WP3
Exascale Coupled-Cluster Methods

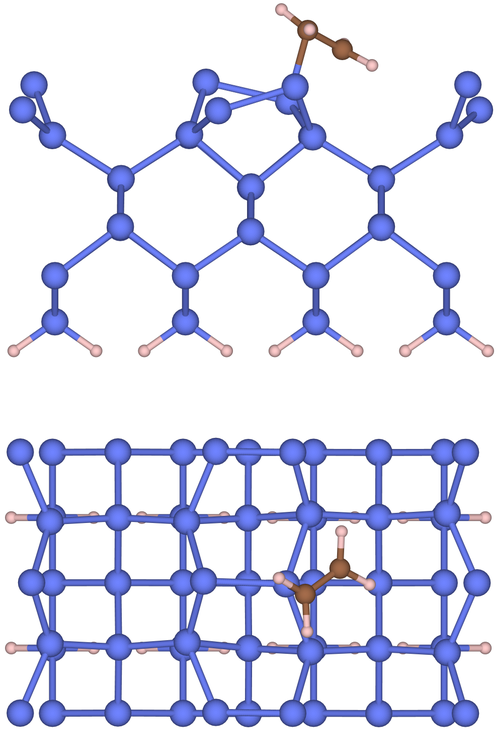
Purpose/Milestone/Deliverables

Andreas Grüneis (TU Wien), 18.6.2022, Riga Workshop

Why Exascale Coupled-Cluster Methods?

Applications and Results

CCSD(T) for C_2H_4@Si(cd)

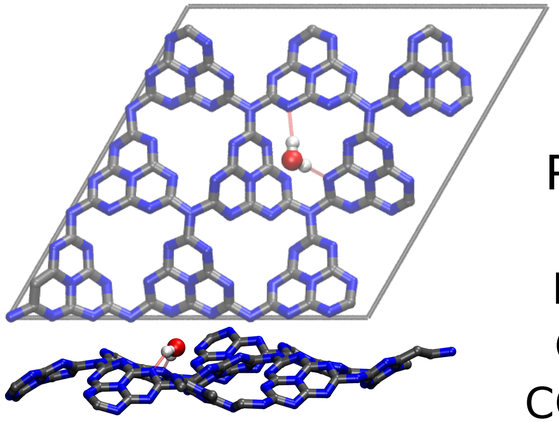


70 atoms with $N_o = 110$. CCSD and (T) calculations with $N_v = 1650$ and $N_v = 550$, respectively

Method	E_{ad}(kcal/mol)
PBE	-12.2
PBE-D3	-17.1
HF	1.6
MP2	-20.0
CCSD	-11.4
CCSD(T)	-9.9

CCSD(T):RPA for H_2O@[C_3N_4]

138 atoms with $N_o = 292$. Embedded CCSD(T) calculations with up to $N_o = 40$ and $N_v = 800$



Method	E_{ad} (kcal/mol)
PBE	-8.1
PBE-D3	-12.2
PBE-TS	-12.7
PBE0-MBD	-12.4
HF	-4.8
MP2:dMP2	-12.4
CCSD:RPA	-10.7
CCSD(T):RPA	-11.9

WP3 Exascale Coupled-Cluster Methods

Objectives

1. ***Interfacing*** the coupled-cluster-theory library ***CC4S-X to widely used electronic structure theory packages***, serving all code families. Currently only an interface to the PAW code *VASP* exists;
2. To ***benchmark and improve the efficiency of CC4S and CTF*** and the newly developed interfaces for CC, based on a detailed performance analysis of all three parts. This objective crucially requires ***co-design*** to reduce the time-to-solution as well as energy-to-solution;
3. Expand the scope of CC theory calculations, by ***adding support for block sparsity to CTF*** to reduce CPU time. We note that tensors occurring in CC calculations become block-sparse for systems with certain symmetries, such as translational symmetries in periodic crystals, as well as for local-orbital representations. The computational cost can be reduced by supporting block sparsity in the underlying tensor framework efficiently and can finally enable the usage of CC for system sizes that are not reachable with today's hard- and software.

WP3 Exascale Coupled-Cluster Methods

Task 1 / Deliverable 1

Task 1: Interfacing *ab initio* codes to *CC4S* **(TUW, MPG-FHI, DTU, HUB, UCL, MPG-MPCDF)** Coupled-cluster calculations using *CC4S* require (a) Hamiltonian matrix elements and (b) intermediate quantities from the underlying *ab initio* code. These interfaces will be developed for each code family. The most important matrix elements are those of *non-local operators*, which will be developed in collaboration with WP1 and WP2. In particular, (a) corresponds to Fock matrix elements in the basis of density functional or Hartree-Fock orbitals, whereas (b) corresponds to 3-index intermediate quantities needed for the computation of *non-local* Coulomb interaction matrix elements.

WP3	D3.1	D5	Software and documentation of interfaces of CC4S to ab initio codes	TUW	Other	Confidential, on 31 Mar 2023	Pending

JOSS
The Journal of Open Source Software

Interface to high-performance periodic coupled-cluster theory calculations with atom-centered, localized basis functions

Evgeny Moerman[1], Felix Hummel[2], Andreas Grüneis[2], Andreas Irmler[2], and Matthias Scheffler[1]

At least one more interface needed:

exciting

FLAPW, NAO, PAW ..?

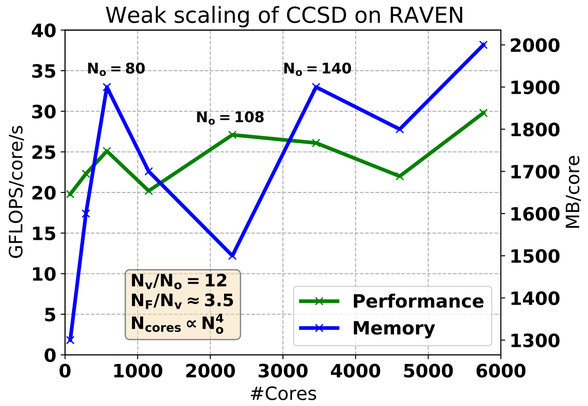
WP3 Exascale Coupled-Cluster Methods

Task 2 / Deliverable 2

Task 2: Profiling *CC4S*, *CTF* and interfaces (TUW, MPG-MPCDF, MPG-FHI, DTU, HUB, UCL) We will identify archetypal test sets for *CC4S* calculations and perform a detailed benchmark and profiling study of *CC4S*, *CTF* and the interfaces developed in Task 1. Once bottlenecks are identified, computationally efficient modifications will be developed and implemented, considering co-design aspects such as minimising time-to-solution and energy-to-solution. Contributions to the *CTF, resulting in* ***CTF-X*** will be carried out in collaboration with its main developer Edgar Solomonik, who is a member of the second shell. Benchmark and profile studies will address various architectures including investigations for different topologies and exascale platforms.

WP3	D3.2	D6	Benchmark report	TUW	Report	Confidential, onl	31 Mar 2023	Pending

Weak scaling



Paper in preparation A. Irmler

Hardware issues of (pre-)exascale computers:
Too much compute power/node compared to network bandwidth?
Wait for next generation GPU accelerated nodes with faster internode connection?

Efficient use of Accelerators in Tensor Frameworks:
Switch to CuTensor in future once they have good multi-node efficiency
Computer scientists/hardware vendors need to develop GPU-optimized libraries

Co-design and energy-to-solution are still open issues.
Advanced internode connection for accelerated nodes,
save energy by maximzing GFLOPS/core

WP3 Exascale Coupled-Cluster Methods

Task 3 / Deliverable 3

Task 3: Support of block-sparsity (TUW, MPG-MPCDF) We will add support for ***block-sparse tensor data algebra*** to *CTF-X*. We note that sparse tensor algebra is already supported in *CTF*; the generalisation to block-sparsity can greatly benefit from the existing code infrastructure. This task will be carried out in collaboration with the main developer of *CTF* Edgar Solomonik.

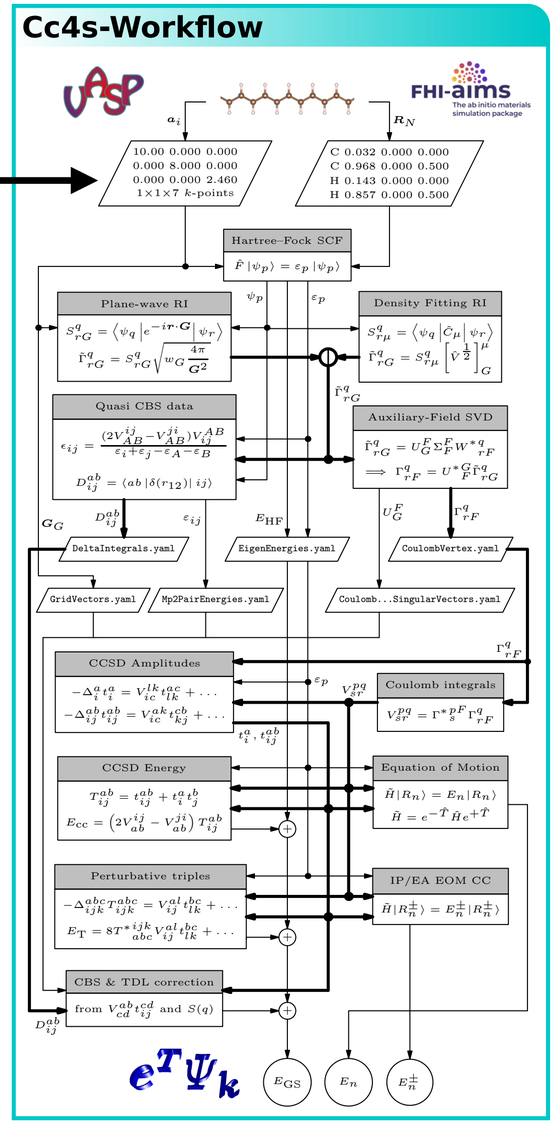
WP3	D3.3	D7	Report and documentation of CC4S-X and CTF-X including block-sparse tensors	TUW	Report	Public	31 Mar 2024	Pending

Prototype of block-sparse tensors in CTF and CC4S exist but more work/time needed

WP3 Exascale Coupled-Cluster Methods

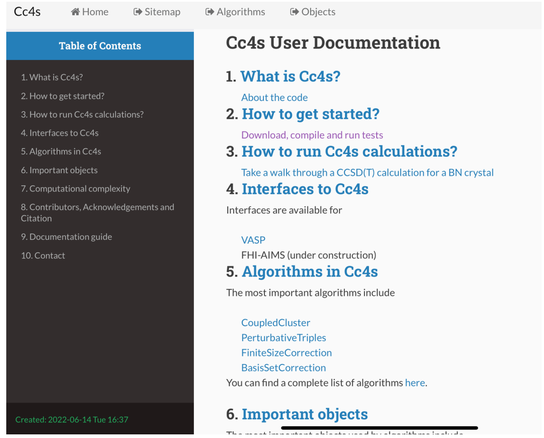
Next steps for Task 1 / Deliverable 1

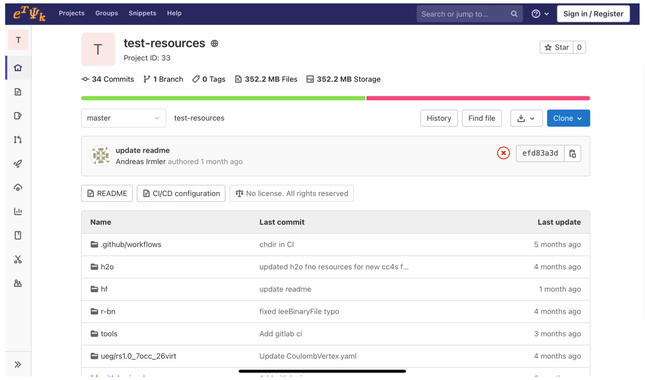
Task 1: Interfacing *ab initio* codes to *CC4S*

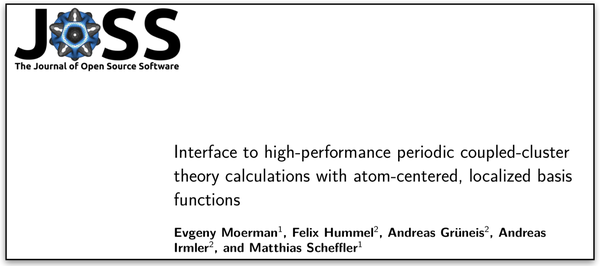


<u>Available documentation and resources:</u>

Online user manual, Git-Repo with test resources, Interface to FHI-aims





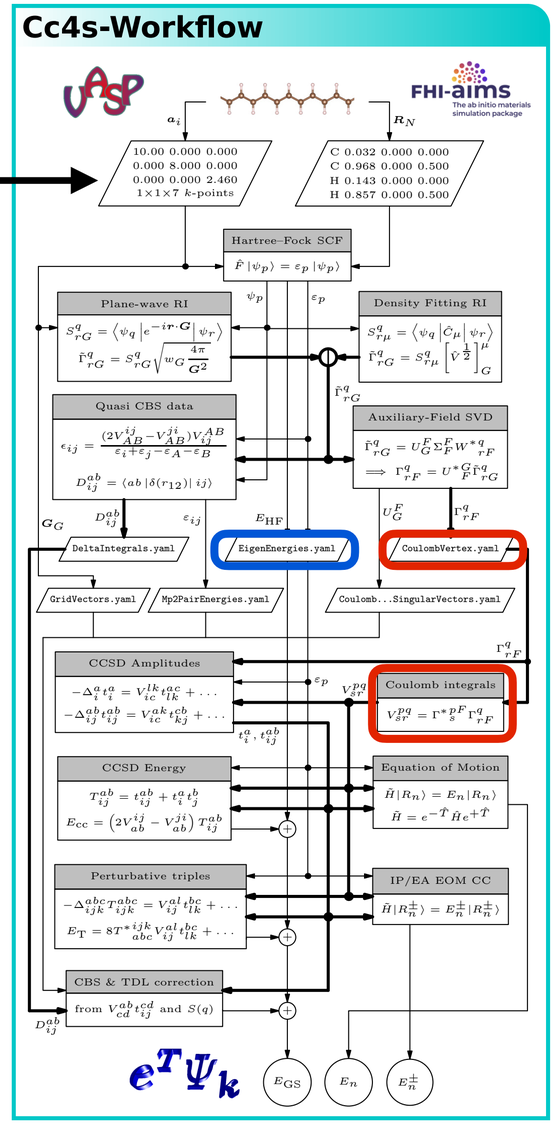


WP3 Exascale Coupled-Cluster Methods

Next steps for Task 1 / Deliverable 1

Task 1: Interfacing *ab initio* codes to *CC4S*

exciting



Required input from interfaced ab-initio codes, e.g. the HF molecule:

https://gitlab.cc4s.org/cc4s/test-resources/-/tree/master/hf

Name
..
CoulombVertex.elements
{..} CoulombVertex.yaml
EigenEnergies.elements
{..} EigenEnergies.yaml

CoulombVertex.yaml 239 Bytes

```yaml
version: 100
type: Tensor
scalarType: Complex64
dimensions:
  - length:     88
    type: AuxiliaryField
  - length:     12
    type: State
  - length:     12
    type: State
elements:
  type: IeeeBinaryFile
unit: 1
metaData:
  halfGrid: 1
```

EigenEnergies.yaml 475 Bytes

```yaml
version: 100
type: Tensor
scalarType: Real64
dimensions:
  - length:   12
    type: State
elements:
  type: TextFile
unit: 1
metaData:
  fermiEnergy:  0.00000000000000
  energies:
    -  -26.28781513595019
    -  -1.60185818997174
    -  -0.75310079870926
    -  -0.64301123475492
    -  -0.64301123475492
    -  0.21310327672019
    -  1.05741764792928
    -  1.05741764792928
    -  1.12787893896500
    -  1.16324598061683
    -  1.87925217029884
    -  55.03320444362973
```