f.) surface hopping : NACVs projected on velocities (NACME)
  - BO force for all electronic states
  - BO calculations should be stable for all trajectories
- Coupled-trajectory algorithm is efficient, but less efficient than the conventional surface hopping algorithm

Modified surface hopping algorithm based on exact factorization

➡ *Decoherence-induced surface hopping based on exact factorization (DISH-XF)*

J. Phys. Chem. Lett., 9, 1097 (2018).

Independent (auxiliary)-trajectory method: Surface hopping based on exact factorization (SHXF)

- Nuclear equation: same as the conventional surface hopping,
  i.e. nuclear force from the running state or force state

- Electronic equation with the decoherence term

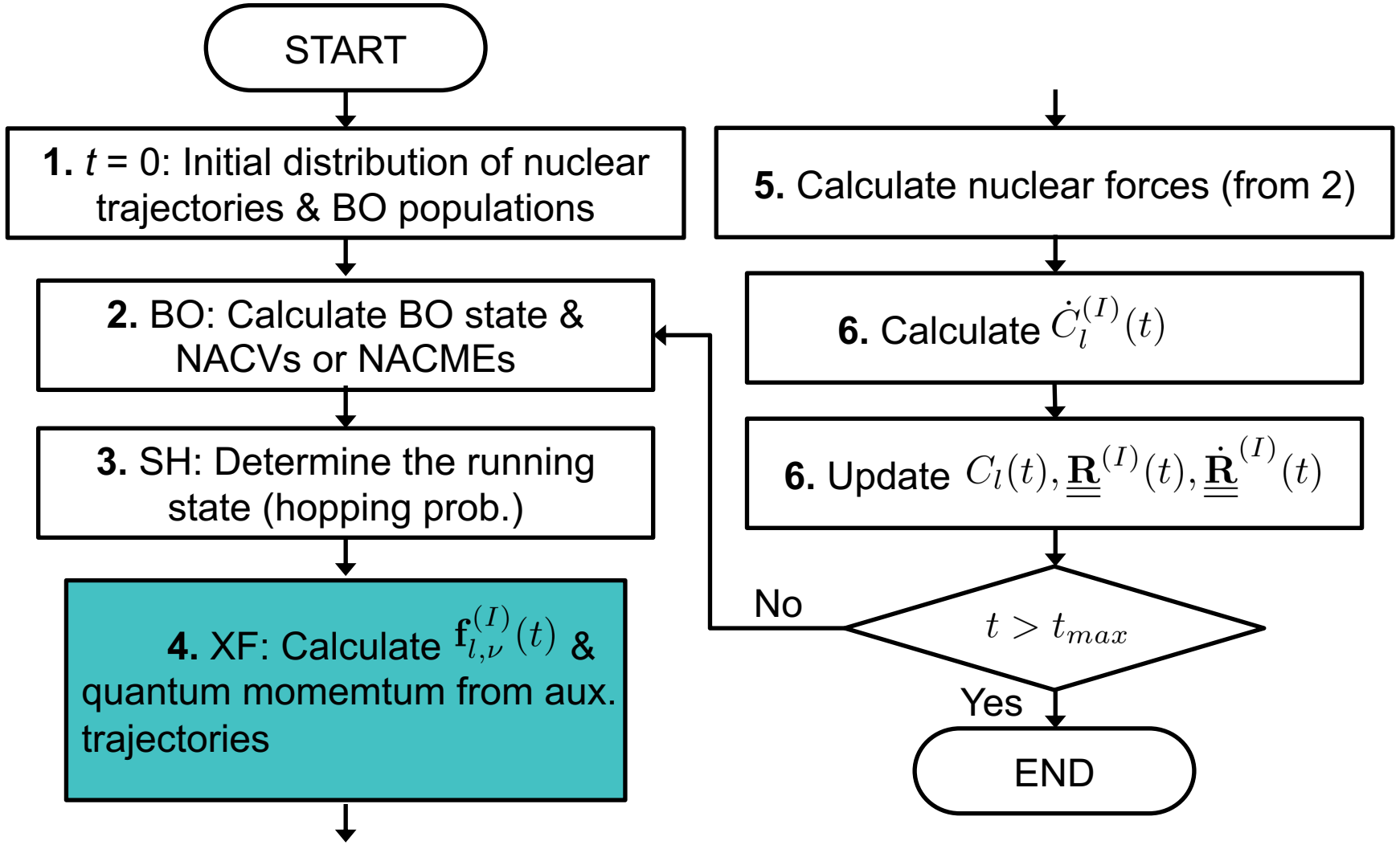
$$\dot{C}_l^{(I)}(t) = \frac{-i}{\hbar}\epsilon_{BO}^{(l)}\left(\underline{\underline{\mathbf{R}}}^{(I)}(t)\right)C_l^{(I)}(t) - \sum_{\nu=1}^{N_n}\mathbf{v}_\nu^{(I)}\cdot\sum_{k=1}^{N_{st}}C_k^{(I)}(t)\mathbf{d}_{\nu,lk}^{(I)}$$
$$+\sum_{\nu=1}^{N_n}\frac{1}{M_\nu}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{(I)}\cdot\Big(\sum_k\rho_{kk}^{(I)}(t)\mathbf{f}_{k,\nu}(t) - \mathbf{f}_{l,\nu}^{(I)}(t)\Big)C_l^{(I)}(t)$$

- Hopping probability: same as the conventional surface hopping

- Quantum momentum from auxiliary trajectories :
  put "frozen" (finite width as an external parameter) Gaussian wave packet
  when population exchange occurs

J. Phys. Chem. Lett., 9, 1097 (2018).

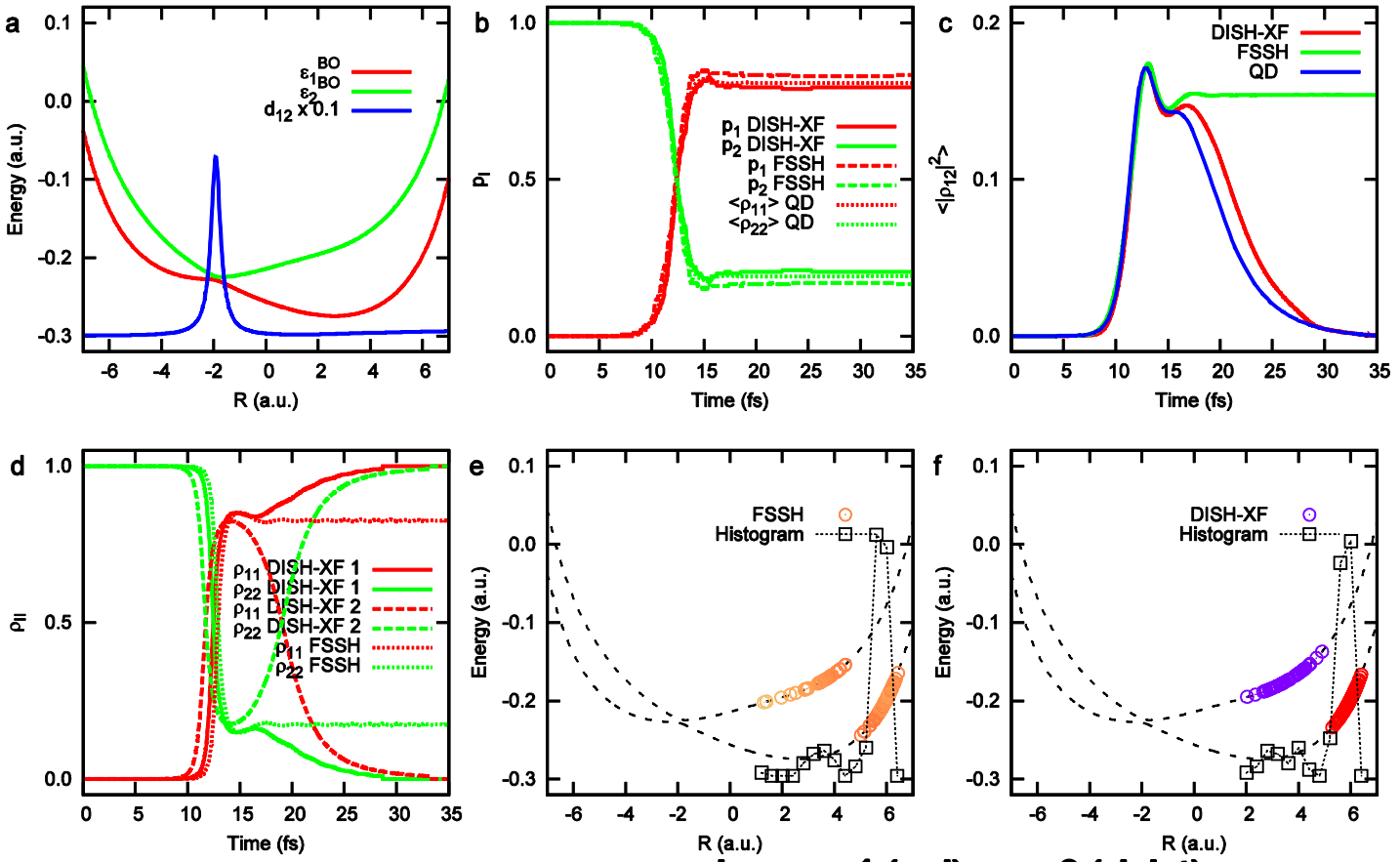
Numerical implementation of SHXF



J. Phys. Chem. Lett., 9, 1097 (2018).

Applications of SHXF

- **Shin-Metiu model (single crossing)**

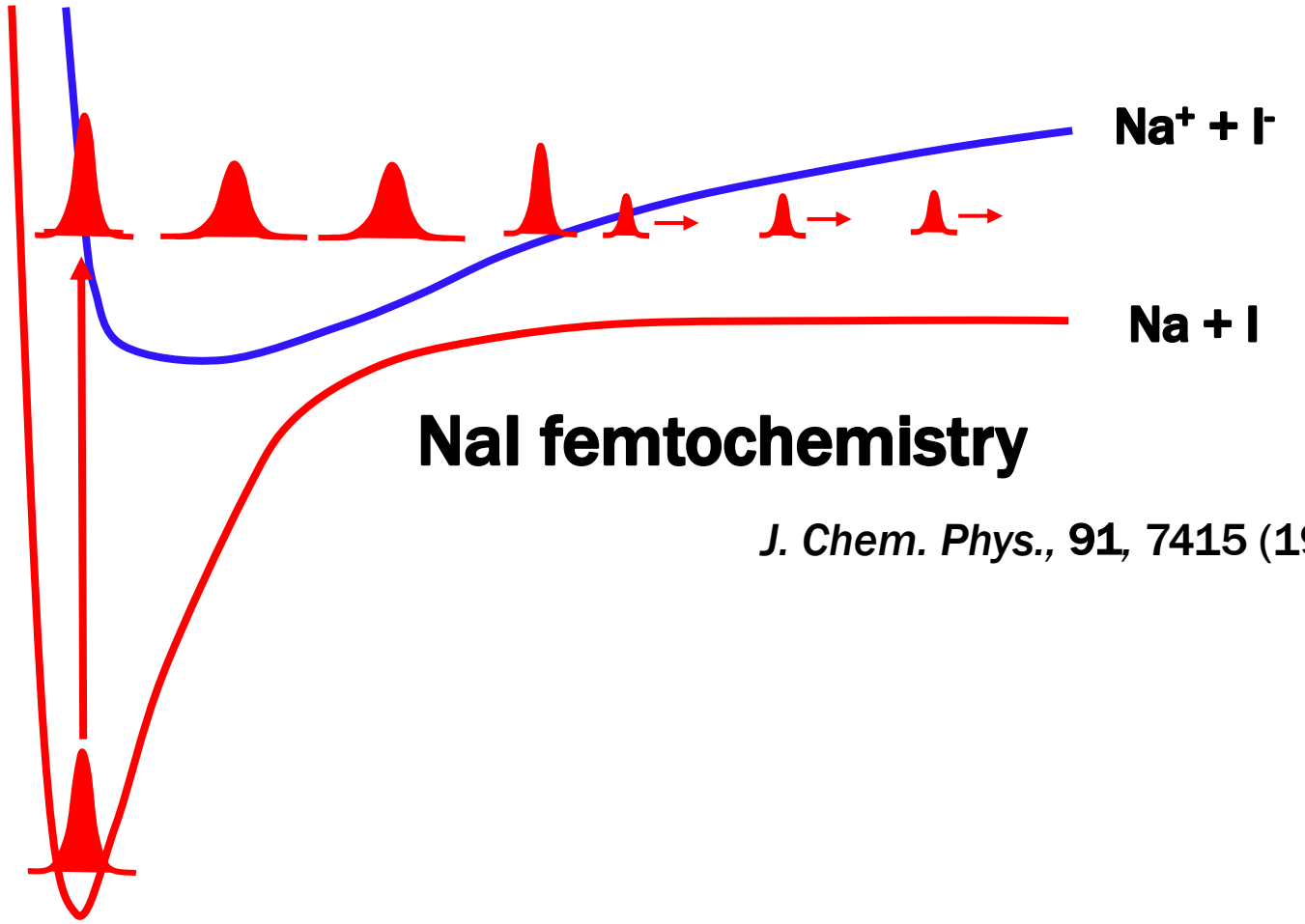


color: ρ_{11}=1 (red), ρ_{11}=0 (violet), 0<ρ_{11}<1 (orange)

J. Phys. Chem. Lett., 9, 1097 (2018).

Applications of SHXF

- NaI pump probe experiments (multiple crossing)



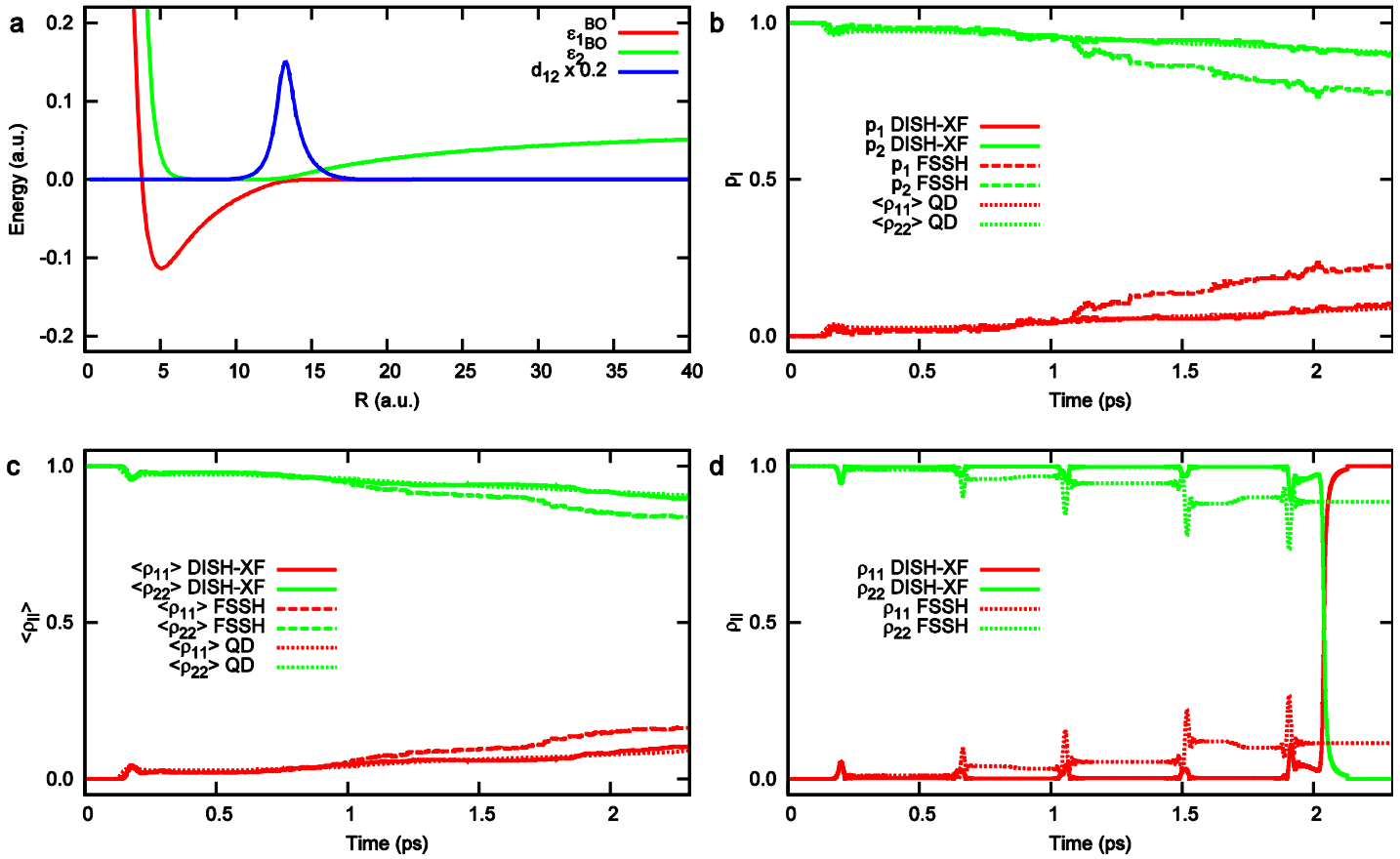
NaI femtochemistry

J. Chem. Phys., **91**, 7415 (1989).

J. Phys. Chem. Lett., 9, 1097 (2018).

Applications of SHXF

- NaI pump probe experiments (multiple crossing)

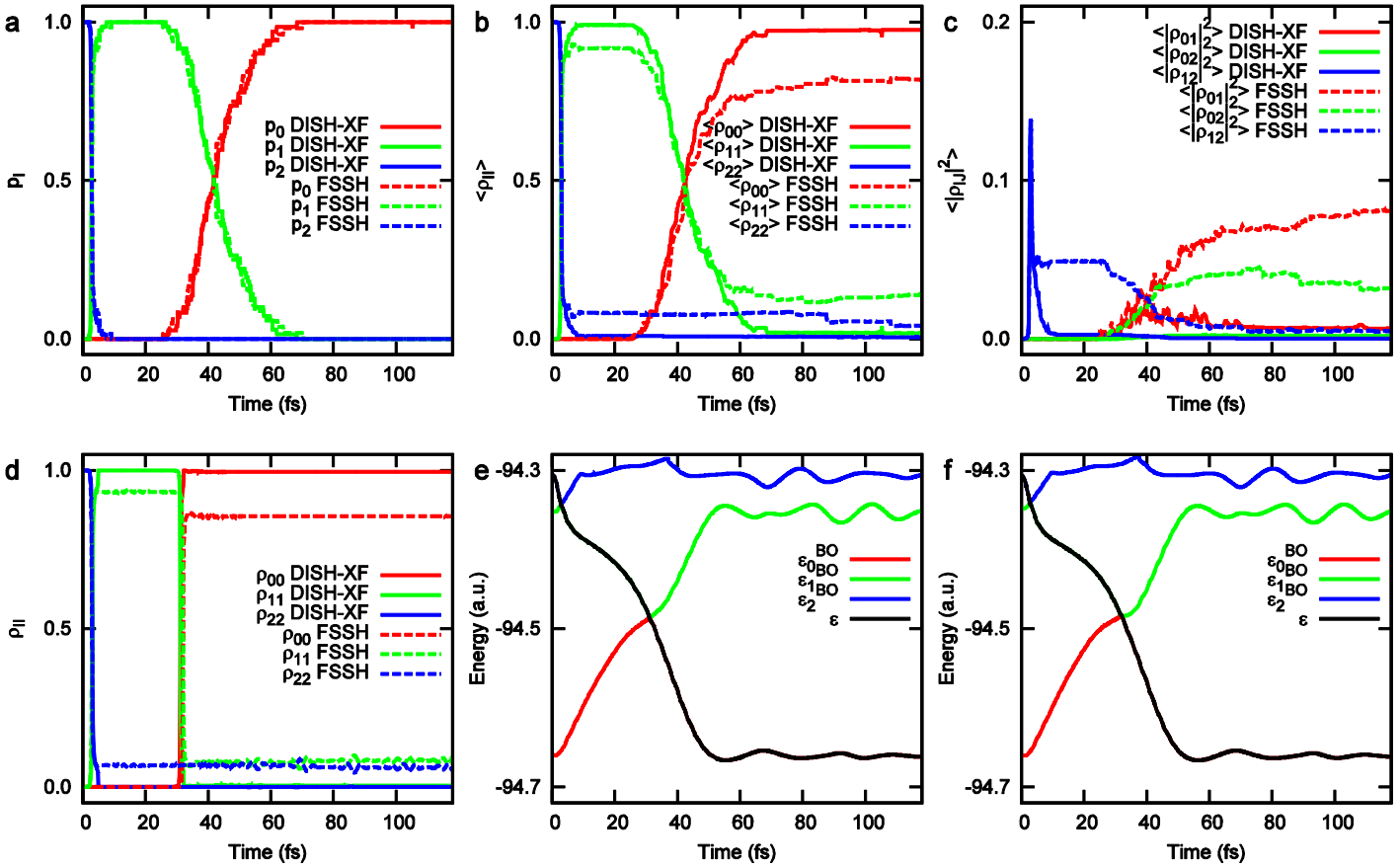


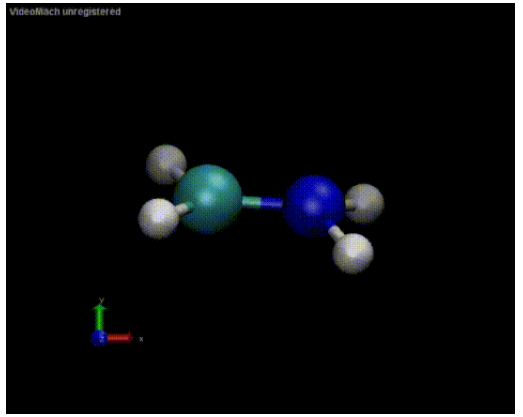
p_I: population from the number of trajectories
$\langle\rho_{II}\rangle$: population from coefficients (averaged over trajectories or space)

J. Phys. Chem. Lett., 9, 1097 (2018).

Applications of SHXF

- **$CH_2NH_2^+$: CASSCF(2,2)/MRCI calculation, 300K, $S_2 \rightarrow (CI) \rightarrow S_1 \rightarrow (CI) \rightarrow S_0$**





p_I: population from the number of trajectories
$\langle \rho_{II} \rangle$: population from coefficients (averaged over trajectories or space)

J. Phys. Chem. Lett., 9, 1097 (2018).

Role of the electron-nuclear correlation (ENC) term

- **Population change in time**

$$\frac{d}{dt}\rho_{ll}^{(I)}(t) = -2\sum_{k=1}^{N_{st}} \mathrm{Re}\left(\rho_{lk}^{(I)}(t)\sum_{\nu=1}^{N_n}\mathbf{v}_{\nu}^{(I)}\cdot\mathbf{d}_{\nu,lk}^{(I)}\right)$$
$$+\sum_{\nu=1}^{N_n}\frac{2}{M_\nu}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{(I)}\cdot\sum_{k=1}^{N_{st}}\left[\mathbf{f}_{k,\nu}^{(I)}-\mathbf{f}_{l,\nu}^{(I)}\right]\rho_{kk}^{(I)}(t)\rho_{ll}^{(I)}(t)$$

If $\mathbf{d}_{\nu,lk}^{(1)} = 0$,

$$\frac{d}{dt}\rho_{ll}^{(I)}(t) = \sum_{\nu=1}^{N_n}\frac{2}{M_\nu}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{(I)}\cdot\sum_{k=1}^{N_{st}}\left[\mathbf{f}_{k,\nu}^{(I)}-\mathbf{f}_{l,\nu}^{(I)}\right]\rho_{kk}^{(I)}(t)\rho_{ll}^{(I)}(t)$$
$$\neq 0 \text{ if } 0 < \rho_{ll}^{(I)} < 1$$
$$= 0 \text{ if } \rho_{ll}^{(I)} = 0 \text{ or } 1$$

Phys. Rev. Lett., **115**, 073001 (2015).

Role of the electron-nuclear correlation (ENC) term

- When we put Gaussian wave packets on surfaces, we obtain

$$\frac{d}{dt}\rho_{ll}^{(I)}(t) = \sum_{\nu=1}^{N_n} \left.\frac{2}{M_\nu}\frac{\nabla_\nu|\chi|}{|\chi|}\right|_{(I)} \cdot \sum_{k=1}^{N_{st}} \left[\mathbf{f}_{k,\nu}^{(I)} - \mathbf{f}_{l,\nu}^{(I)}\right] \rho_{kk}^{(I)}(t)\rho_{ll}^{(I)}(t)$$

$$= -\sum_{\nu=1}^{N_n} \frac{1}{M_\nu}\frac{1}{\sigma^2}\left(\mathbf{R}_\nu^{(I)} - \langle \mathbf{R}_\nu \rangle\right) \cdot \sum_{k=1}^{N_{st}} \left[\mathbf{f}_{k,\nu}^{(I)} - \mathbf{f}_{l,\nu}^{(I)}\right] \rho_{kk}^{(I)}(t)\rho_{ll}^{(I)}(t)$$

"Position" dependence — Averaged center of overall Gaussians

"Momentum" change — $\left(\mathbf{f}_{l,\nu}^{(I)}(t) = -\int_0^t dt' \nabla_\nu \epsilon_{BO}^{(l)}\right)$

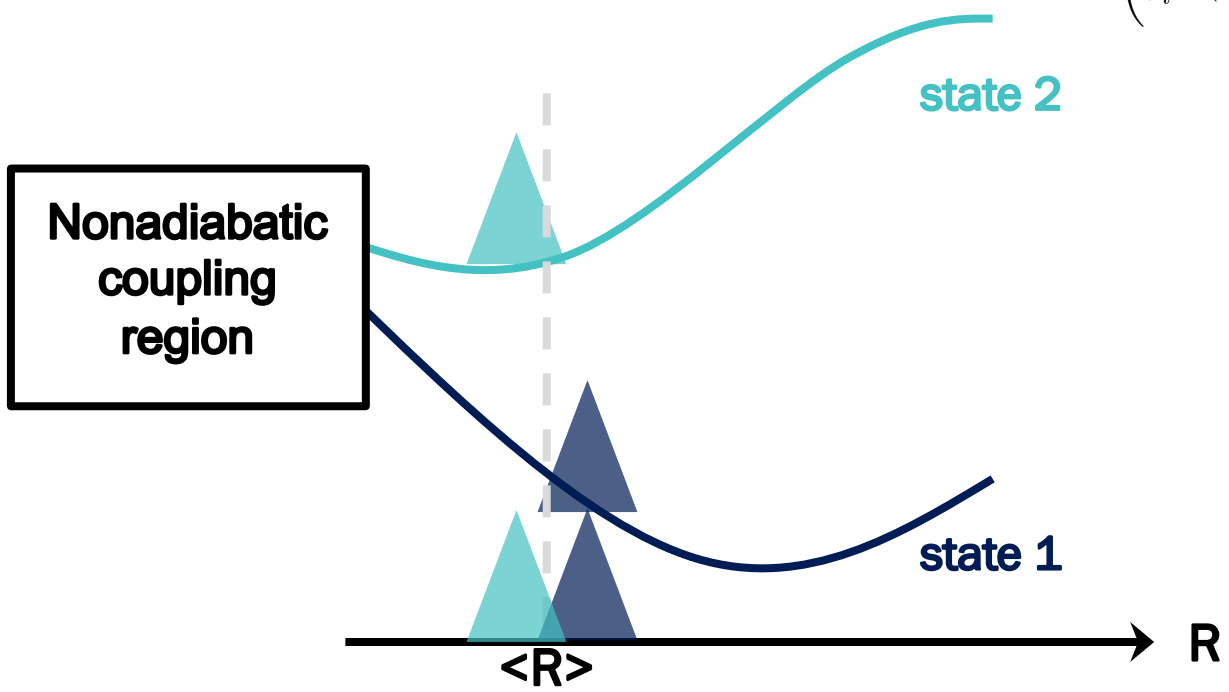
Role of the electron-nuclear correlation (ENC) term

- **e.g. 1D 2-state problem**

$$\frac{d}{dt}\rho_{11}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_2^{(I)} - f_1^{(I)}\right]\rho_{22}^{(I)}(t)\rho_{11}^{(I)}(t)$$

$$\frac{d}{dt}\rho_{22}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_1^{(I)} - f_2^{(I)}\right]\rho_{11}^{(I)}(t)\rho_{22}^{(I)}(t)$$

$$\left(f_l^{(I)}(t) = -\int_0^t dt'\partial_R\epsilon_{BO}^{(l)}\right)$$



Role of the electron-nuclear correlation (ENC) term

- e.g. 1D 2-state problem

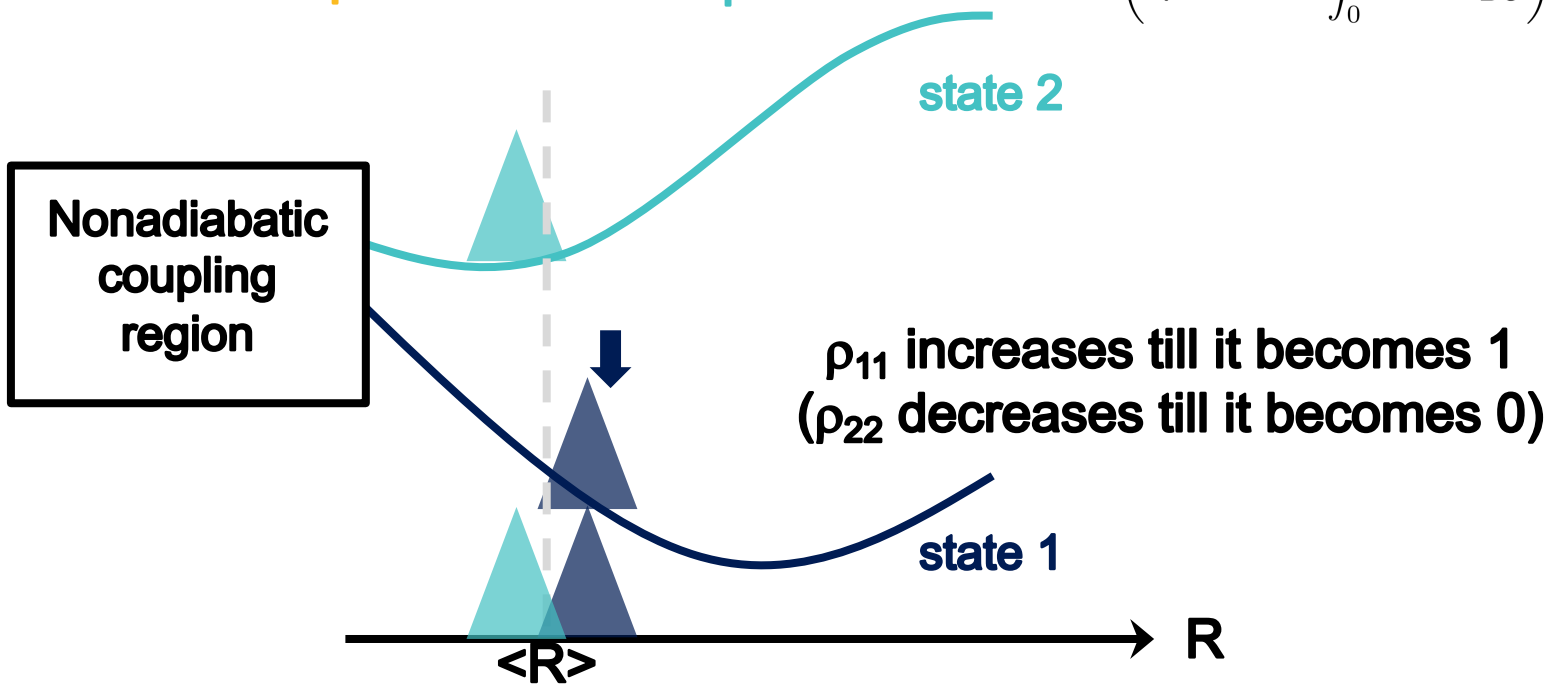
negative

$$\frac{d}{dt}\rho_{11}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_2^{(I)} - f_1^{(I)}\right]\rho_{22}^{(I)}(t)\rho_{11}^{(I)}(t) \qquad > 0$$

$$\frac{d}{dt}\rho_{22}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_1^{(I)} - f_2^{(I)}\right]\rho_{11}^{(I)}(t)\rho_{22}^{(I)}(t) \qquad < 0$$

positive positive

$$\left(f_l^{(I)}(t) = -\int_0^t dt' \partial_R \epsilon_{BO}^{(l)}\right)$$



Role of the electron-nuclear correlation (ENC) term

- e.g. 1D 2-state problem

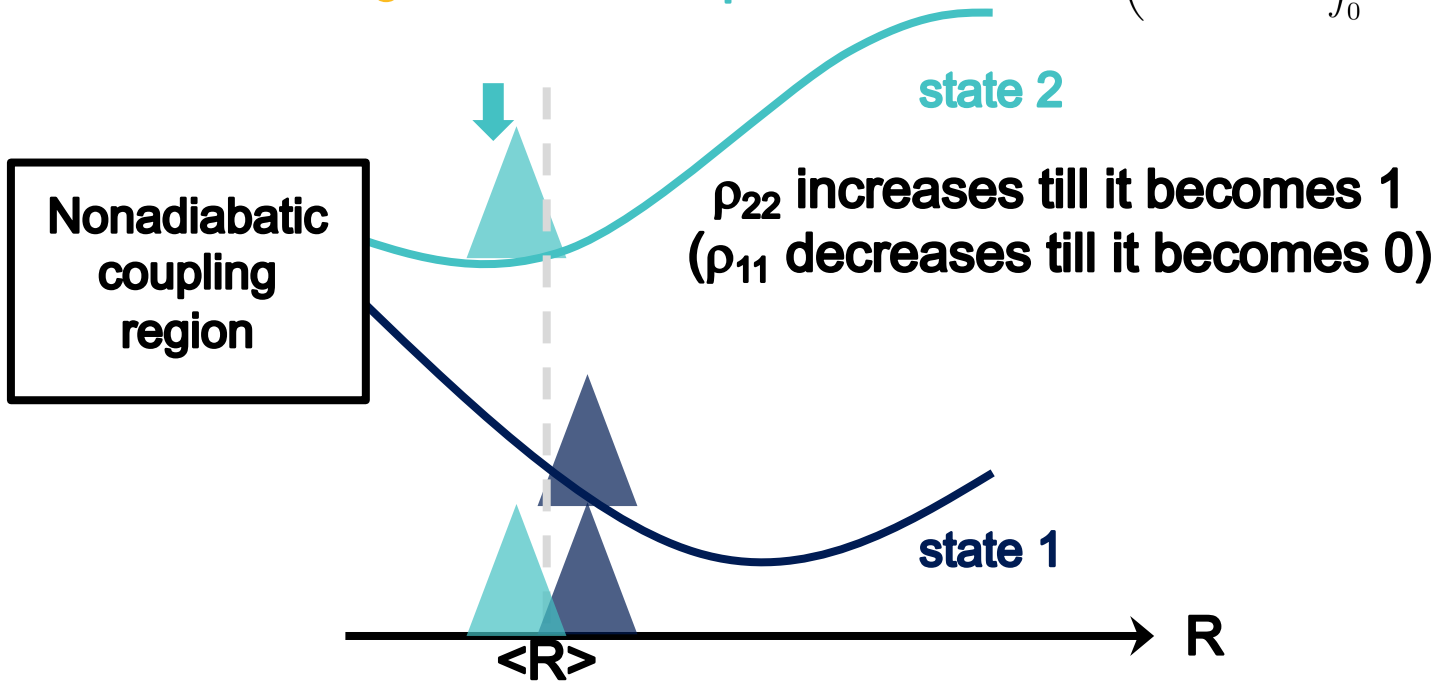
negative

$$\frac{d}{dt}\rho_{11}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_2^{(I)} - f_1^{(I)}\right]\rho_{22}^{(I)}(t)\rho_{11}^{(I)}(t) \qquad < 0$$

$$\frac{d}{dt}\rho_{22}^{(I)}(t) = -\frac{1}{M}\frac{1}{\sigma^2}\left(R^{(I)} - \langle R\rangle\right)\cdot\left[f_1^{(I)} - f_2^{(I)}\right]\rho_{11}^{(I)}(t)\rho_{22}^{(I)}(t) \qquad > 0$$

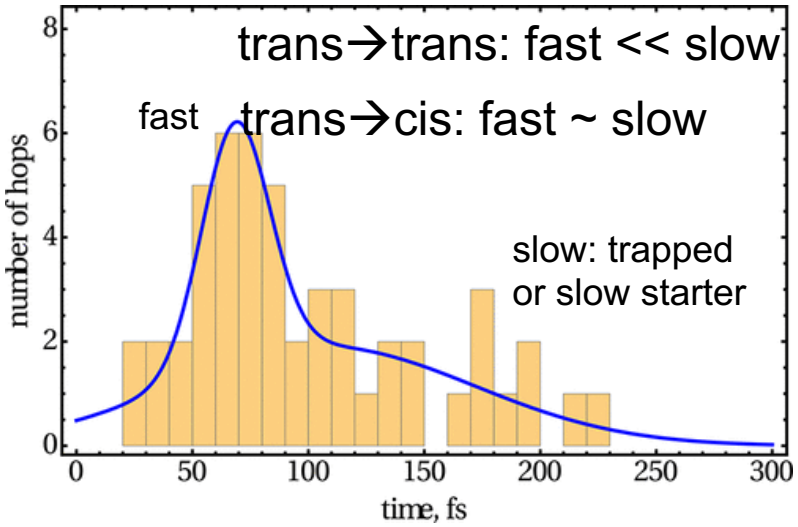
negative positive

$$\left(f_l^{(I)}(t) = -\int_0^t dt' \partial_R \epsilon_{BO}^{(l)}\right)$$

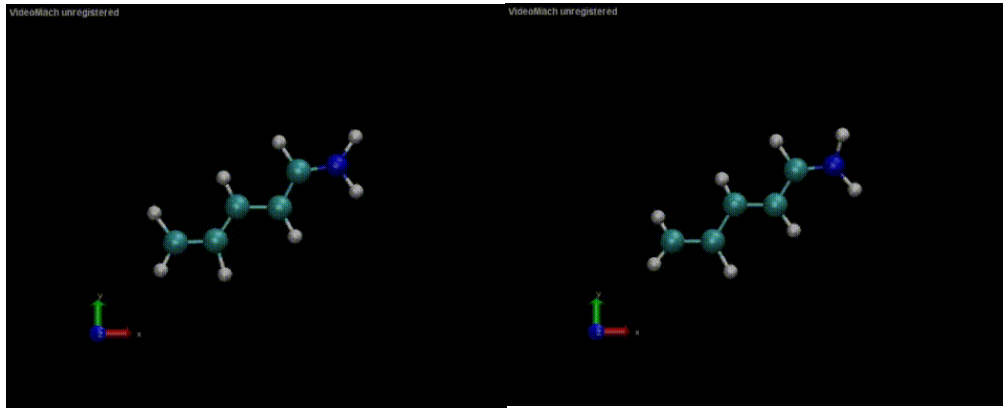


Applications of SHXF

- **Dynamics of PSB3**

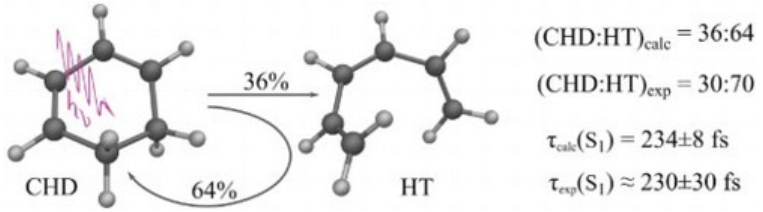


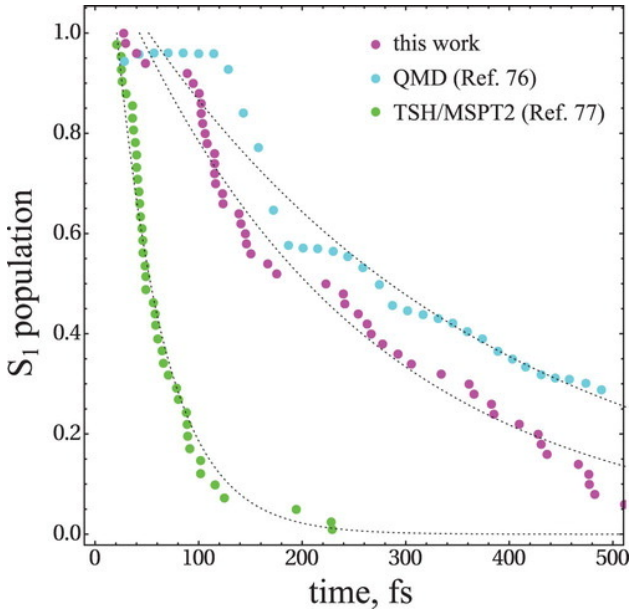
analysis of hop time



J. Chem. Theory Comput., 14, 4499 (2018).

- **Ring opening dynamics of Cyclohexadiene**

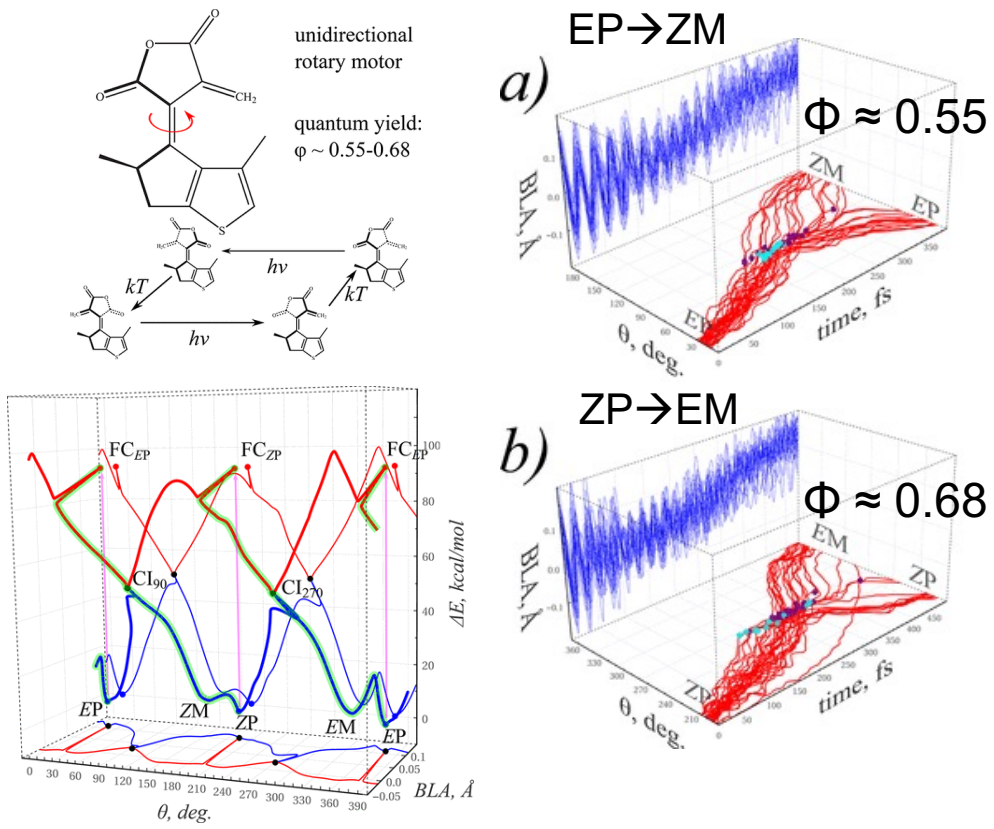




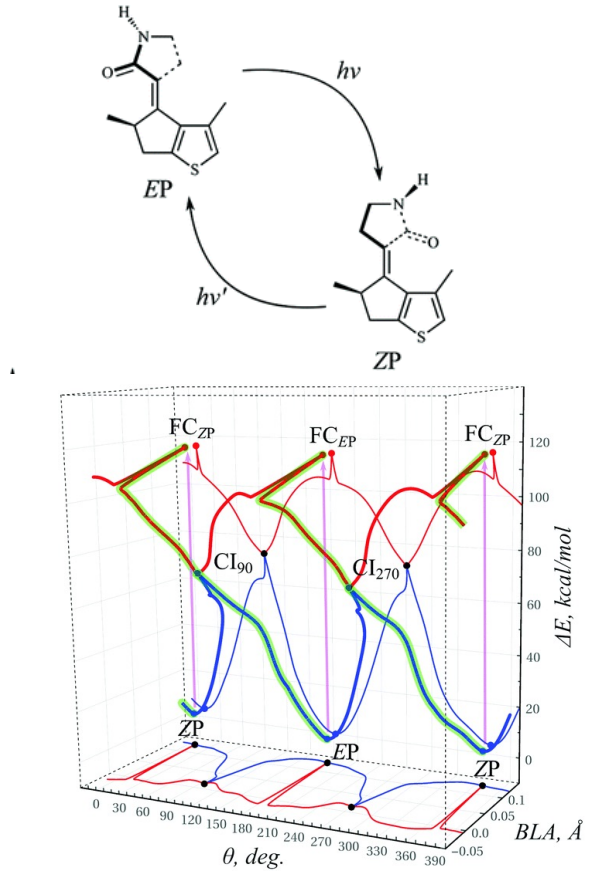
Mol. Phys. 117, 1128 (2019).

Applications of SHXF

- **molecular rotary motors**



J. Phys. Chem. Lett., 9, 4995 (2018).



Chem. Comm., 55, 5247 (2019).

In collaboration with Prof. Michael Filatov and Prof. C.H.Choi

Python-based UNIversal eXcited-state Molecular Dynamics

- **pyUNIxMD program**
  - **Python-based program suite for trajectory-based excited state molecular dynamics (BOMD, FSSH+α, Ehrenfest, CTMQC, DISH-XF, …**

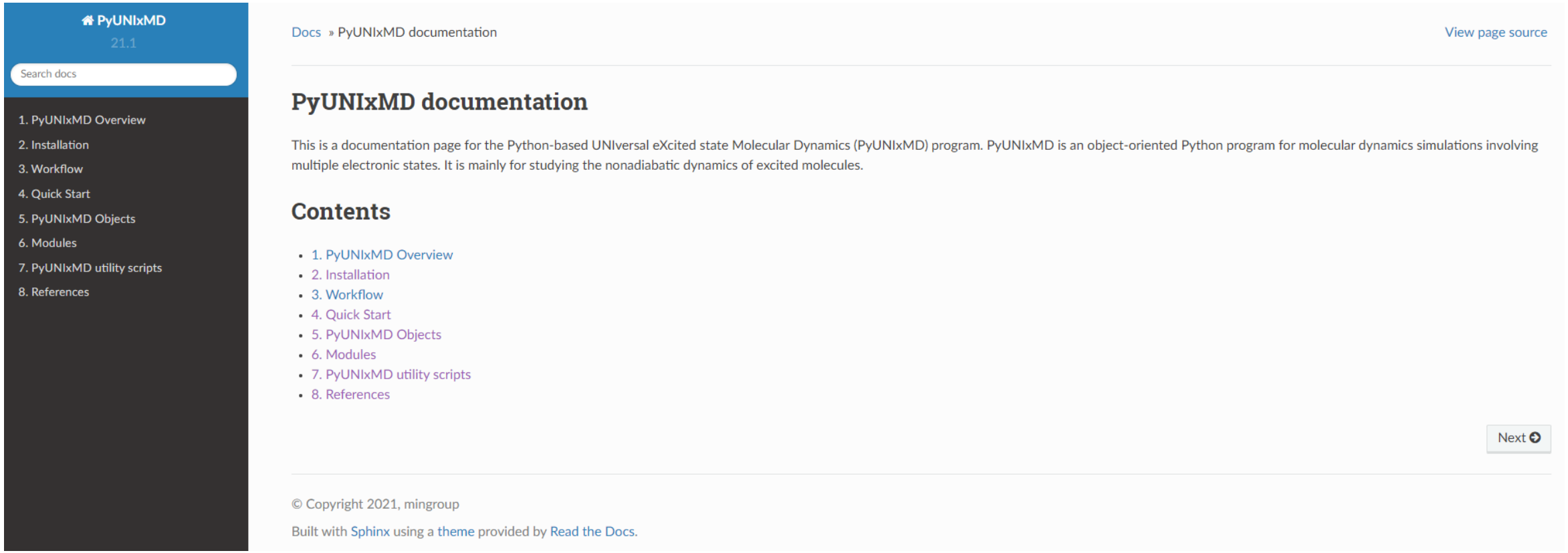
  - **Interfaced with many quantum chemistry programs as:**

COLUMBUS	Molpro	TeraChem	GAMESS
Q-Chem	Gaussian	DFTB+	Any home-made program

  - **Public (https://github.com/skmin-lab/unixmd) with MIT license**

J. Comp. Chem., 42, 1755-1766 (2021).

Documentation of pyUNIxMD



PyUNIxMD
21.1

Search docs

1. PyUNIxMD Overview
2. Installation
3. Workflow
4. Quick Start
5. PyUNIxMD Objects
6. Modules
7. PyUNIxMD utility scripts
8. References

Docs » PyUNIxMD documentation

View page source

PyUNIxMD documentation

This is a documentation page for the Python-based UNIversal eXcited state Molecular Dynamics (PyUNIxMD) program. PyUNIxMD is an object-oriented Python program for molecular dynamics simulations involving multiple electronic states. It is mainly for studying the nonadiabatic dynamics of excited molecules.

Contents

Next

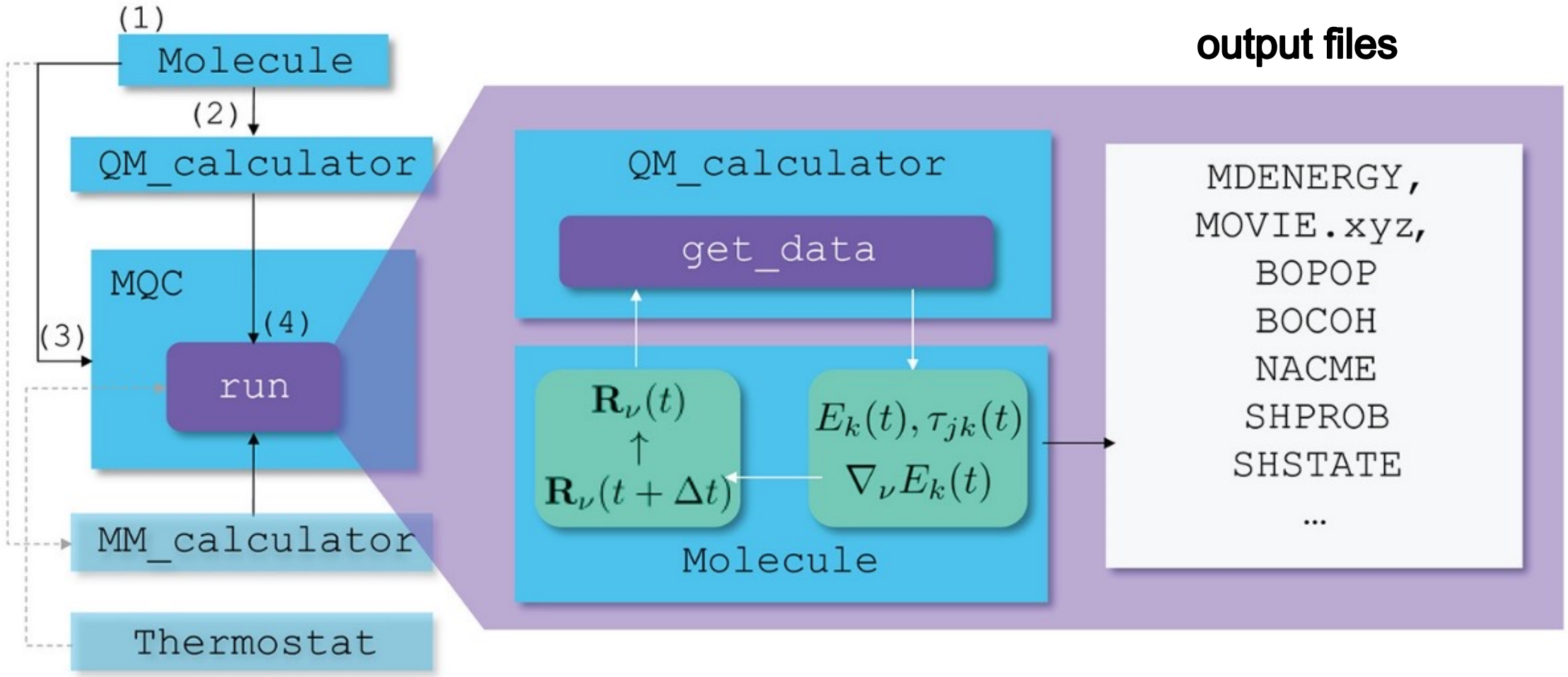
Available MQC algorithms

- Ehrenfest dynamics
  - Ehrenfest dynamics based on exact factorization
- Surface hopping dynamics and its variants
  - Fewest-switch surface hopping (FSSH) (Original Tully's version)
  - Energy-based decoherence correction (EDC)
  - instanteneous decoherence correction (IDC)
  - Surface hopping based on exact factorization (SHXF)
- Coupled-trajectory MQC (CTMQC)

Structures and Flow of pyUNIxMD

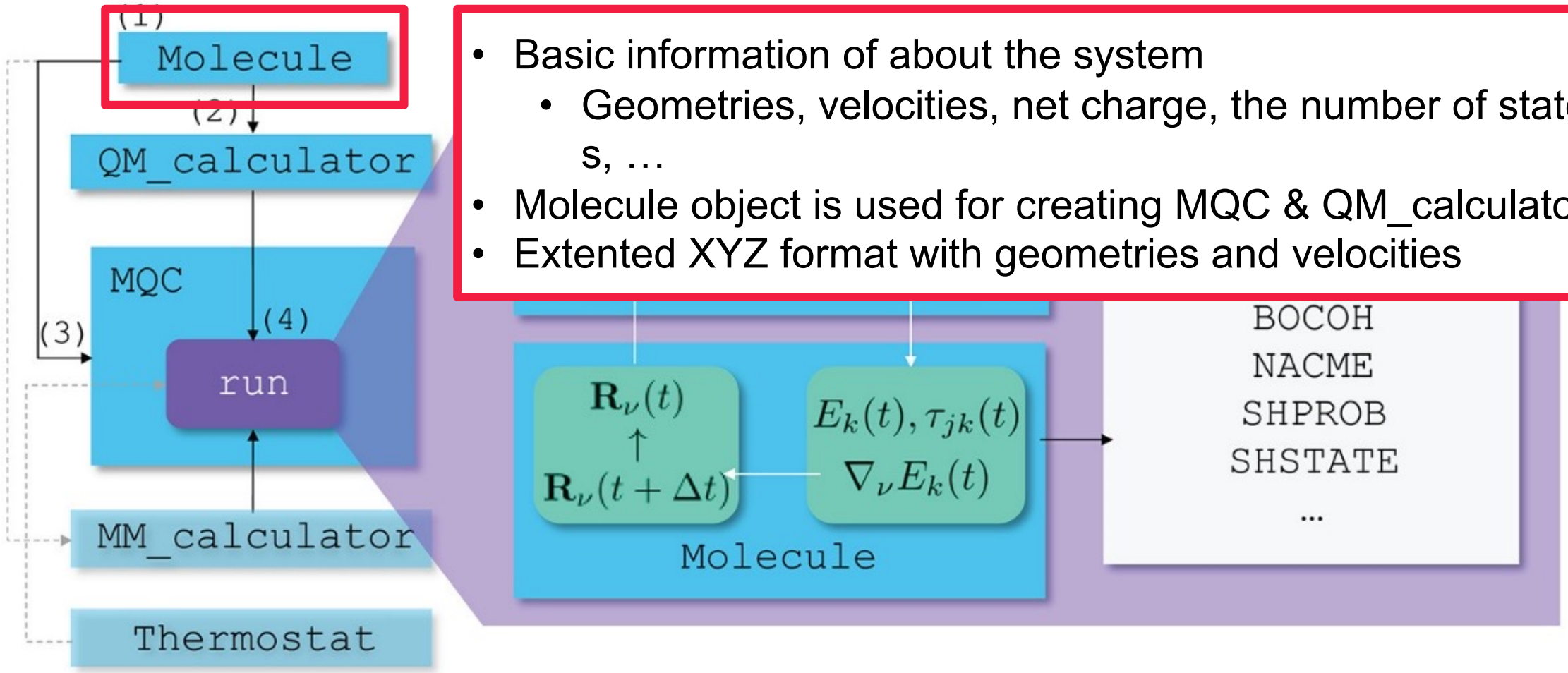
- Object-oriented programming – various classes for their own jobs
  - Main classes : `Molecule`, `QM_calculator`, `MQC`, …
  - `Molecule` – definition of a target system
  - `QM_calculator` – definition of quantum chemical methodologies
  - `MQC` – definition of mixed quantum classical methods
  - `MM_calculator` – definition of MM calculations (*optional*)
  - `Thermostat` – definition of thermostat (Berensden, rescaling, …) (*optional*)
  - …

Structures and Flow of pyUNIxMD



*Arrows represent the flow of information

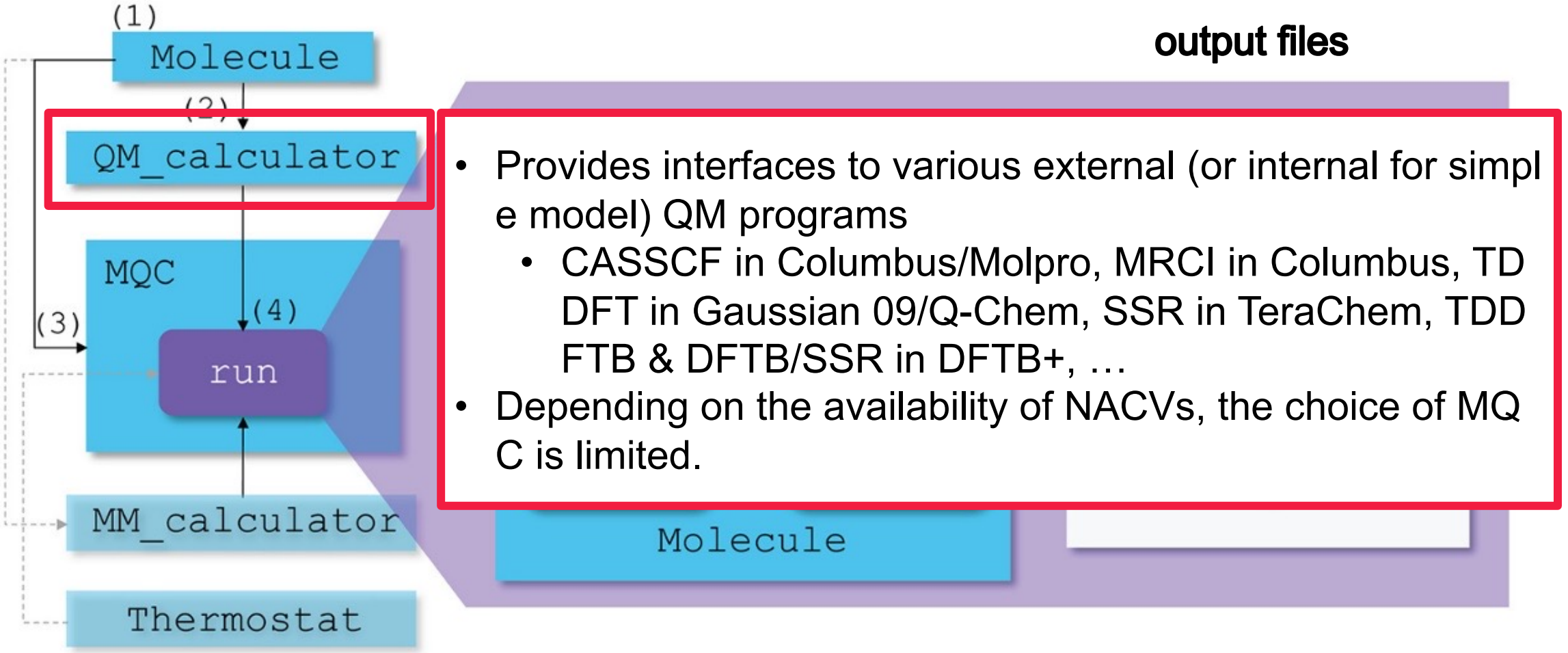
Structures and Flow of pyUNIxMD



- Basic information of about the system
  - Geometries, velocities, net charge, the number of state s, …
- Molecule object is used for creating MQC & QM_calculator
- Extented XYZ format with geometries and velocities

***Arrows represent the flow of information**

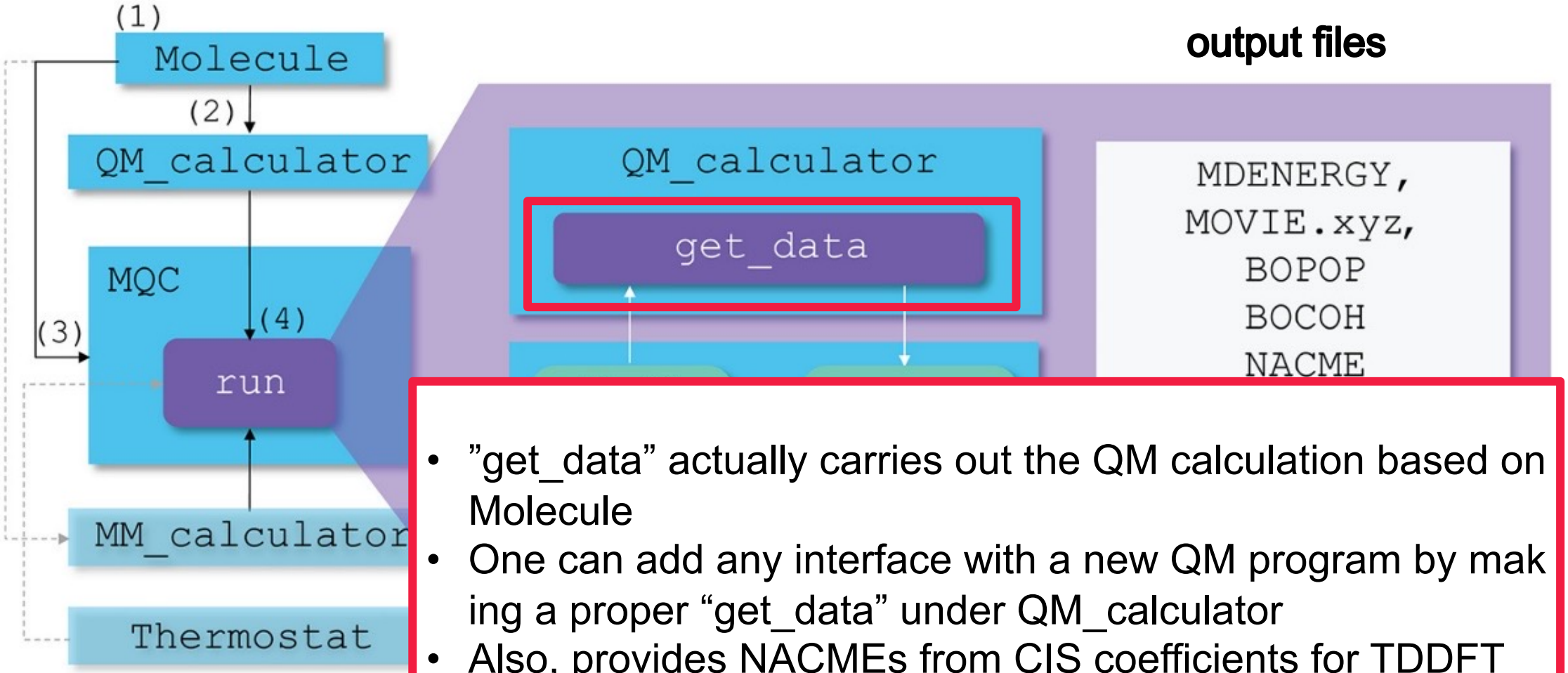
Structures and Flow of pyUNIxMD



- Provides interfaces to various external (or internal for simple model) QM programs
  - CASSCF in Columbus/Molpro, MRCI in Columbus, TD DFT in Gaussian 09/Q-Chem, SSR in TeraChem, TDDFTB & DFTB/SSR in DFTB+, …
- Depending on the availability of NACVs, the choice of MQC is limited.

***Arrows represent the flow of information**

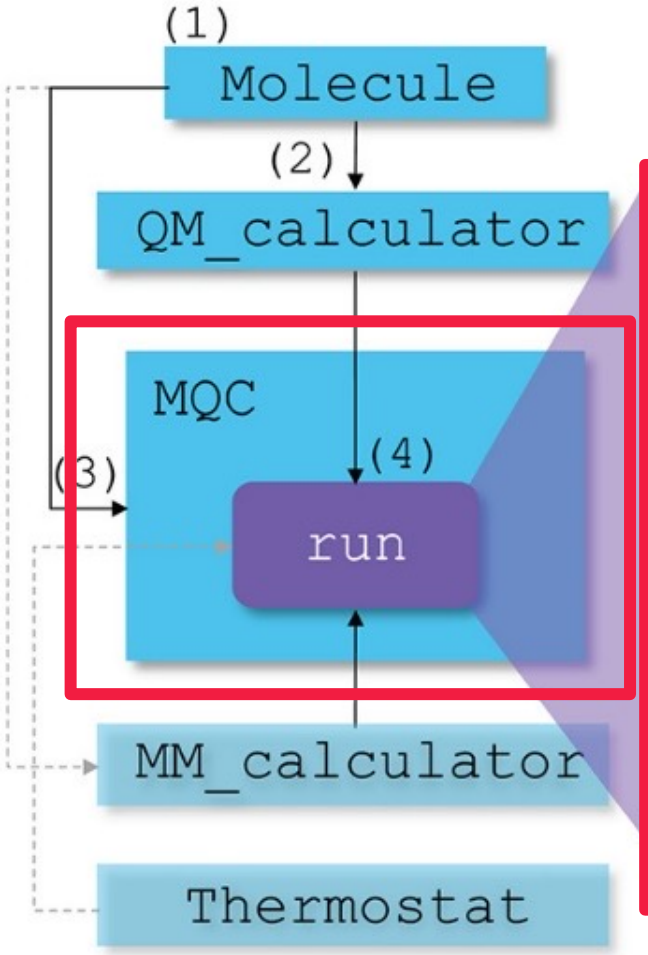
Structures and Flow of pyUNIxMD



*Arrows represent th

- ”get_data” actually carries out the QM calculation based on Molecule
- One can add any interface with a new QM program by mak ing a proper “get_data” under QM_calculator
- Also, provides NACMEs from CIS coefficients for TDDFT (B) calculations from Gaussian and DFTB+

Structures and Flow of pyUNIxMD



output files

- Central class of pyUNIxMD which defines and runs NAMD simulations
  - BOMD for Born-Oppenheimer dynamics, SH for FSSH and its derivatives (EDC, IDC, …), SHXF for SHXF, CT for CTMQC
- Requires a geometry object (molecule), # of time steps (ns teps), a nuclear time step size (dt), the initial electronic stat e (istate), how to rescale after hops, …
- MQC.”run” – execution of MQC with QM calculations at ea ch time, *all the dirty jobs (see the next slide)*

***Arrows represent the flow of information**

In the *MQC.run* method, the following process is repeated until the dynamics reaches the maximum time step (*nsteps*): (a) The molecular geometry is transferred to the *QM_calculator* object. (b) QM calculation is executed, and the calculated properties are stored to the *Molecule* object. (c) Using the properties, the *run* method updates the atomic positions and the electronic properties according to the selected MQC algorithm. (d) If *Thermostat* object exists, the velocities are adjusted according to the thermostat. (e) Finally, the *run* method writes the information about the trajectory. PyUNIxMD writes the following output files during a simulation depending on the type of MQC algorithm: *RESTART.bin*, *MDENERGY*, *BOPOP*, *BOCOH*, *NACME*, *MOVIE.xyz*, *SHPROB*, and *SHSTATE* contain dynamics information at the last successful step, energies, BO populations, BO coherence, NACMEs, the nuclear trajectory, hopping probabilities, and the running state, respectively (Figure 2).

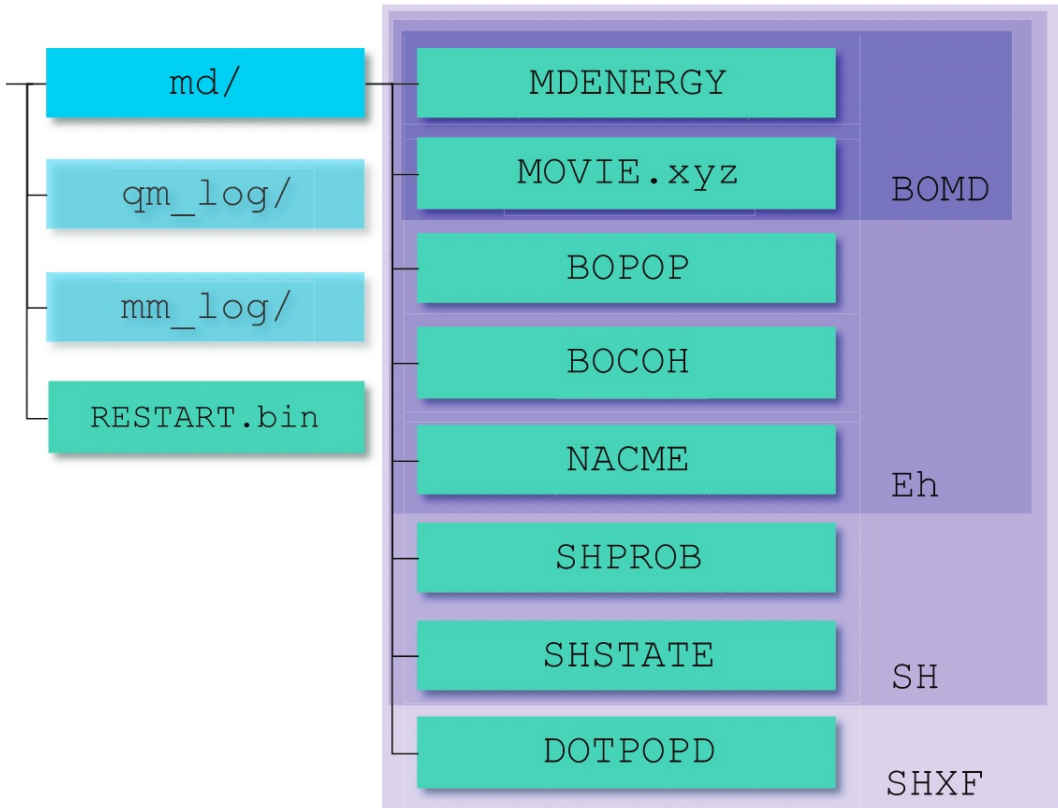
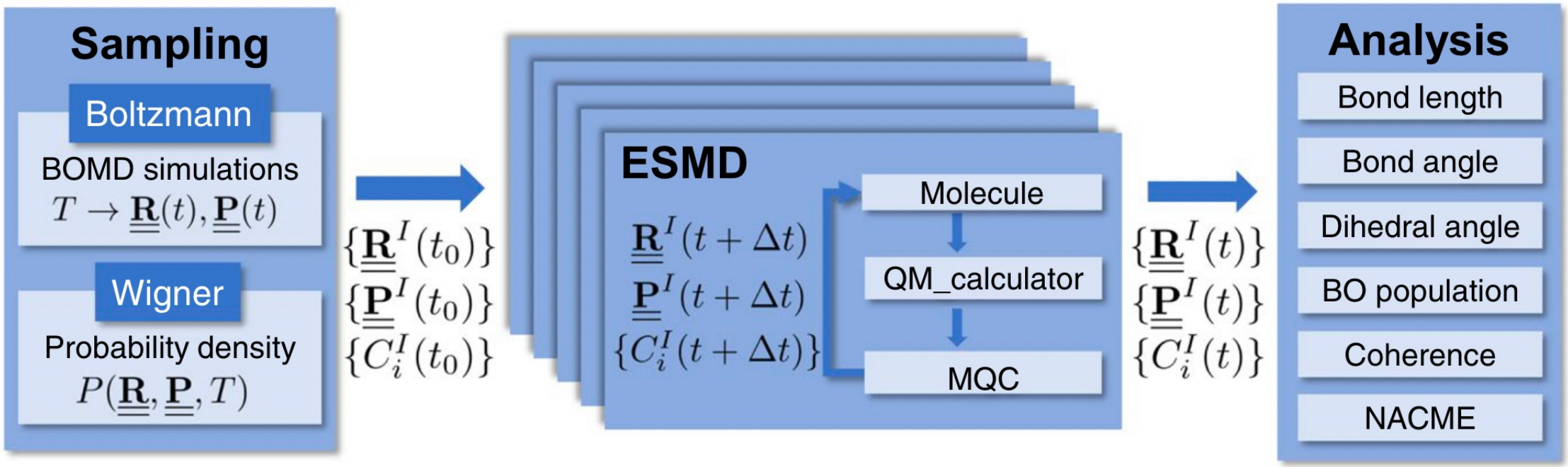


FIGURE 2 A file tree generated by PyUNIxMD. PyUNIxMD generates directories for MD outputs, logs from QM and MM calculations. The MD outputs vary according to the MQC method. The blue and light green boxes represent directories and files, respectively. The purple boxes distinguish output files that vary according to the MQC method

A sample input of pyUNIxMD

```
from molecule import Molecule
import qm, mqc, thermostat
# Initial geometry/velocity with extended XYZ format
geom = """
NUMBER_OF_ATOMS
 comment line
C        -2.8349  -1.2124  -0.2922    -0.00050  -0.0003 0.0000
…
"""
# (1) Molecule object for the target system with the above initial condition
mol = Molecule(geometry=geom, nstates=..., charge=...)
# (2) QM program/method interface for on-the-fly calculations
bo = qm.QM_PROG.QM_METHOD(molecule=mol, ...)
# (optional) Thermostat definition
bathT = thermostat.THERMO_TYPE(temperature=...,)
# (3) MD type and condtion specification
md = mqc.MD_TYPE(molecule=mol, thermostat=bathT, nsteps=..., istate=...)
# (4) Execute simulation with the dynamics condition defined in the above.
md.run(qm=bo)
```

Flow of ESMD simulations with pyUNIxMD



Interfacing with any other QC program

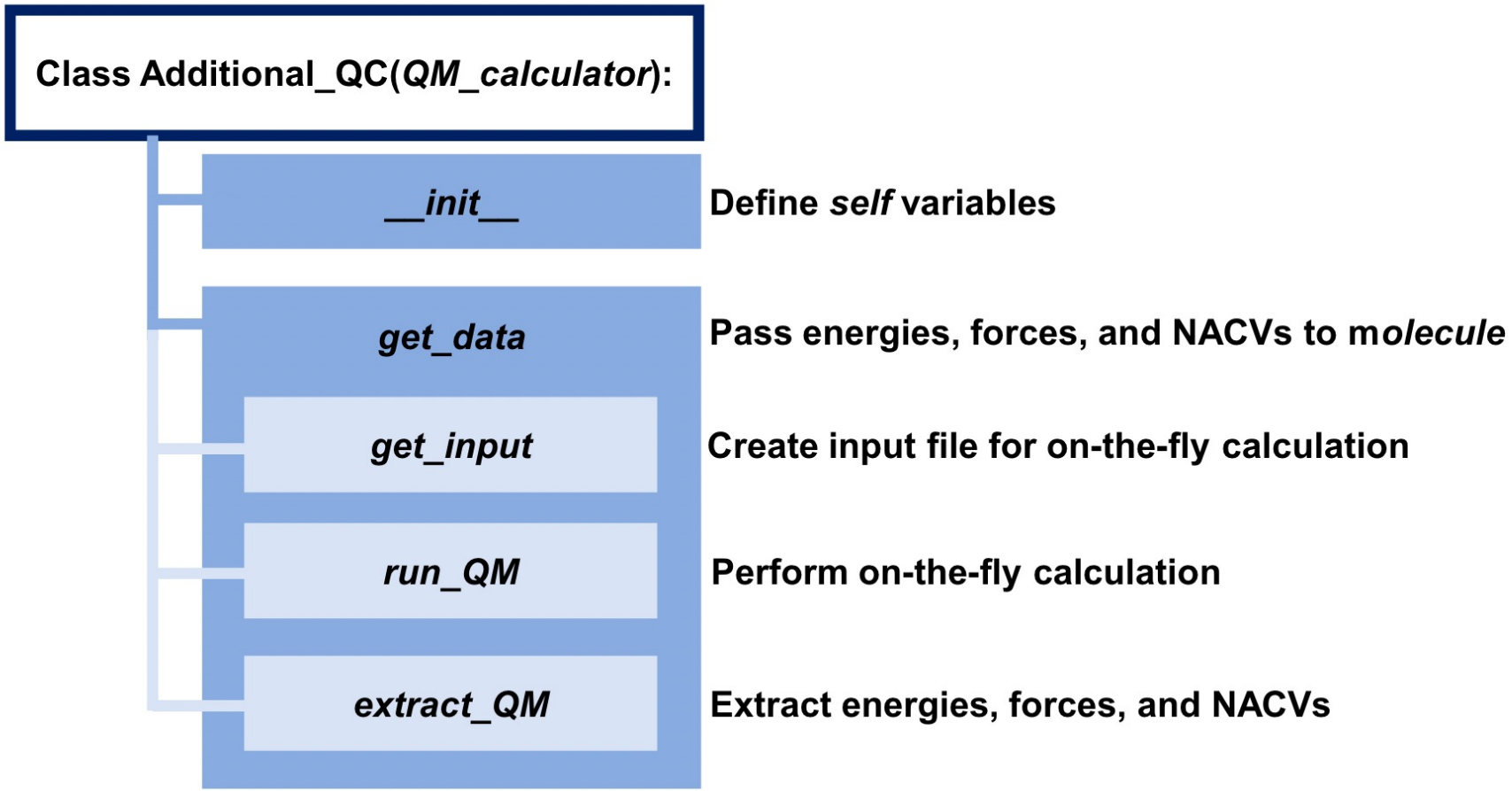


Fig. 3 A structure of additional class to the interface code with quantum chemistry program packages

Additional algorithms & improvements

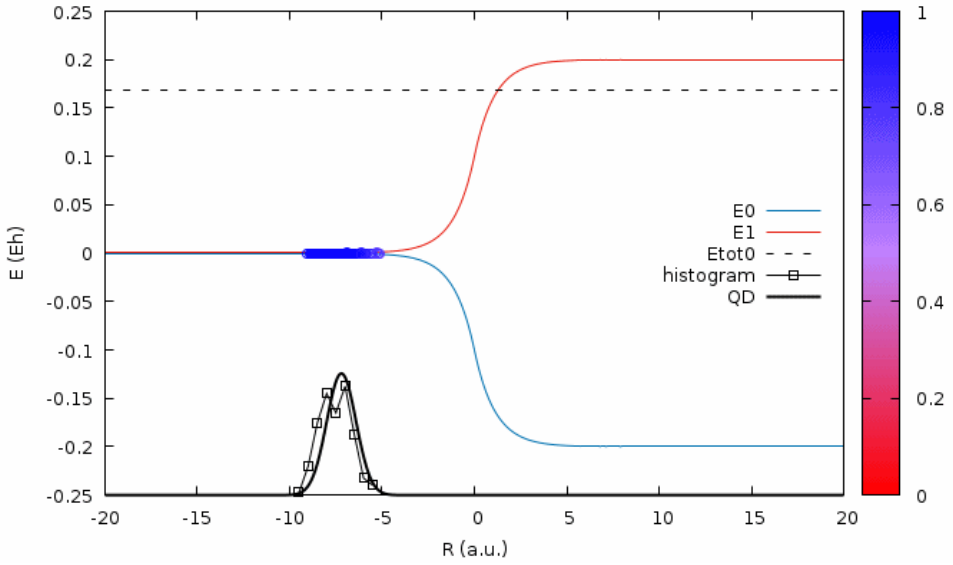
- **Ehrenfest dynamics + exact factorization (EhXF)**

$$\mathbf{F}_\nu(t) = -\sum_l \rho_{ll}(t)\nabla_\nu E_l - \sum_{l,m} \rho_{ml}(t)\,(E_l - E_m)\,\mathbf{d}_{ml,\nu}$$
$$-\sum_{l,m} \rho_{ll}(t)\rho_{mm}(t)\left[\sum_{\nu'} \frac{2\mathcal{P}_{\nu'}(t)}{\hbar M_{\nu'}} \cdot (\mathbf{f}_{m,\nu'}(t) - \mathbf{f}_{l,\nu'}(t))\right]\mathbf{f}_{l,\nu}(t)$$

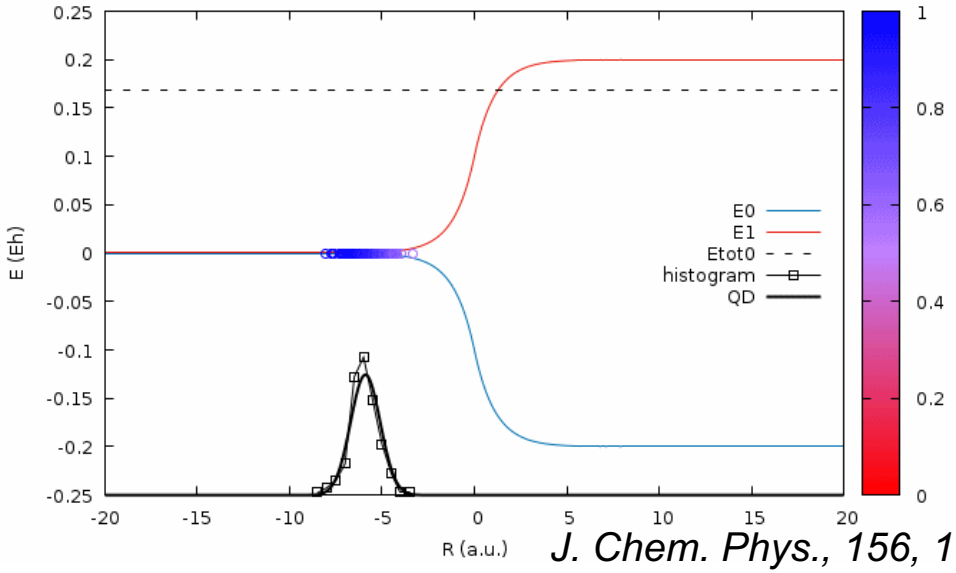
$$\dot{c}_l(t) = -\frac{i}{\hbar}E_l c_l(t) - \sum_m c_m(t)\sum_\nu \dot{\mathbf{R}}_\nu(t)\cdot\mathbf{d}_{lm,\nu}$$
$$-\sum_m \rho_{mm}(t)\left[\sum_\nu \frac{\mathcal{P}_\nu(t)}{\hbar M_\nu} \cdot (\mathbf{f}_{m,\nu}(t) - \mathbf{f}_{l,\nu}(t))\right]c_l(t)$$

- more accurate description to the exact potential energy surface

DISH-XF



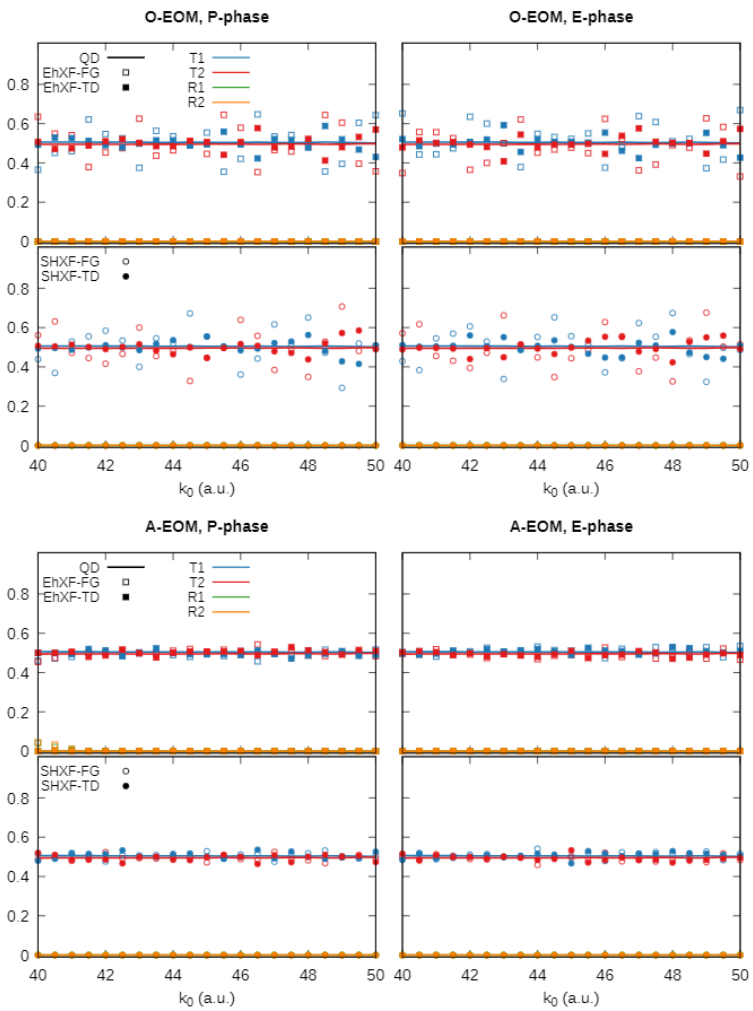
EhXF

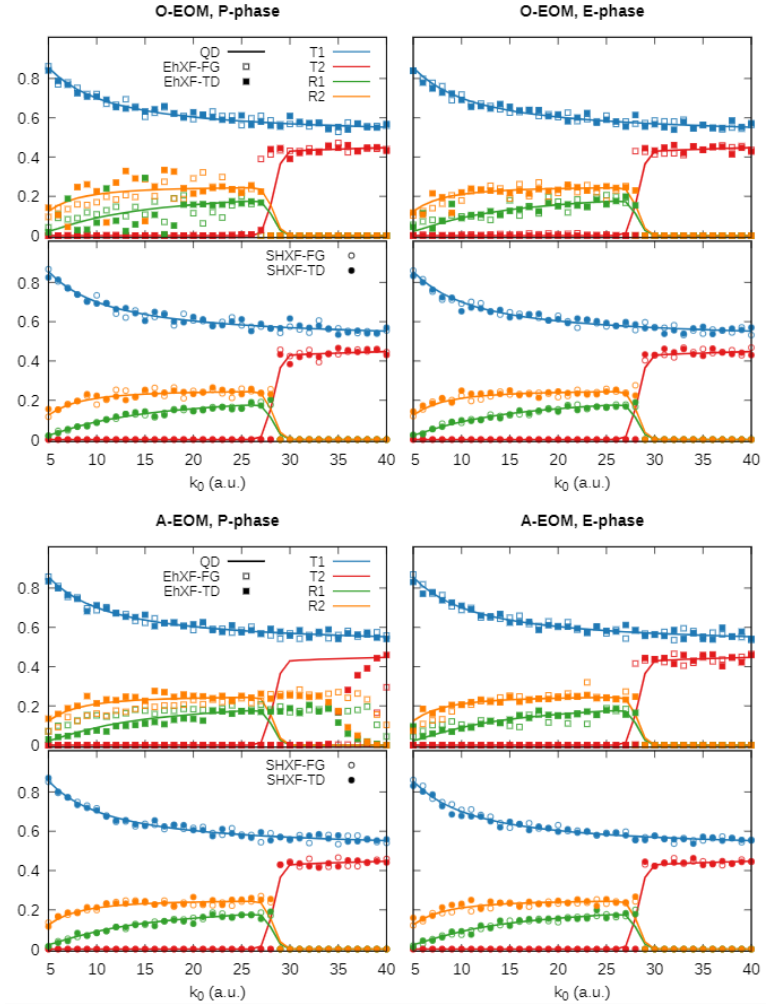


J. Chem. Phys., 156, 174109 (2022)

Additional algorithms & improvements

- **Time-dependent width of Gaussian to describe quantum momentum properly**

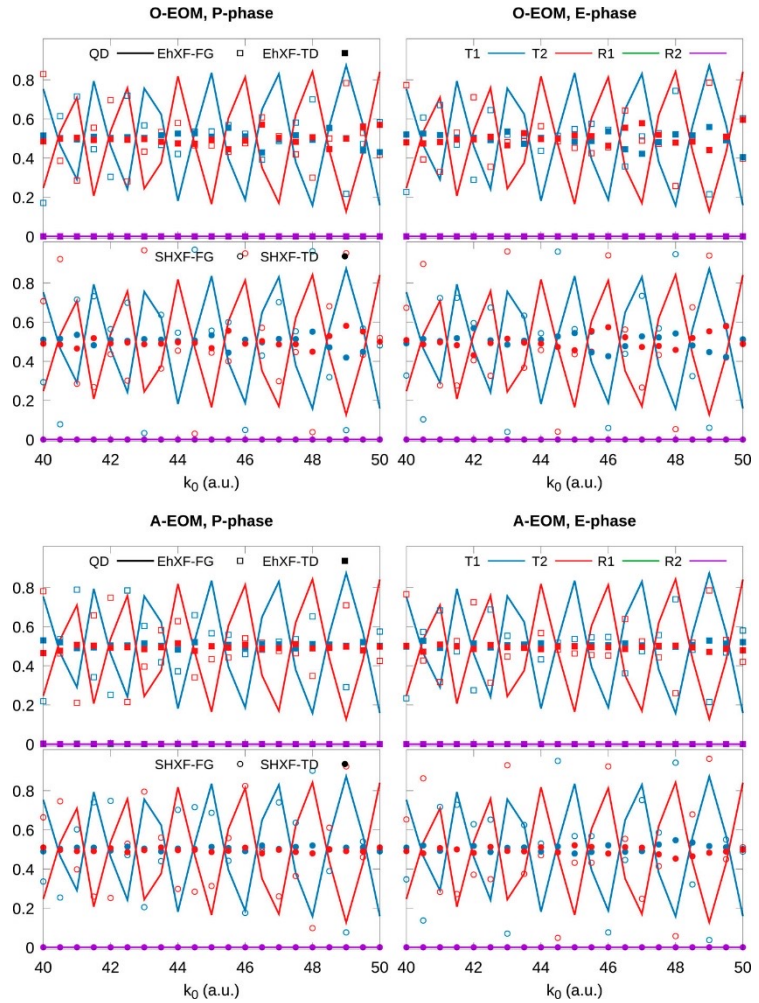


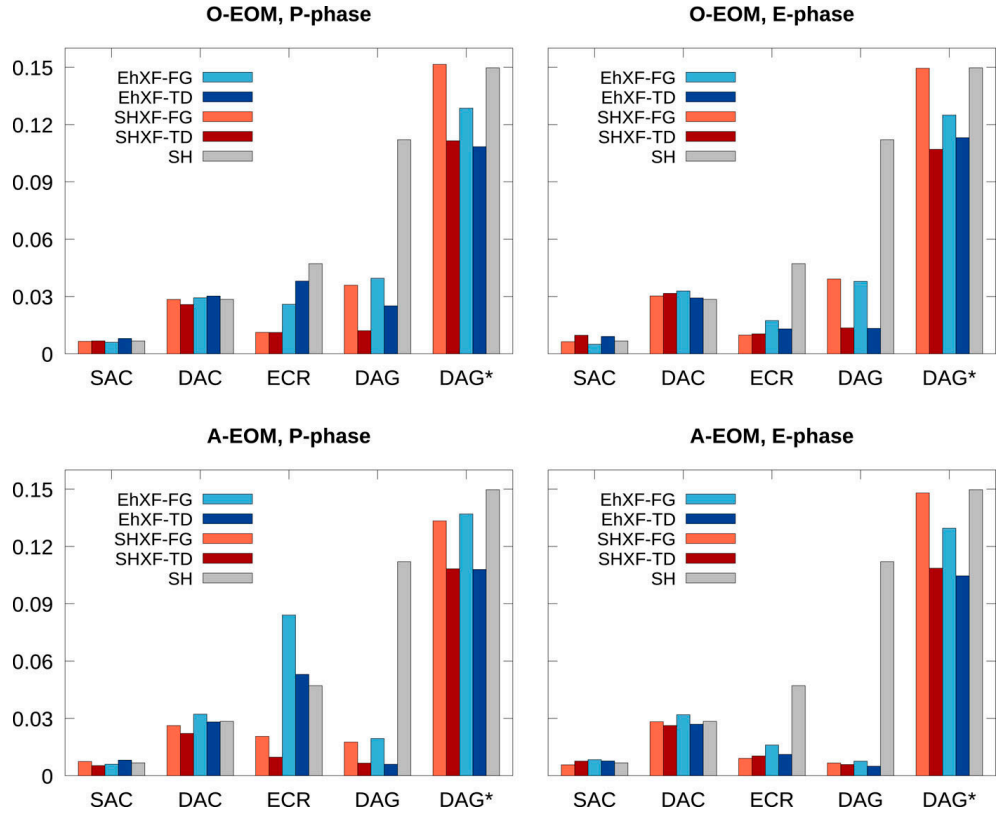


J. Chem. Phys., 156, 174109 (2022)

Failure of current approaches

- **Dynamics with wider nuclear wave packets…**
  → Stueckelberg oscillation in double-arch geometry





J. Chem. Phys., 156, 174109 (2022)

Orbital-based MQC dynamics

- Toward material calculations
  - For a large-scale system, it is extremely difficult to calculate many-body electronic states
  - Instead of many-body states, we employ the concept of orbitals to simulate MQC dynamics
  - However, the previous equations for electron-nuclear correlation should be represented in terms of orbitals → NOT straightforward

NAMD dynamics with DFT frameworks
+Decoherence corrections based on the exact factorization

D.H. Han
(UNIST)

To be submitted

Issues on the current version

- **Non-Hermitian propagator?**

$$i\hbar\frac{d}{dt}\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\underline{\mathbf{r}}},t) = \hat{H}_{BO}(\underline{\underline{\mathbf{r}}};\underline{\underline{\mathbf{R}}}^{(I)}(t))\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\underline{\mathbf{r}}},t)$$
$$+\left[i\hbar\sum_{\nu}^{N_n}\frac{\nabla_\nu|\chi|}{|\chi|}\bigg|_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}\cdot\left(\mathbf{A}_\nu^{(I)}\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\underline{\mathbf{r}}},t)+i\hbar\nabla_\nu\Phi_{\underline{\underline{\mathbf{R}}}^{(I)}(t)}(\underline{\underline{\mathbf{r}}},t)\right)\right]$$

Non-Hermitian…

However, the norm is conserving… $\dfrac{d}{dt}\displaystyle\int d\underline{\underline{\mathbf{r}}}|\Phi_{\underline{\underline{\mathbf{R}}}}(\underline{\underline{\mathbf{r}}},t)|^2 = 0$

We may find an alternative Hermitian Hamiltonian which provides the same dynamical properties…

$$\frac{d}{dt}\hat{\rho} = -\frac{i}{\hbar}\left[\hat{H}^{BO},\hat{\rho}\right]+\{\Gamma,\hat{\rho}\} \quad \longleftrightarrow \quad \frac{d}{dt}\hat{\rho} = -\frac{i}{\hbar}\left[\hat{\tilde{H}},\hat{\rho}\right]$$

Issues on the current version

- **Non-Hermitian propagator?**

Crucial for dynamics with solid states (real-time TDDFT scheme)

➔ Usual “numerical” time evolution operators only work with Hermitian operators

Finding a Hermitian operator is POSSIBLE with a simple mathematics in this scheme.

$$\frac{d}{dt}\hat{\rho} = -\frac{i}{\hbar}\left[\hat{\tilde{H}}, \hat{\rho}\right] \qquad \text{with} \qquad \hat{\tilde{H}} = \hat{H}^{BO} + \hat{Q} \circ \hat{\rho}$$

$$\text{where} \qquad (\hat{Q})_{lk} = \sum_{\nu=1}^{N_n} \frac{2}{M_\nu} \frac{\nabla_\nu |\chi|}{|\chi|}\bigg|_{(I)} \cdot \sum_{k=1}^{N_{st}} \left[\mathbf{f}_{k,\nu}^{(I)} - \mathbf{f}_{l,\nu}^{(I)}\right]$$

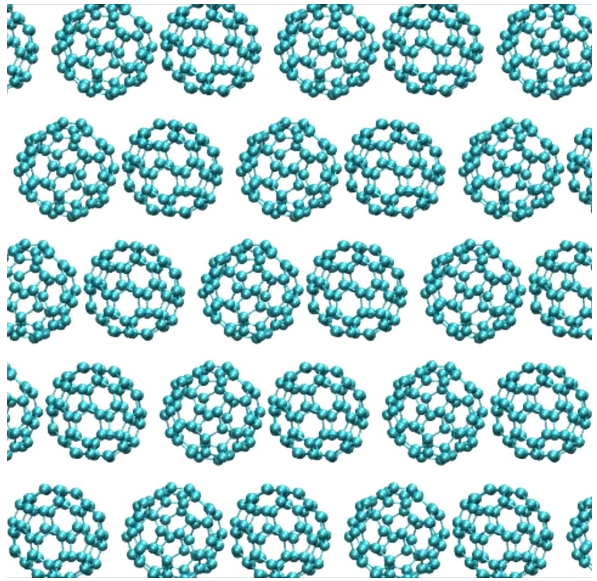
(according to BO scheme)

on-going with Quantum-Espresso

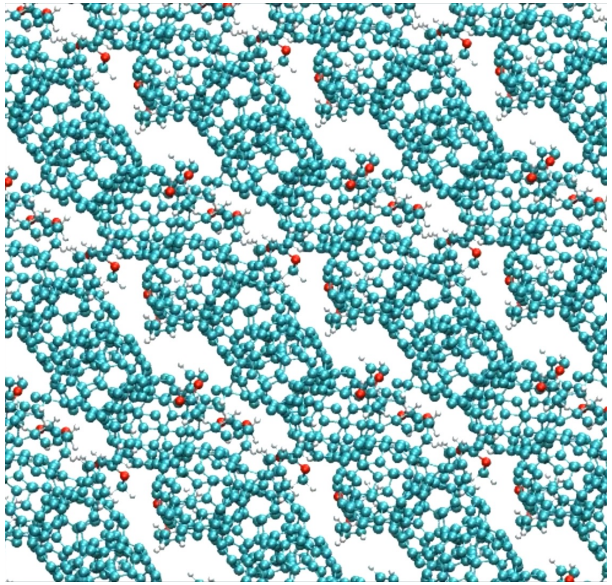
Orbital-based ESMD + α

- Toward material calculations

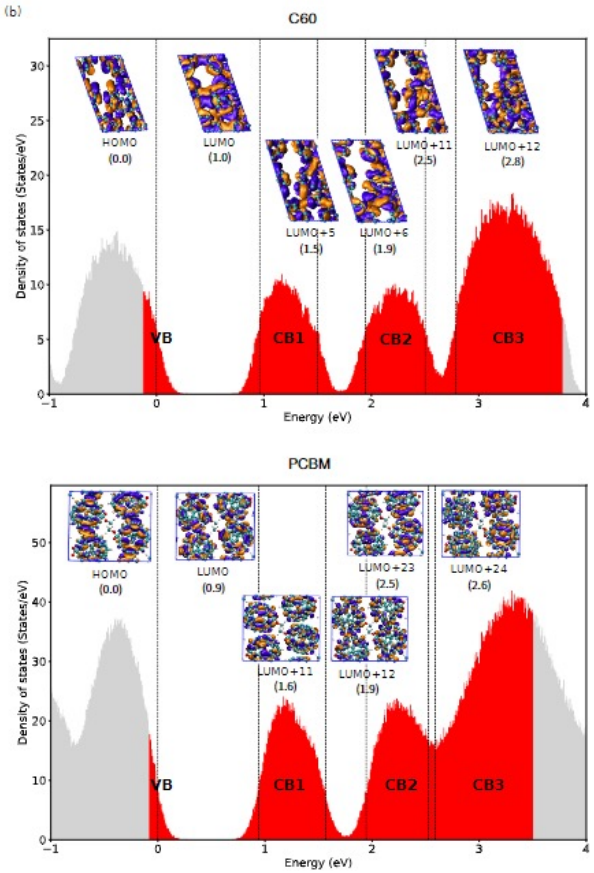
Dynamics of electron density coupled to nuclear motion
→ Charge transfer, exciton transfer, polaron dynamics, …



C_{60}



PCBM

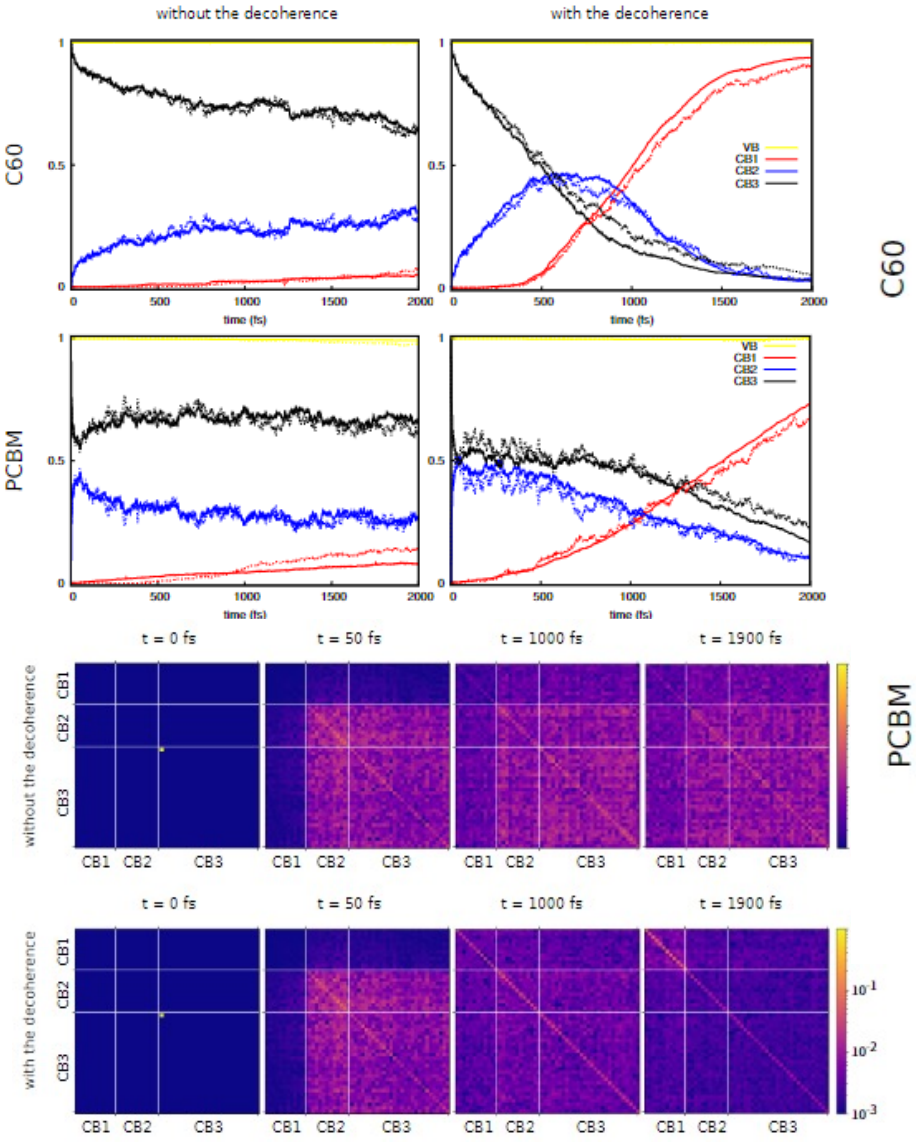


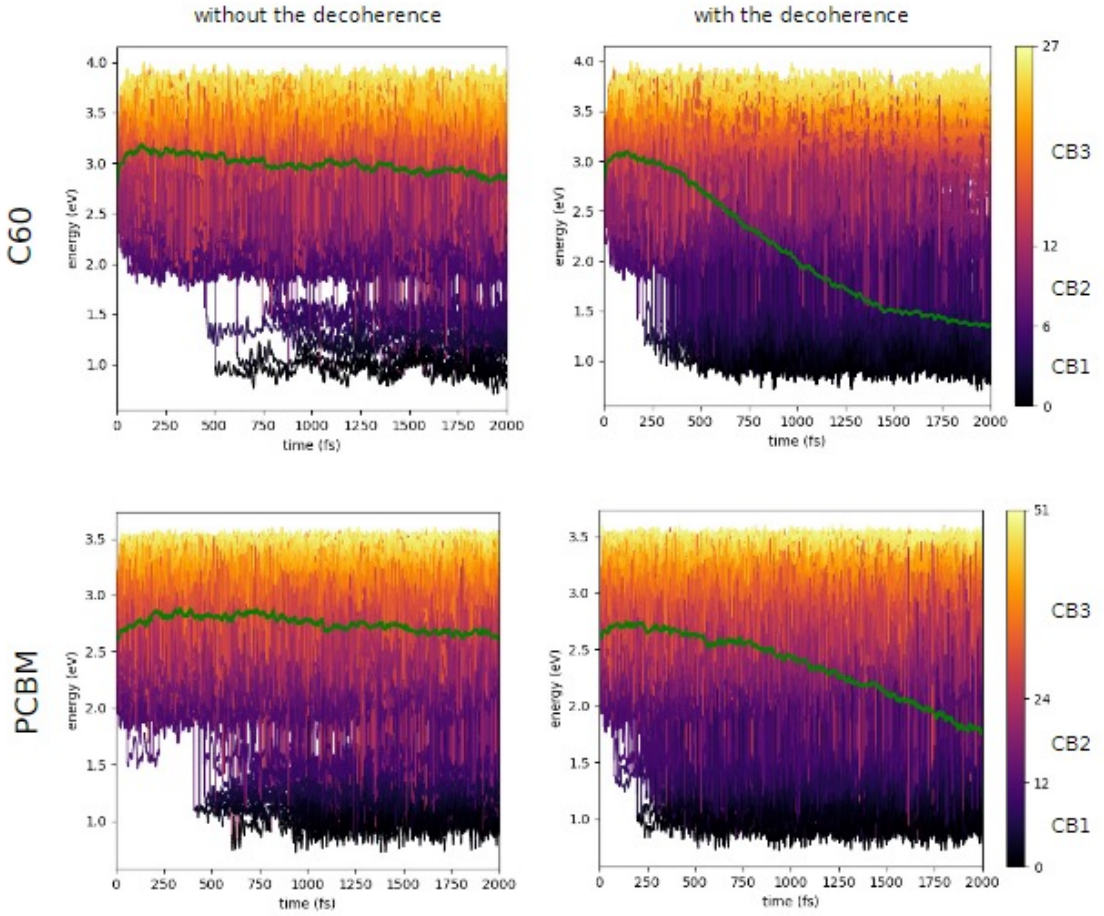
Density of states (DOS) from trajectories

Orbital-based ESMD + α

- **Toward material calculations**

population profile based on "bands"





Energy relaxation profile of electron transfer

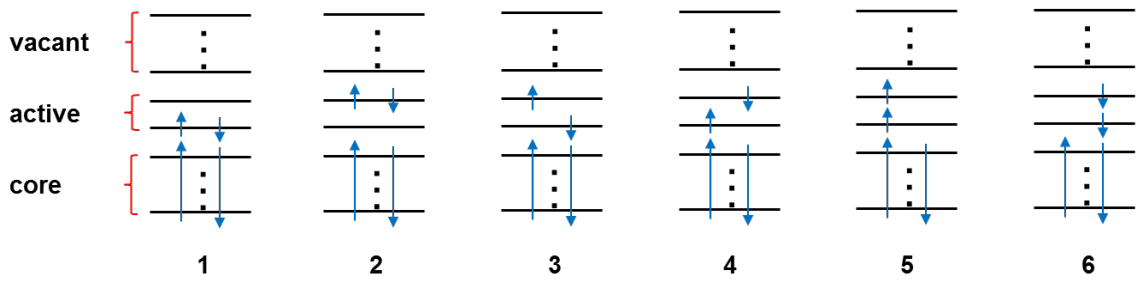
Density-Functional Tight-binding (DFTB) approach

- **Ensemble density functional theory (eDFT)**
  [Spin-Restricted Ensemble Kohn-Sham Method (REKS)] + DFTB

Efficient electronic structure calculation for accurate excited states!

I.S. Lee
(UNIST)

REKS(2,2) approach



microstates

State-interaction state-averaged REKS

$$\begin{pmatrix} E^{PPS} & \Delta^{SA} \\ \Delta^{SA} & E^{OSS} \end{pmatrix} \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix} = \begin{pmatrix} E_0^{SSR} & 0 \\ 0 & E_1^{SSR} \end{pmatrix} \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix}$$

$$E^{PPS} = \sum_{L=1}^{6} C_L^{PPS} E_L$$

$$E^{OSS} = \sum_{L=3}^{6} C_L^{OSS} E_L$$

$$\Delta^{SA} = (\sqrt{n_a} - \sqrt{n_b})\epsilon_{ab}^{SA}$$

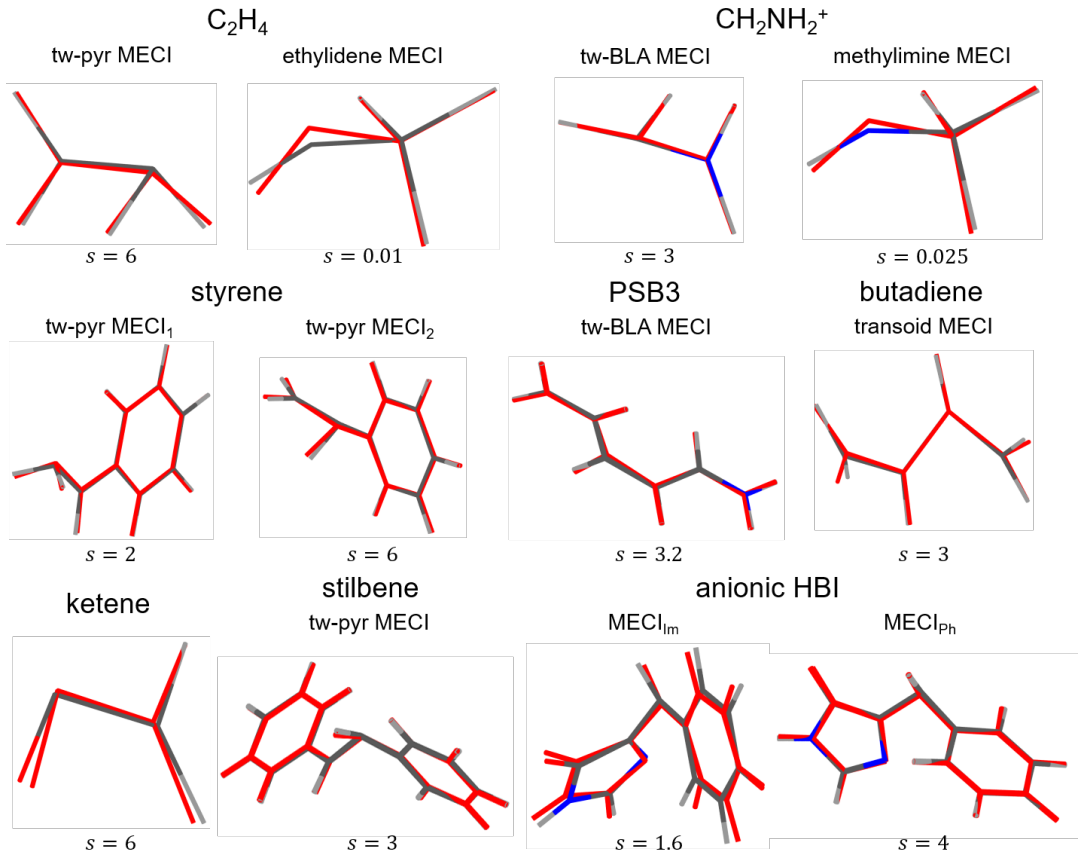
dynamic electron correlation → description of Conical Intersections (CI)

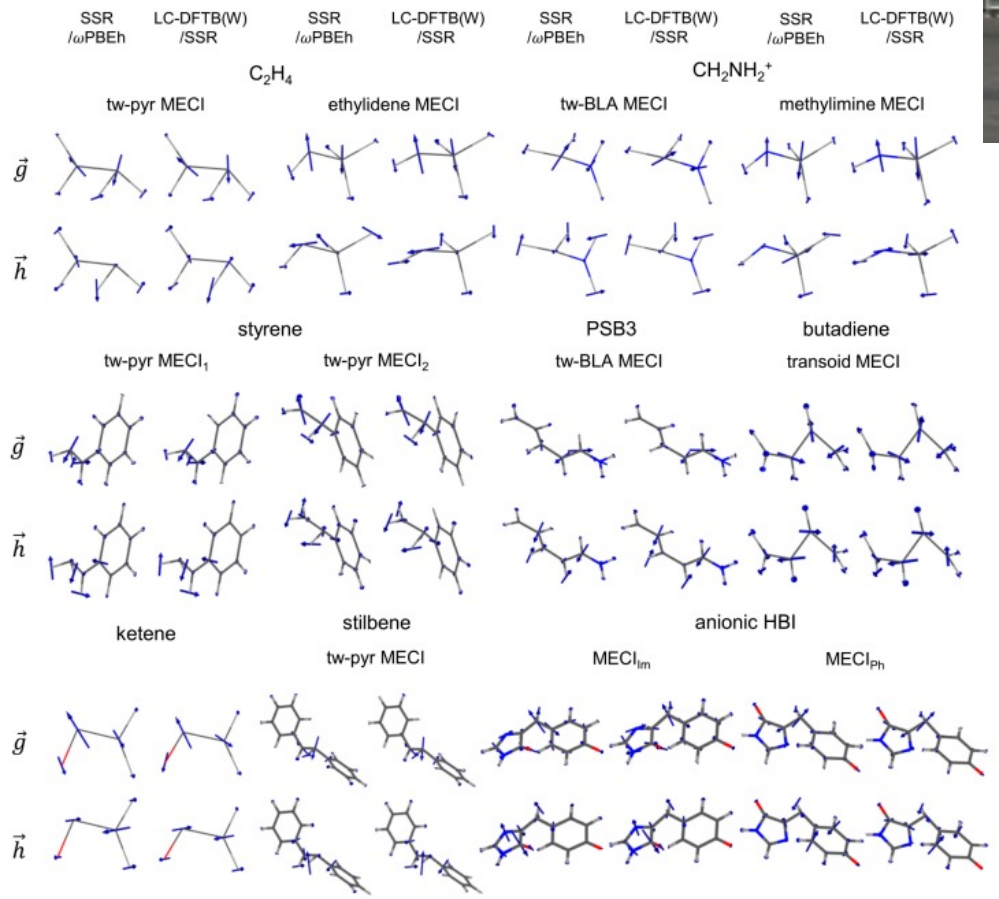
analytic gradients → minimum energy conical intersection (MECI),
excited state molecular dynamics

J. Chem. Theory Comput. 15, 3021-3032 (2019), J. Chem. Phys. 152, 124101 (2020)

Density-Functional Tight-binding (DFTB) approach

- **Benchmark**
  - **Minimum energy conical intersection (MECI) search**





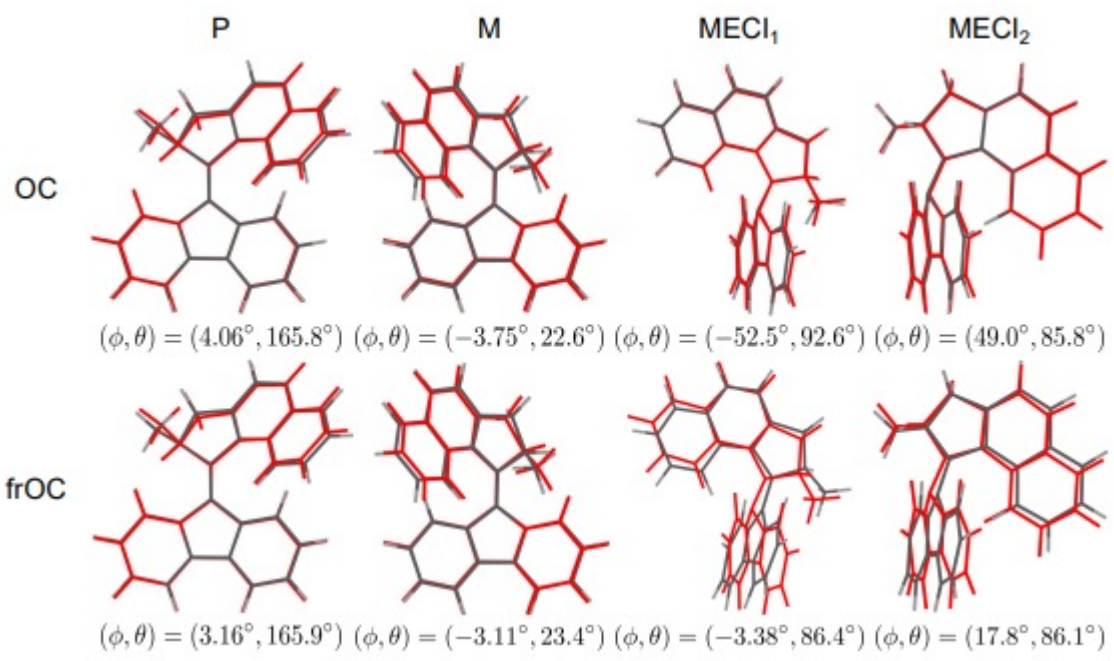
I.S. Lee
(UNIST)

J. Chem. Theory Comput. 15, 3021-3032 (2019), J. Chem. Phys. 152, 124101 (2020)

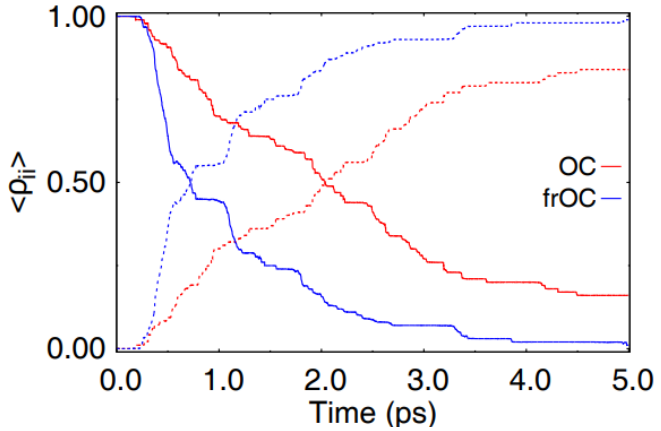
Density-Functional Tight-binding (DFTB) approach

- **Development of more generalized calculations**

inclusion of more parameters to describe electron-electron correlation (e.g. on-site atomic correction)

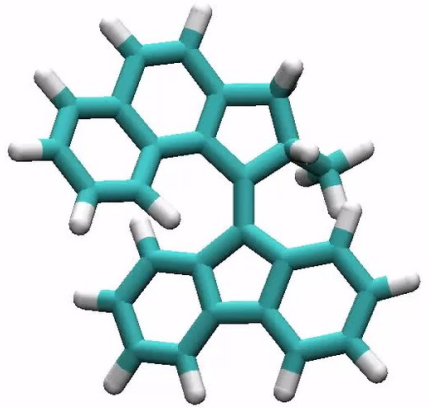


Description of correct MECI structures (pyramidalization) of Feringa's molecular motor



$S_1 \rightarrow S_0$ population transfer

I.S. Lee (UNIST)



****Computing time:**
<1min (1 core) vs. 30 min (w/ heavy GPU machine)

J. Chem. Theory Comput., in revision (2022)

Summary

- Development of excited state molecular dynamics approaches
- Dynamics : CTMQC, DISH-XF, EhXF, …
- Coming soon: more interesting features for material calculations
- pyUNIxMD program : Public (https://github.com/skmin-lab/unixmd)

Aknowledgements

Developers of pyUNIxMD

I.S. Lee
(UNIST)

J.K. Ha
(UNIST)

D.H. Han
(UNIST)

T.I. Han
(UNIST)

S.W. Moon
(UNIST)

Funding

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY 2009

NRF


UNIST
ULSAN NATIONAL INSTITUTE OF
SCIENCE AND TECHNOLOGY
2007

THANK YOU