



Automated Generator Environment in ORCA for Accurate Many-Body Theory

Hang Xu, Frank Neese

Department of Molecular Theory and Spectroscopy
Max-Planck-Institut für Kohlenforschung

September 30, 2024

Quantum Chemistry - What do we want?

Quantum chemists focus on determining the electronic structure of chemical systems with high accuracy, which enables the calculations of:

- Energies
- Molecular properties

Quantum Chemistry - What do we want?

Quantum chemists focus on determining the electronic structure of chemical systems with high accuracy, which enables the calculations of:

- Energies
- Molecular properties

What are the current goals?

- To tackle systems that were not accessible before (like heavy-metal complexes, large biomolecules, etc.)
- To achieve expected accuracy (chemical accuracy $\sim 1\text{kcal/mol}$)
- To calculate (approximate) numbers faster (lower scaling)
- ...

N -Particle Space and Wavefunction

Suppose we have N particles with labels $1, 2, 3, \dots, n$ and their respective single-particle Hilbert space:

$$\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \dots, \mathcal{H}_n \quad (1)$$

N -Particle Space and Wavefunction

Suppose we have N particles with labels $1, 2, 3, \dots, n$ and their respective single-particle Hilbert space:

$$\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \dots, \mathcal{H}_n \quad (1)$$

Then the N -particle Hilbert space and respective wavefunction (Slater Determinant) look like:

$$\mathcal{H}^N = \mathcal{A}(\underbrace{\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3 \otimes \dots \otimes \mathcal{H}_n}_{N\text{particles}}) \quad (2)$$

$$\Psi(x_1, x_2, \dots, x_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \psi_1(x_1) & \psi_2(x_1) & \dots & \psi_N(x_1) \\ \psi_1(x_2) & \psi_2(x_2) & \dots & \psi_N(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(x_N) & \psi_2(x_N) & \dots & \psi_N(x_N) \end{vmatrix} \quad (3)$$

N-Particle Space and Wavefunction

Suppose we have N particles with labels $1, 2, 3, \dots, n$ and their respective single-particle Hilbert space:

$$\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \dots, \mathcal{H}_n \quad (1)$$

Then the N -particle Hilbert space and respective wavefunction (Slater Determinant) look like:

$$\mathcal{H}^N = \underbrace{\mathcal{A}(\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3 \otimes \dots \otimes \mathcal{H}_n)}_{N\text{particles}} \quad (2)$$

$$\Psi(x_1, x_2, \dots, x_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \psi_1(x_1) & \psi_2(x_1) & \dots & \psi_N(x_1) \\ \psi_1(x_2) & \psi_2(x_2) & \dots & \psi_N(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(x_N) & \psi_2(x_N) & \dots & \psi_N(x_N) \end{vmatrix} \quad (3)$$

and the N -particle Fock space and the wavefunction (Occupation Number State, or Fock State):

$$\mathcal{F}^N(\mathcal{H}) = \bigoplus_{i=0}^N \mathcal{H}^i = \mathcal{H}^0 \oplus \mathcal{H}^1 \oplus \mathcal{H}^2 \oplus \dots \oplus \mathcal{H}^N \quad (4)$$

$$|\Psi\rangle = \underbrace{|1, 1, 1, \dots, 0, 0, 0, \dots\rangle}_{\text{Noccupied}} = \hat{a}_{i_1}^\dagger \hat{a}_{i_2}^\dagger \dots \hat{a}_{i_n}^\dagger |0\rangle \quad (5)$$

The Hamiltonian

The non-relativistic electronic Hamiltonian under Bonn-Oppenheimer (BO) approximation:

$$\hat{H}_{\text{elec}} = \underbrace{-\sum_i^{N_e} \frac{1}{2} \nabla_i^2}_{\hat{T}_e} + \underbrace{\sum_{i,j>i}^{N_e} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}}_{\hat{V}_{ee}} - \underbrace{\sum_i^{N_e} \sum_A^{N_N} \frac{Z_A}{|\mathbf{r}_i - \mathbf{R}_A|}}_{\hat{V}_{eN}} + \underbrace{\sum_{A,B>A}^{N_N} \frac{Z_A Z_B}{|\mathbf{R}_A - \mathbf{R}_B|}}_{\hat{V}_{NN}} \quad (6)$$

The Hamiltonian

The non-relativistic electronic Hamiltonian under Bonn-Oppenheimer (BO) approximation:

$$\hat{H}_{\text{elec}} = \underbrace{-\sum_i^{N_e} \frac{1}{2} \nabla_i^2}_{\hat{T}_e} + \underbrace{\sum_{i,j>i}^{N_e} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}}_{\hat{V}_{ee}} - \underbrace{\sum_i^{N_e} \sum_A^{N_N} \frac{Z_A}{|\mathbf{r}_i - \mathbf{R}_A|}}_{\hat{V}_{eN}} + \underbrace{\sum_{A,B>A}^{N_N} \frac{Z_A Z_B}{|\mathbf{R}_A - \mathbf{R}_B|}}_{\hat{V}_{NN}} \quad (6)$$

Using second quantization and neglecting the nuclear-nuclear repulsion term, it could be written as:

$$\hat{H}_{\text{elec}} = \sum_{pq} h_{pq} \hat{a}_p^\dagger \hat{a}_q + \frac{1}{2} \sum_{p,q,r,s} \langle pq|rs \rangle \hat{a}_p^\dagger \hat{a}_q^\dagger \hat{a}_s \hat{a}_r \quad (7)$$

The Hamiltonian

The non-relativistic electronic Hamiltonian under Bonn-Oppenheimer (BO) approximation:

$$\hat{H}_{\text{elec}} = \underbrace{-\sum_i^{N_e} \frac{1}{2} \nabla_i^2}_{\hat{T}_e} + \underbrace{\sum_{i,j>i}^{N_e} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}}_{\hat{V}_{ee}} - \underbrace{\sum_i^{N_e} \sum_A^{N_N} \frac{Z_A}{|\mathbf{r}_i - \mathbf{R}_A|}}_{\hat{V}_{eN}} + \underbrace{\sum_{A,B>A}^{N_N} \frac{Z_A Z_B}{|\mathbf{R}_A - \mathbf{R}_B|}}_{\hat{V}_{NN}} \quad (6)$$

Using second quantization and neglecting the nuclear-nuclear repulsion term, it could be written as:

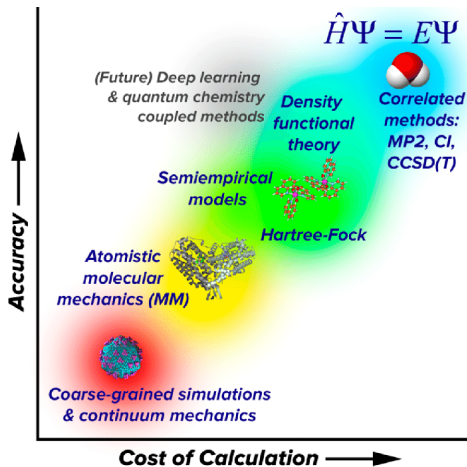
$$\hat{H}_{\text{elec}} = \sum_{pq} h_{pq} \hat{a}_p^\dagger \hat{a}_q + \frac{1}{2} \sum_{p,q,r,s} \langle pq|rs \rangle \hat{a}_p^\dagger \hat{a}_q^\dagger \hat{a}_s \hat{a}_r \quad (7)$$

with:

$$h_{pq} = \langle p|\hat{h}|q \rangle = \int \psi_p^*(\mathbf{r}) \left(-\frac{1}{2} \nabla^2 - \sum_A^{N_N} \frac{Z_A}{|\mathbf{r} - \mathbf{R}_A|} \right) \psi_q(\mathbf{r}) d\mathbf{r} \quad (8)$$

$$\langle pq|rs \rangle = \iint \frac{\psi_p^*(\mathbf{r}_1) \psi_q^*(\mathbf{r}_2) \psi_r(\mathbf{r}_1) \psi_s(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 \quad (9)$$

Overview of Quantum Chemistry Methods



1

¹Borges, Ricardo, Et al. (2021). Focus Review. Chemical Reviews

Starting Point: Hartree-Fock Method

- N -electron wavefunction: single Slater determinant $|\Phi_{\text{HF}}\rangle = |1, 1, 1, \dots, 0, 0, 0, \dots\rangle$

Starting Point: Hartree-Fock Method

- N -electron wavefunction: single Slater determinant $|\Phi_{\text{HF}}\rangle = |1, 1, 1, \dots, 0, 0, 0, \dots\rangle$
- The molecular orbitals ψ_i are solved by minimizing the total electronic energy

Starting Point: Hartree-Fock Method

- N -electron wavefunction: single Slater determinant $|\Phi_{\text{HF}}\rangle = |1, 1, 1, \dots, 0, 0, 0, \dots\rangle$
- The molecular orbitals ψ_i are solved by minimizing the total electronic energy
- Eigenvalue problem

$$\hat{F}\psi_i = \epsilon_i\psi_i \quad (10)$$

Starting Point: Hartree-Fock Method

- N -electron wavefunction: single Slater determinant $|\Phi_{\text{HF}}\rangle = |1, 1, 1, \dots, 0, 0, 0, \dots\rangle$
- The molecular orbitals ψ_i are solved by minimizing the total electronic energy
- Eigenvalue problem

$$\hat{F}\psi_i = \epsilon_i\psi_i \quad (10)$$

- The Fock operator \hat{F} depends on ψ_i , making it a Self-Consistent Field (SCF) problem:

$$\hat{F} = \hat{h} + \sum_j (2\hat{J}_j - \hat{K}_j) \quad (11)$$

in which the Coulomb and exchanged operators are defined as:

$$\hat{J}_j\psi_i(\mathbf{r}) = \int \frac{|\psi_j(\mathbf{r}')|^2}{|\mathbf{r} - \mathbf{r}'|} \psi_i(\mathbf{r}) d\mathbf{r}' \quad \hat{K}_j\psi_i(\mathbf{r}) = \int \frac{\psi_j^*(\mathbf{r})\psi_i(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \psi_j(\mathbf{r}) d\mathbf{r}' \quad (12)$$

Starting Point: Hartree-Fock Method

- N -electron wavefunction: single Slater determinant $|\Phi_{\text{HF}}\rangle = |1, 1, 1, \dots, 0, 0, 0, \dots\rangle$
- The molecular orbitals ψ_i are solved by minimizing the total electronic energy
- Eigenvalue problem

$$\hat{F}\psi_i = \epsilon_i\psi_i \quad (10)$$

- The Fock operator \hat{F} depends on ψ_i , making it a Self-Consistent Field (SCF) problem:

$$\hat{F} = \hat{h} + \sum_j (2\hat{J}_j - \hat{K}_j) \quad (11)$$

in which the Coulomb and exchanged operators are defined as:

$$\hat{J}_j\psi_i(\mathbf{r}) = \int \frac{|\psi_j(\mathbf{r}')|^2}{|\mathbf{r} - \mathbf{r}'|} \psi_i(\mathbf{r}) d\mathbf{r}' \quad \hat{K}_j\psi_i(\mathbf{r}) = \int \frac{\psi_j^*(\mathbf{r})\psi_i(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \psi_j(\mathbf{r}) d\mathbf{r}' \quad (12)$$

- Once converged, the total energy E_{HF} could be evaluated via:

$$E_{\text{HF}} = \sum_i \langle \psi_i | \hat{h} | \psi_i \rangle + \sum_{i,j} (2\langle \psi_i \psi_j | \psi_i \psi_j \rangle - \langle \psi_i \psi_j | \psi_j \psi_i \rangle) \quad (13)$$

The Wavefunction: Coupled Cluster Ansatz

The coupled-cluster wavefunction

$$|\Psi_{CC}\rangle = e^{\hat{T}}|\Phi_{HF}\rangle = e^{\hat{T}}|1, 1, 1, \dots, 0, 0, 0, \dots\rangle \quad (14)$$

The Wavefunction: Coupled Cluster Ansatz

The coupled-cluster wavefunction

$$|\Psi_{\text{CC}}\rangle = e^{\hat{T}}|\Phi_{\text{HF}}\rangle = e^{\hat{T}}|1, 1, 1, \dots, 0, 0, 0, \dots\rangle \quad (14)$$

in which the coupled-cluster excitation operators are defined as:

$$\hat{T} = \sum_m \hat{T}_m = \sum_m \frac{1}{(m!)^2} \sum_{\substack{ijk\dots \\ abc\dots}} t_{ijk\dots}^{abc\dots} \{ \hat{a}_a^\dagger \hat{a}_i \hat{a}_b^\dagger \hat{a}_j \hat{a}_c^\dagger \hat{a}_k \dots \} \quad (15)$$

The Wavefunction: Coupled Cluster Ansatz

The coupled-cluster wavefunction

$$|\Psi_{CC}\rangle = e^{\hat{T}}|\Phi_{HF}\rangle = e^{\hat{T}}|1, 1, 1, \dots, 0, 0, 0, \dots\rangle \quad (14)$$

in which the coupled-cluster excitation operators are defined as:

$$\hat{T} = \sum_m \hat{T}_m = \sum_m \frac{1}{(m!)^2} \sum_{\substack{ijk\dots \\ abc\dots}} t_{ijk\dots}^{abc\dots} \{ \hat{a}_a^\dagger \hat{a}_i \hat{a}_b^\dagger \hat{a}_j \hat{a}_c^\dagger \hat{a}_k \dots \} \quad (15)$$

Schrödinger Equation:

$$\hat{H}_{\text{elec}}|\Psi_{CC}\rangle = E_{CC}|\Psi_{CC}\rangle \quad (16)$$

The Wavefunction: Coupled Cluster Ansatz

The coupled-cluster wavefunction

$$|\Psi_{CC}\rangle = e^{\hat{T}}|\Phi_{HF}\rangle = e^{\hat{T}}|1, 1, 1, \dots, 0, 0, 0, \dots\rangle \quad (14)$$

in which the coupled-cluster excitation operators are defined as:

$$\hat{T} = \sum_m \hat{T}_m = \sum_m \frac{1}{(m!)^2} \sum_{\substack{ijk\dots \\ abc\dots}} t_{ijk\dots}^{abc\dots} \{ \hat{a}_a^\dagger \hat{a}_i \hat{a}_b^\dagger \hat{a}_j \hat{a}_c^\dagger \hat{a}_k \dots \} \quad (15)$$

Schrödinger Equation:

$$\hat{H}_{\text{elec}}|\Psi_{CC}\rangle = E_{CC}|\Psi_{CC}\rangle \quad (16)$$

or (via the "effective" Hamiltonian):

$$\bar{H}|\Phi_{HF}\rangle = e^{-\hat{T}}\hat{H}_{\text{elec}}e^{\hat{T}}|\Phi_{HF}\rangle = E_{CC}|\Phi_{HF}\rangle \quad (17)$$

The Wavefunction: Commutator Expression

The effective Hamiltonian could be expanded with the Baker-Campbell-Hausdorff (BCH) formula which truncates naturally here:

$$\bar{H} = e^{-\hat{T}} \hat{H} e^{\hat{T}} = \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \quad (18)$$

The Wavefunction: Commutator Expression

The effective Hamiltonian could be expanded with the Baker-Campbell-Hausdorff (BCH) formula which truncates naturally here:

$$\bar{H} = e^{-\hat{T}} \hat{H} e^{\hat{T}} = \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \quad (18)$$

By projecting the Schrödinger equation onto the ground state we obtain the energy in closed-form:

$$E_{CC} = \langle \Phi_{HF} | \bar{H} | \Phi_{HF} \rangle = \langle \Phi_{HF} | e^{-\hat{T}} \hat{H}_{elec} e^{\hat{T}} | \Phi_{HF} \rangle = \langle \Phi_{HF} | \hat{H} + [\hat{H}, \hat{T}] + \dots | \Phi_{HF} \rangle \quad (19)$$

The Wavefunction: Commutator Expression

The effective Hamiltonian could be expanded with the Baker-Campbell-Hausdorff (BCH) formula which truncates naturally here:

$$\bar{H} = e^{-\hat{T}} \hat{H} e^{\hat{T}} = \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \quad (18)$$

By projecting the Schrödinger equation onto the ground state we obtain the energy in closed-form:

$$E_{CC} = \langle \Phi_{HF} | \bar{H} | \Phi_{HF} \rangle = \langle \Phi_{HF} | e^{-\hat{T}} \hat{H}_{elec} e^{\hat{T}} | \Phi_{HF} \rangle = \langle \Phi_{HF} | \hat{H} + [\hat{H}, \hat{T}] + \dots | \Phi_{HF} \rangle \quad (19)$$

projecting onto the μ -body excitation manifold we can solve for the corresponding CC amplitude t_μ :

$$0 = \langle \Phi_\mu | \bar{H} | \Phi_{HF} \rangle = \langle \Phi_\mu | e^{-\hat{T}} \hat{H}_{elec} e^{\hat{T}} | \Phi_{HF} \rangle = \langle \Phi_\mu | \hat{H} + [\hat{H}, \hat{T}] + \dots | \Phi_{HF} \rangle \quad (20)$$

Number of Diagrams for Coupled Cluster Theory

$$\begin{aligned}
 0 &= \text{Diagram 1} + \text{Diagram 2} + \text{Diagram 3} \\
 &+ \text{Diagram 4} + \text{Diagram 5} + \text{Diagram 6} \\
 &+ \text{Diagram 7} + \text{Diagram 8} \\
 &+ \text{Diagram 9} + \text{Diagram 10}
 \end{aligned}$$

Theory	Time (s)	Diagrams per excitation level									
		0	1	2	3	4	5	6	7	8	
CCSD	0.1	3	14	31							
CCSDT	0.7	3	15	37	47						
CCSDTQ	2.4	3	15	38	53	74					
CCSDTQP	6.3	3	15	38	54	80	99				
CCSDTQPH	13.8	3	15	38	54	81	105	135			
CCSDTQPH7	26.0	3	15	38	54	81	106	141	169		
CCSDTQPH78	45.4	3	15	38	54	81	106	142	175	215	

2

Equation Generator: The Commutator Approach

In ORCA_AGE we use the single-particle replacement operator:

$$\hat{E}_q^p = \hat{a}_{p\alpha}^\dagger \hat{q}_{q\alpha} + \hat{a}_{p\beta}^\dagger \hat{q}_{q\beta} \quad (21)$$

Equation Generator: The Commutator Approach

In ORCA_AGE we use the single-particle replacement operator:

$$\hat{E}_q^p = \hat{a}_{p\alpha}^\dagger \hat{q}_{q\alpha} + \hat{a}_{p\beta}^\dagger \hat{q}_{q\beta} \quad (21)$$

We can arrange any target into this form:

$$\gamma_{uv\dots z}^{mn\dots t} = \langle \Phi_{\text{HF}} | \hat{E}_u^m \hat{E}_v^n \dots \hat{E}_z^t | \Phi_{\text{HF}} \rangle \quad (22)$$

Equation Generator: The Commutator Approach

In ORCA_AGE we use the single-particle replacement operator:

$$\hat{E}_q^p = \hat{a}_{p\alpha}^\dagger \hat{q}_{q\alpha} + \hat{a}_{p\beta}^\dagger \hat{q}_{q\beta} \quad (21)$$

We can arrange any target into this form:

$$\gamma_{uv\dots z}^{mn\dots t} = \langle \Phi_{\text{HF}} | \hat{E}_u^m \hat{E}_v^n \dots \hat{E}_z^t | \Phi_{\text{HF}} \rangle \quad (22)$$

and then apply these commutator relation and termination conditions for equation generation:

$$[\hat{E}_q^p, \hat{E}_s^r] = \hat{E}_s^p \delta_{rq} - \hat{E}_q^r \delta_{ps} \quad (23)$$

$$\hat{E}_p^i | \Phi_{\text{HF}} \rangle = 2\delta_{ip} | \Phi_{\text{HF}} \rangle \quad \langle \Phi_{\text{HF}} | \hat{E}_i^p = 2\delta_{ip} \langle \Phi_{\text{HF}} | \quad (24)$$

$$\hat{E}_a^p | \Phi_{\text{HF}} \rangle = 0 \quad \langle \Phi_{\text{HF}} | \hat{E}_p^a = 0 \quad (25)$$

Equation Generator: The Commutator Approach

In ORCA_AGE we use the single-particle replacement operator:

$$\hat{E}_q^p = \hat{a}_{p\alpha}^\dagger \hat{q}_{q\alpha} + \hat{a}_{p\beta}^\dagger \hat{q}_{q\beta} \quad (21)$$

We can arrange any target into this form:

$$\gamma_{uv\dots z}^{mn\dots t} = \langle \Phi_{\text{HF}} | \hat{E}_u^m \hat{E}_v^n \dots \hat{E}_z^t | \Phi_{\text{HF}} \rangle \quad (22)$$

and then apply these commutator relation and termination conditions for equation generation:

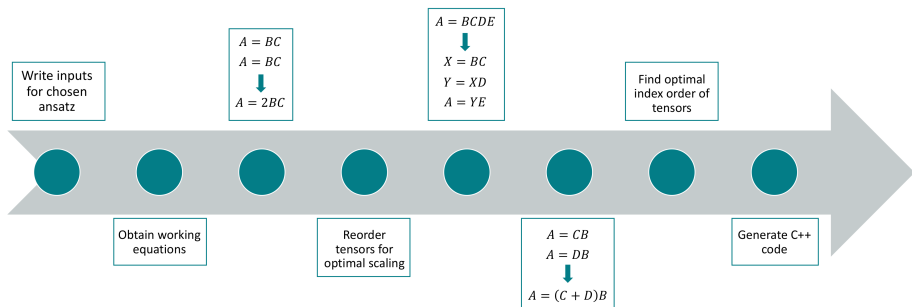
$$[\hat{E}_q^p, \hat{E}_s^r] = \hat{E}_s^p \delta_{rq} - \hat{E}_q^r \delta_{ps} \quad (23)$$

$$\hat{E}_p^i | \Phi_{\text{HF}} \rangle = 2\delta_{ip} | \Phi_{\text{HF}} \rangle \quad \langle \Phi_{\text{HF}} | \hat{E}_i^p = 2\delta_{ip} \langle \Phi_{\text{HF}} | \quad (24)$$

$$\hat{E}_a^p | \Phi_{\text{HF}} \rangle = 0 \quad \langle \Phi_{\text{HF}} | \hat{E}_p^a = 0 \quad (25)$$

- Move indices around with commutation relations
- Creates 0 and Kronecker deltas

Equation Processing: The Toolchain



- Equation Generation → Equation Processing (Factorization) → Code Generation
- Maths utilities: indices, tensors, contractions, symmetry, etc.
- Text-based symbol manipulation

An equation file contains:

- Header: containing information about tensor storage, permutational symmetry etc.
- Equations: containing actual contractions:
 - Target tensor
 - Source tensors
 - factors and summation information

An equation file contains:

- Header: containing information about tensor storage, permutational symmetry etc.
- Equations: containing actual contractions:
 - Target tensor
 - Source tensors
 - factors and summation information

Example:

$$S_{ij}^{ab} \leftarrow - \sum_{kc} (ki|bc) \tau_{kj}^{ac} \quad (26)$$

```
Sijab(a0,i0,b0,j0) += -1.0 I(K0,i0,b0,C0) Tau(a0,K0,C0,j0)
```

- Summation indices are capitalized
- Rank-4 tensors like S_{ij}^{ab} stored on disk, as a Matrix Container: $S_{[i,j]}(a,b)$

Utilizing permutational symmetry to reduce redundant terms, e.g. for the target:

$$E_{\text{CCSD}} \leftarrow \langle \Phi_{\text{HF}} | \hat{H} \hat{T}_2 | \Phi_{\text{HF}} \rangle \quad (27)$$

during the equation generation there are many redundant terms:

$$E_{\text{CCSD}} \leftarrow \sum_{ijab} (ia|jb) T_{ij}^{ab} \quad (28)$$

$$E_{\text{CCSD}} \leftarrow \sum_{ijab} (jb|ia) T_{ij}^{ab} \quad (29)$$

Utilizing permutational symmetry to reduce redundant terms, e.g. for the target:

$$E_{\text{CCSD}} \leftarrow \langle \Phi_{\text{HF}} | \hat{H} \hat{T}_2 | \Phi_{\text{HF}} \rangle \quad (27)$$

during the equation generation there are many redundant terms:

$$E_{\text{CCSD}} \leftarrow \sum_{ijab} (ia|jb) T_{ij}^{ab} \quad (28)$$

$$E_{\text{CCSD}} \leftarrow \sum_{ijab} (jb|ia) T_{ij}^{ab} \quad (29)$$

Using the symmetry relation $(ia|jb) = (jb|ia)$ we can merge these two terms into one:

$$E_{\text{CCSD}} \leftarrow 2 \sum_{ijab} (ia|jb) T_{ij}^{ab} \quad (30)$$

Given the following contractions:

$$A = BCDEF \quad (31)$$

$$X = PQCDR \quad (32)$$

$$Y = JDEFL \quad (33)$$

Given the following contractions:

$$A = BCDEF \quad (31)$$

$$X = PQCDR \quad (32)$$

$$Y = JDEFL \quad (33)$$

There might be several different ways to factorize term A :

$$A = (((BC)D)E)F \quad (34)$$

$$A = B((CD)(EF)) \quad (35)$$

$$A = (BC)(D(EF)) \quad (36)$$

$$\dots \quad (37)$$

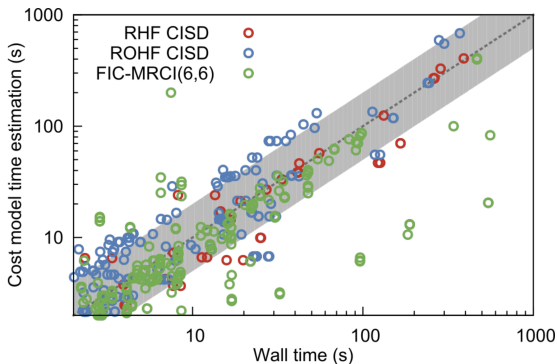
Which one shall we pick? → The global minimization problem

Cost Model

- Estimates the FLOP count for a given contraction
- Based on the index space sizes
- Determines the factorization scheme

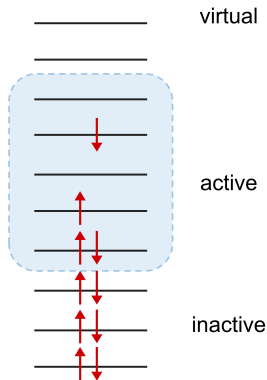
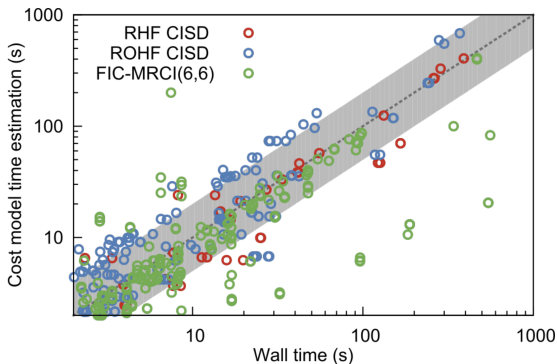
Cost Model

- Estimates the FLOP count for a given contraction
- Based on the index space sizes
- Determines the factorization scheme



Cost Model

- Estimates the FLOP count for a given contraction
- Based on the index space sizes
- Determines the factorization scheme



Cost Model: An Example

Contraction:

$$E_{(T)} \leftarrow \sum_{ijkabc} t_i^a t_{ijk}^{abc} (jb|kc) \quad (38)$$

Factorization Scheme (1):

$$X_{jk}^{bc} = \sum_{ia} t_i^a t_{ijk}^{abc} \quad (39)$$

$$E_{(T)} \leftarrow \sum_{jkab} X_{jk}^{bc} (jb|kc) \quad (40)$$

$$\text{FLOP} = 6.402 \times 10^{10}$$

Factorization Scheme (2):

$$X_i^a = \sum_{jkbc} t_{ijk}^{abc} (jb|kc) \quad (41)$$

$$E_{(T)} \leftarrow \sum_{ia} t_i^a X_i^a \quad (42)$$

$$\text{FLOP} = 6.400 \times 10^{10}$$

Cost Model: An Example

Contraction:

$$E_{(T)} \leftarrow \sum_{ijkabc} t_i^a t_{ijk}^{abc} (jb|kc) \quad (38)$$

Factorization Scheme (1):

$$X_{jk}^{bc} = \sum_{ia} t_i^a t_{ijk}^{abc} \quad (39)$$

$$E_{(T)} \leftarrow \sum_{jkab} X_{jk}^{bc} (jb|kc) \quad (40)$$

$$\text{FLOP} = 6.402 \times 10^{10}$$

Factorization Scheme (2):

$$X_i^a = \sum_{jkbc} t_{ijk}^{abc} (jb|kc) \quad (41)$$

$$E_{(T)} \leftarrow \sum_{ia} t_i^a X_i^a \quad (42)$$

$$\text{FLOP} = 6.400 \times 10^{10}$$

Note that in practice, $N_{\text{virt}} \gg N_{\text{inactive}} > N_{\text{active}}$

- Summation Intermediates (distributive law):

$$S \leftarrow AC \quad (43)$$

$$S \leftarrow BC \quad (44)$$

as addition is cheaper than multiplication:

$$D \leftarrow A + B \quad (45)$$

$$S \leftarrow DB \quad (46)$$

- Summation Intermediates (distributive law):

$$S \leftarrow AC \quad (43)$$

$$S \leftarrow BC \quad (44)$$

as addition is cheaper than multiplication:

$$D \leftarrow A + B \quad (45)$$

$$S \leftarrow DB \quad (46)$$

- I/O minimization:

$$\sigma_{[i,j]}(a, b) \leftarrow X_{[i,j]}(a, b) \quad \text{vs} \quad \sigma_{[i,j]}(a, b) \leftarrow X_{[a,b]}(i, j) \quad (47)$$

Contraction Engine (I): Optimized Efficiency

- Part of the C++ code generator
- Inserts specialized hand-written function calls based on pattern matching
- Reduces I/O as much as possible
- Improves efficiency as much as possible

⇒ Load as much as possible into memory, use BLAS whenever possible

Contraction Engine (I): Optimized Efficiency

- Part of the C++ code generator
- Inserts specialized hand-written function calls based on pattern matching
- Reduces I/O as much as possible
- Improves efficiency as much as possible

⇒ Load as much as possible into memory, use BLAS whenever possible

Example:

$$C_{pq}^{rs} \leftarrow \sum_{tuv} A_{stu}^{prv} B_{tu}^{qv} \quad (48)$$

$$C_{[p,q]}(r, s) \leftarrow A_{[s,T,U]}(p, r, V) B_{[T,U]}(q, V) \quad (49)$$

Contraction Engine (II)

$$C_{[p,q]}(r, s) \leftarrow A_{[s,T,U]}(p, r, V)B_{[T,U]}(q, V) \quad (50)$$

```
loop p:  
  loop q:  
    load C_pq;  
    loop T,U:  
      load B_TU;  
      loop s:  
        load A_sTU  
        ...
```

Very slow, as well as redundant and unnecessary I/O with the p, q loop.

Contraction Engine (III)

$$C_{[p,q]}(r, s) \leftarrow A_{[s,T,U]}(p, r, V) B_{[T,U]}(q, V) \quad (51)$$

Alternatively, we define the intermediates A_T, B_T, C_T (BLAS-able and kept in memory!):

```
loop s,T,U:  
  load A_sTU;  
  loop p,r,V:  
    A_T(prs,TUV) ← A_sTU(p,r,V)
```

Load A into A_T

```
loop T,U:  
  load B_TU;  
  loop q,V:  
    B_T(q,TUV) ← B_TU(q,V)
```

Load B into B_T

```
C_T(prs,q) ← A_T(prs,TUV) B_T(q,TUV)T
```

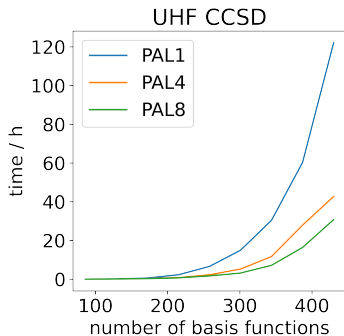
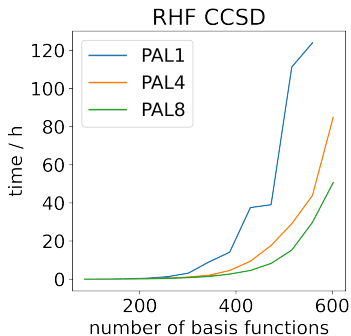
BLAS Call

```
loop p,q:  
  load C_pq;  
  loop r,s:  
    C_pq(r,s) ← C_T(prs,q)
```

Store C_T into C

- All terms applicable would invoke this! (Pattern matching)

Nuclear Gradient (Analytic)



- Maxcore: 5000 MB per core
- CPU: AMD EPYC 75F3
- Basis: def2-TZVP
- Frozen core, single gradient step

Response Properties(I)

The total electronic energy under external perturbations:

$$E(\mathbf{X}) = E_0 + \sum_i \left(\frac{\partial E}{\partial X_i} \right)_0 X_i + \frac{1}{2} \sum_{i,j} \left(\frac{\partial^2 E}{\partial X_i \partial X_j} \right)_{(0,0)} X_i X_j + \dots \quad (52)$$

where $\mathbf{X} = (\mathbf{F}, \mathbf{B}, \mathbf{I}, \dots)$ is the collection of perturbations.

We can identify various molecular properties as the derivatives in the Taylor series:

- $\frac{\partial E}{\partial F_i}$: electric dipole p_i
- $\frac{\partial^2 E}{\partial F_i \partial F_j}$: polarizability α_{ij}
- $\frac{\partial^2 E}{\partial B_i \partial B_j}$: magnetizability β_{ij}
- $\frac{\partial^2 E}{\partial B_i \partial m_{Nj}}$: NMR shielding σ_{ij}

Response Properties(II)

$$\frac{\partial^2 E}{\partial x \partial y} = \sum_{\mu\nu} D_{\mu\nu} \frac{\partial^2 h_{\mu\nu}}{\partial x \partial y} + \sum_{\mu\nu} \frac{\partial D_{\mu\nu}}{\partial x} \frac{\partial h_{\mu\nu}}{\partial y} + \dots \quad (53)$$

We can post-process the generated equations to obtain new equations for derivatives:

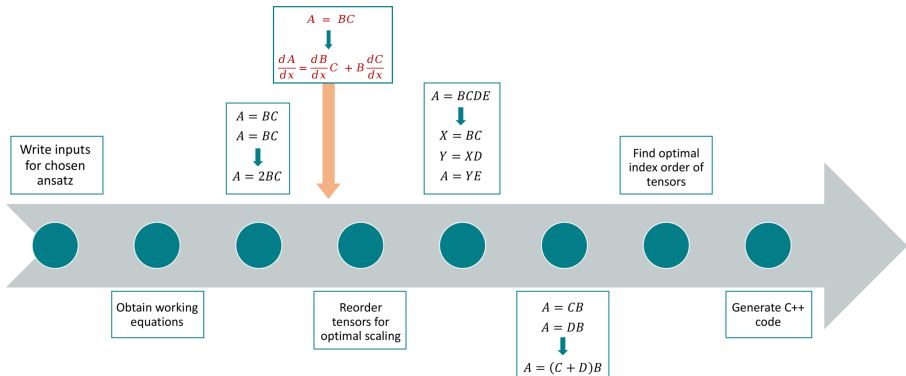
- Product rule
- Relabel the quantities with corresponding derivatives

Response Properties(II)

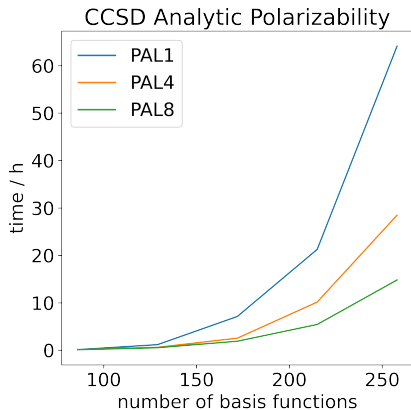
$$\frac{\partial^2 E}{\partial x \partial y} = \sum_{\mu\nu} D_{\mu\nu} \frac{\partial^2 h_{\mu\nu}}{\partial x \partial y} + \sum_{\mu\nu} \frac{\partial D_{\mu\nu}}{\partial x} \frac{\partial h_{\mu\nu}}{\partial y} + \dots \quad (53)$$

We can post-process the generated equations to obtain new equations for derivatives:

- Product rule
- Relabel the quantities with corresponding derivatives



Response Properties(III)



- Maxcore: 5000 MB per core
- CPU: AMD EPYC 75F3
- Basis: def2-TZVP
- Frozen core, single gradient step

- Scales the same as unperturbed methods, with a pre-factor

Summary and Outlook

Summary:

- In quantum chemistry (electronic structure theory), we strive for highly accurate electronic energy and molecular properties
- Accurate wavefunction method like higher-order Coupled Cluster methods are hard implement
- Automated code generation gives us the access towards these "advanced" methods

Summary and Outlook

Summary:

- In quantum chemistry (electronic structure theory), we strive for highly accurate electronic energy and molecular properties
- Accurate wavefunction method like higher-order Coupled Cluster methods are hard implement
- Automated code generation gives us the access towards these "advanced" methods

In the future, we wish to have:

- Better performance in parallelization
- Lower scaling methods (new approximations)
- ...

Acknowledgements

Anastasios Papadopoulos

Ingolf Harden

Kantharuban Sivalingam

Ute Becker

Marvin Lechner

AGE Team

Frank Neese



MAX-PLANCK-INSTITUT
FÜR KOHLENFORSCHUNG

