Orbital Optimization

1. Motivation
2. Implementation
3. Tutorial on C_2

Joshua P. Townsend & Amanda E. Dumi
Sandia National Laboratories
Albuquerque, NM 87185

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Motivation

- **Slater-Jastrow (SJ) is a simple & common ansatz:**

$$\Psi^{SJ}(X) = e^{J(X)} \det \begin{vmatrix} \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \cdots & \phi_1(\mathbf{x}_N) \\ \phi_2(\mathbf{x}_1) & \phi_2(\mathbf{x}_2) & \cdots & \phi_2(\mathbf{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_N(\mathbf{x}_1) & \phi_N(\mathbf{x}_2) & \cdots & \phi_N(\mathbf{x}_N) \end{vmatrix}$$

- **Quality (variance) is set by nodal surface, i.e. by the ϕ_k's**
- **How to get 'good' orbitals? (especially where DFT struggles!)**
- ***Orbital optimization improves wave function quality with essentially no performance cost!***

Implementation: Key Ideas

- **Express each SPO as a linear combination in some space:**

$$\tilde{\phi}_P = \sum_Q \phi_Q U_{QP}$$

'Rotated' SPO **'bare' SPO** **Unitary matrix**

- **In QMCPACK, U is built from an anti-hermitian matrix, κ :**

$$U = e^{\kappa} \qquad \kappa = -\kappa^{\dagger}$$

Implementation: Key Ideas

- **Elements of kappa matrix are related to single excitations[1,2,3]:**

$$\hat{\kappa} = \sum_{ij} \kappa_{ij} \hat{E}_{ij}$$

Variational parameter

$$\hat{E}^{\sigma}_{ij} = \hat{a}^{\dagger}_{i\sigma} \hat{a}_{j\sigma}$$

Excitation from $j \rightarrow i$ for spin channel σ

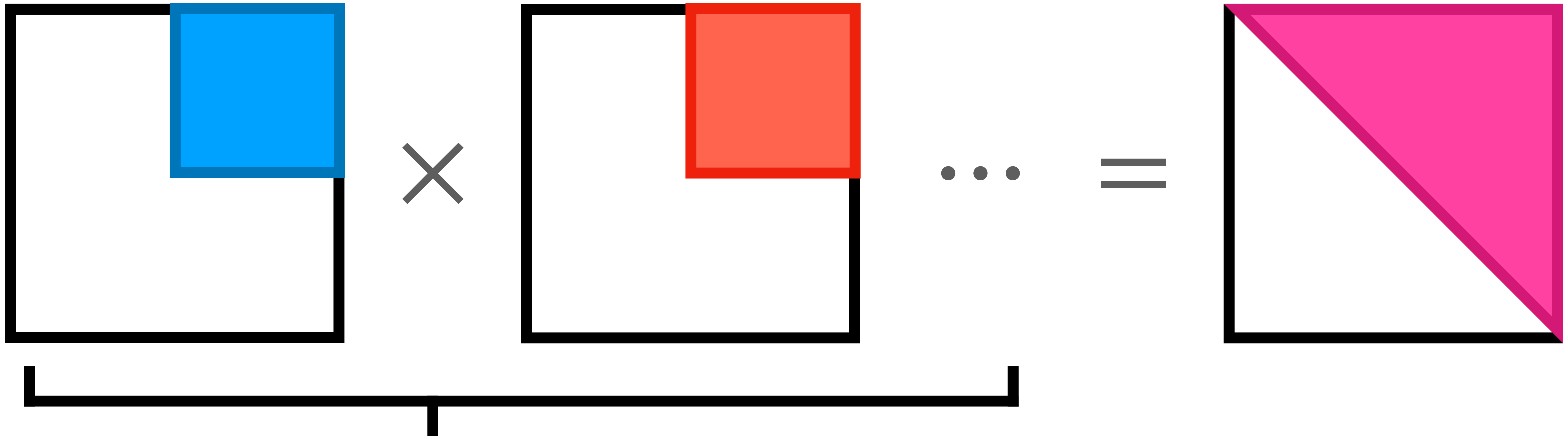
- **Action of $\hat{E}^{\sigma}_{ij}$ on a determinant is to replace $|\phi_j\rangle$ with $|\phi_i\rangle$**
- **$\hat{\kappa}$ has structure:**

$$\kappa = \begin{array}{cc} \text{Nocc} & \text{Nvirt} \\ \hline 0 & \\ \text{-}c.c. & 0 \end{array}$$

$N_{occ} \times N_{virt}$ parameters

Implementation: Key Ideas

- **After optimization the net rotation from bare SPOs to optimal SPOs is dense:**

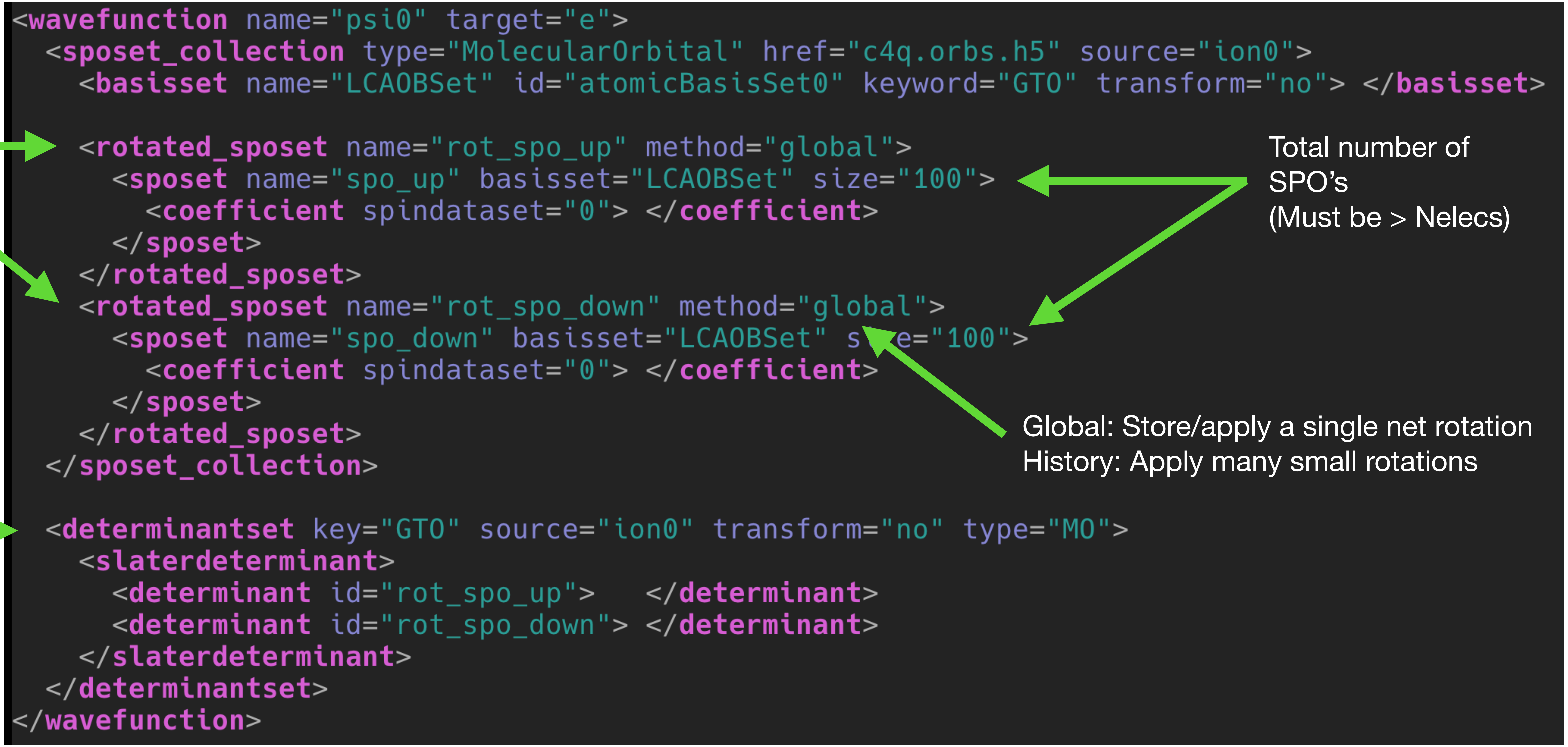


List of kappa matrices for each optimization step
‘**history**’ scheme

Single dense kappa matrix updated at each optimization step
‘**global**’ scheme

- **QMCPACK provides i/o for both schemes. Equivalent results but storage and start-up time are different. Global is default.**

Input Files: How to turn on orbopt



```
<wavefunction name="psi0" target="e">
  <sposet_collection type="MolecularOrbital" href="c4q.orbs.h5" source="ion0">
    <basisset name="LCAOBSet" id="atomicBasisSet0" keyword="GTO" transform="no"> </basisset>

    <rotated_sposet name="rot_spo_up" method="global">
      <sposet name="spo_up" basisset="LCAOBSet" size="100">
        <coefficient spindataset="0"> </coefficient>
      </sposet>
    </rotated_sposet>
    <rotated_sposet name="rot_spo_down" method="global">
      <sposet name="spo_down" basisset="LCAOBSet" size="100">
        <coefficient spindataset="0"> </coefficient>
      </sposet>
    </rotated_sposet>
  </sposet_collection>

  <determinantset key="GTO" source="ion0" transform="no" type="MO">
    <slaterdeterminant>
      <determinant id="rot_spo_up">   </determinant>
      <determinant id="rot_spo_down"> </determinant>
    </slaterdeterminant>
  </determinantset>
</wavefunction>
```

RotatedSPO wraps a bare SPOset

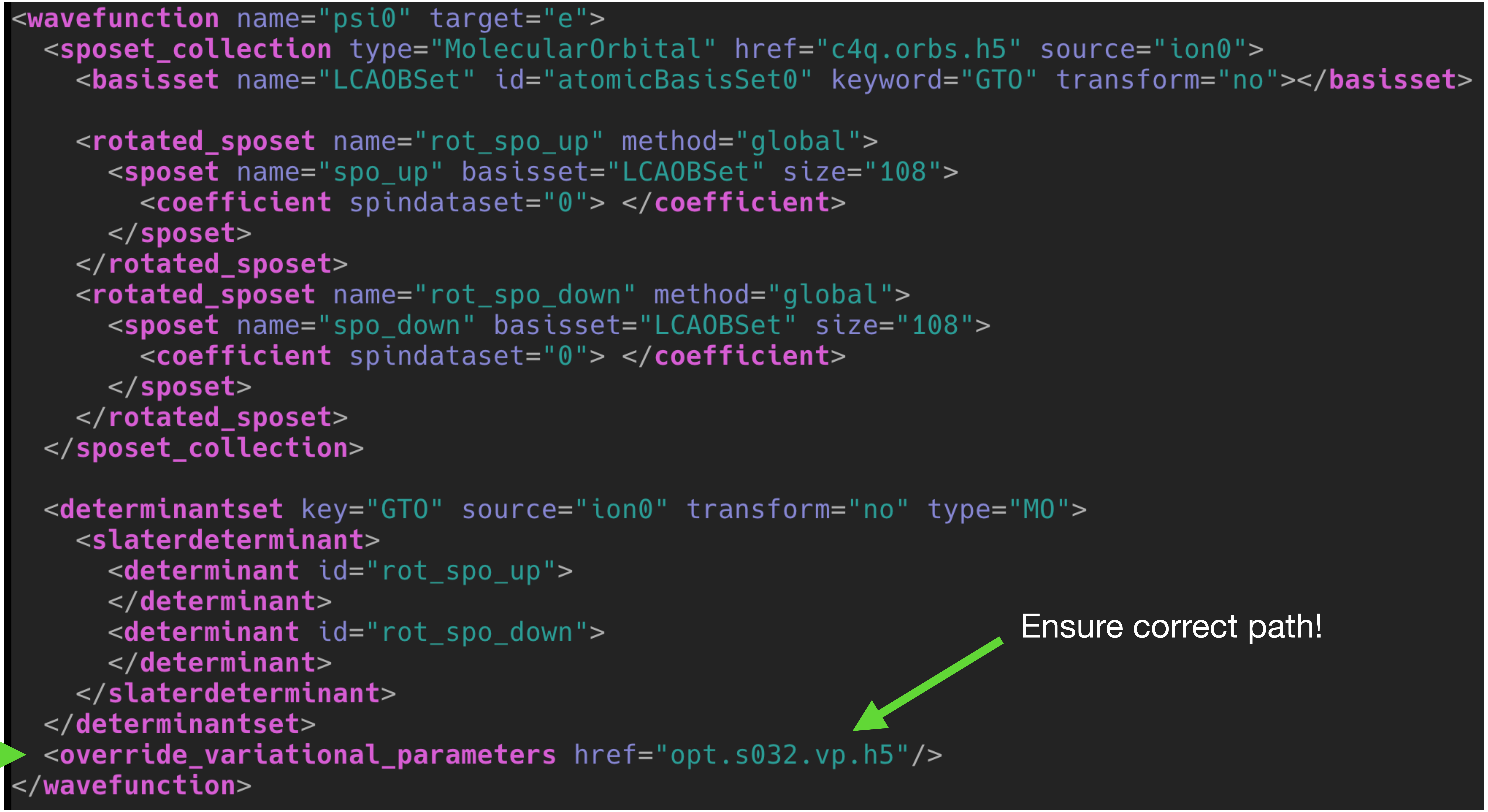
Total number of SPO's (Must be > Nelecs)

Global: Store/apply a single net rotation
History: Apply many small rotations

Determinant built from RotatedSPOs

Input Files: How to run optimized wfn

NB: Jastrow is ignored in this screenshot, but you should probably use one!

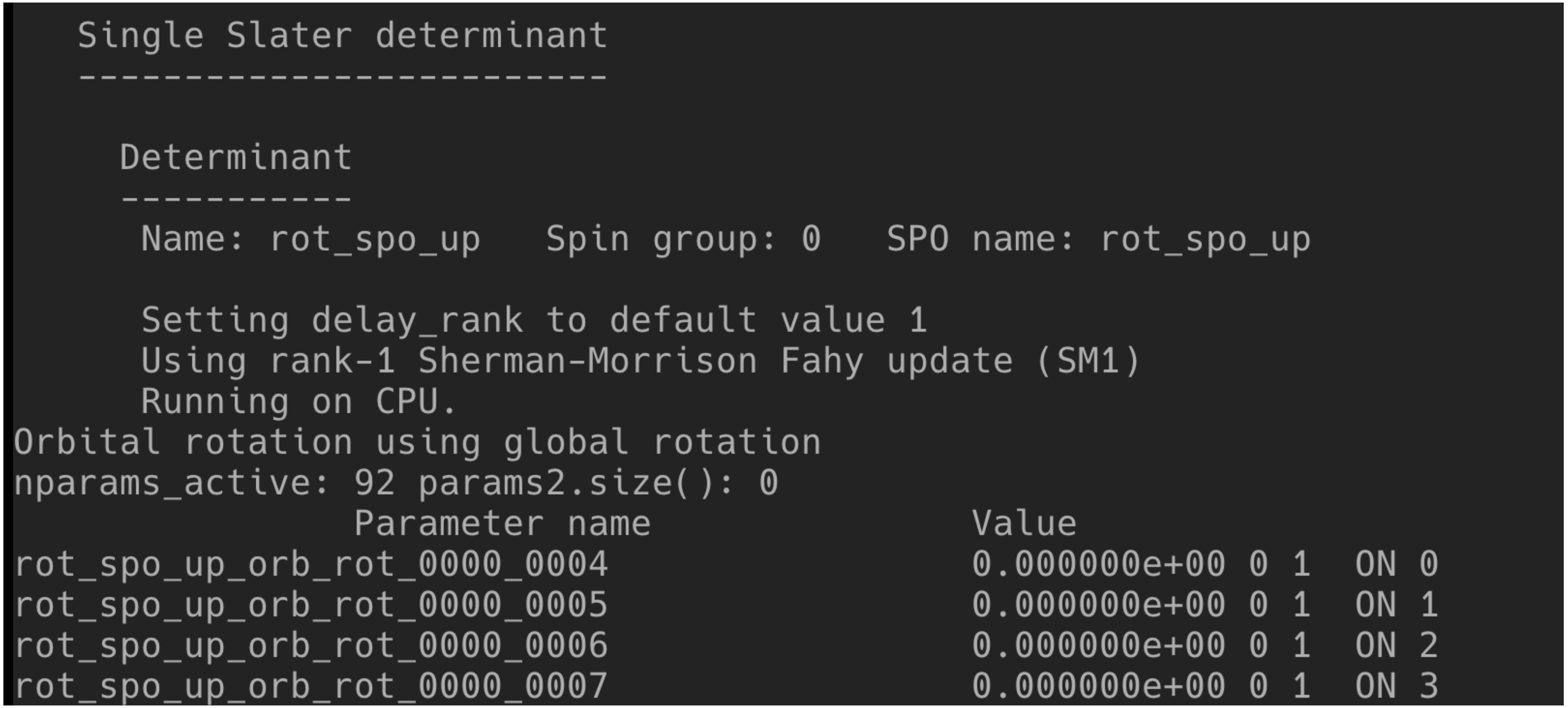


```
<wavefunction name="psi0" target="e">
  <sposet_collection type="MolecularOrbital" href="c4q.orbs.h5" source="ion0">
    <basisset name="LCAOBSet" id="atomicBasisSet0" keyword="GTO" transform="no"></basisset>

    <rotated_sposet name="rot_spo_up" method="global">
      <sposet name="spo_up" basisset="LCAOBSet" size="108">
        <coefficient spindataset="0"> </coefficient>
      </sposet>
    </rotated_sposet>
    <rotated_sposet name="rot_spo_down" method="global">
      <sposet name="spo_down" basisset="LCAOBSet" size="108">
        <coefficient spindataset="0"> </coefficient>
      </sposet>
    </rotated_sposet>
  </sposet_collection>

  <determinantset key="GTO" source="ion0" transform="no" type="MO">
    <slaterdeterminant>
      <determinant id="rot_spo_up">
      </determinant>
      <determinant id="rot_spo_down">
      </determinant>
    </slaterdeterminant>
  </determinantset>
  <override_variational_parameters href="opt.s032.vp.h5"/>
</wavefunction>
```

Ensure correct path!

Read in rotation params stored in vp.h5 file

Output Files: How to tell it’s working (Optimization)



```
   Single Slater determinant
   -------------------------


     Determinant
     -----------
      Name: rot_spo_up   Spin group: 0   SPO name: rot_spo_up

      Setting delay_rank to default value 1
      Using rank-1 Sherman-Morrison Fahy update (SM1)
      Running on CPU.
Orbital rotation using global rotation
nparams_active: 92 params2.size(): 0
            Parameter name                 Value
rot_spo_up_orb_rot_0000_0004              0.000000e+00 0 1  ON 0
rot_spo_up_orb_rot_0000_0005              0.000000e+00 0 1  ON 1
rot_spo_up_orb_rot_0000_0006              0.000000e+00 0 1  ON 2
rot_spo_up_orb_rot_0000_0007              0.000000e+00 0 1  ON 3
```

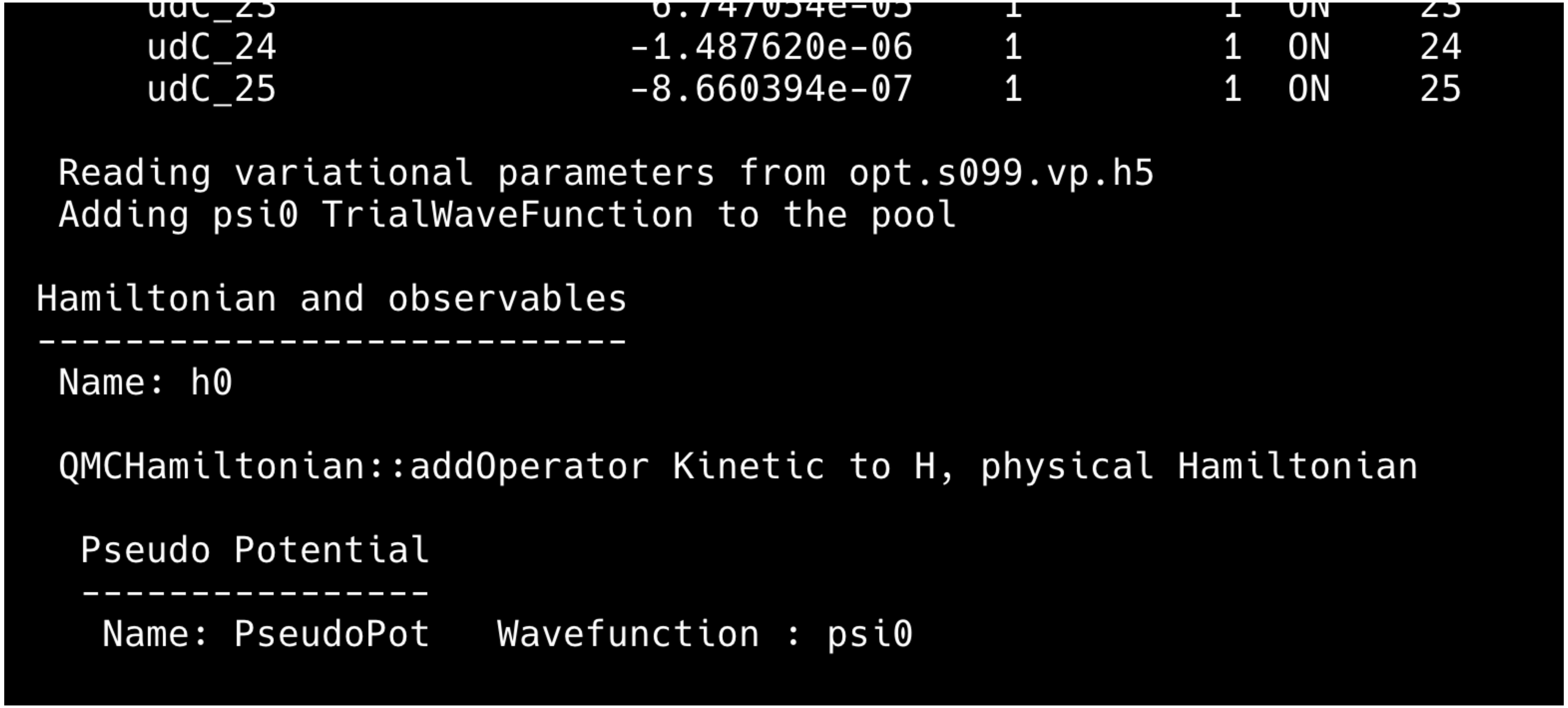
Check name

Check method

Check params

Output Files: How to tell it’s working (Production)

Loads in rotation parameters



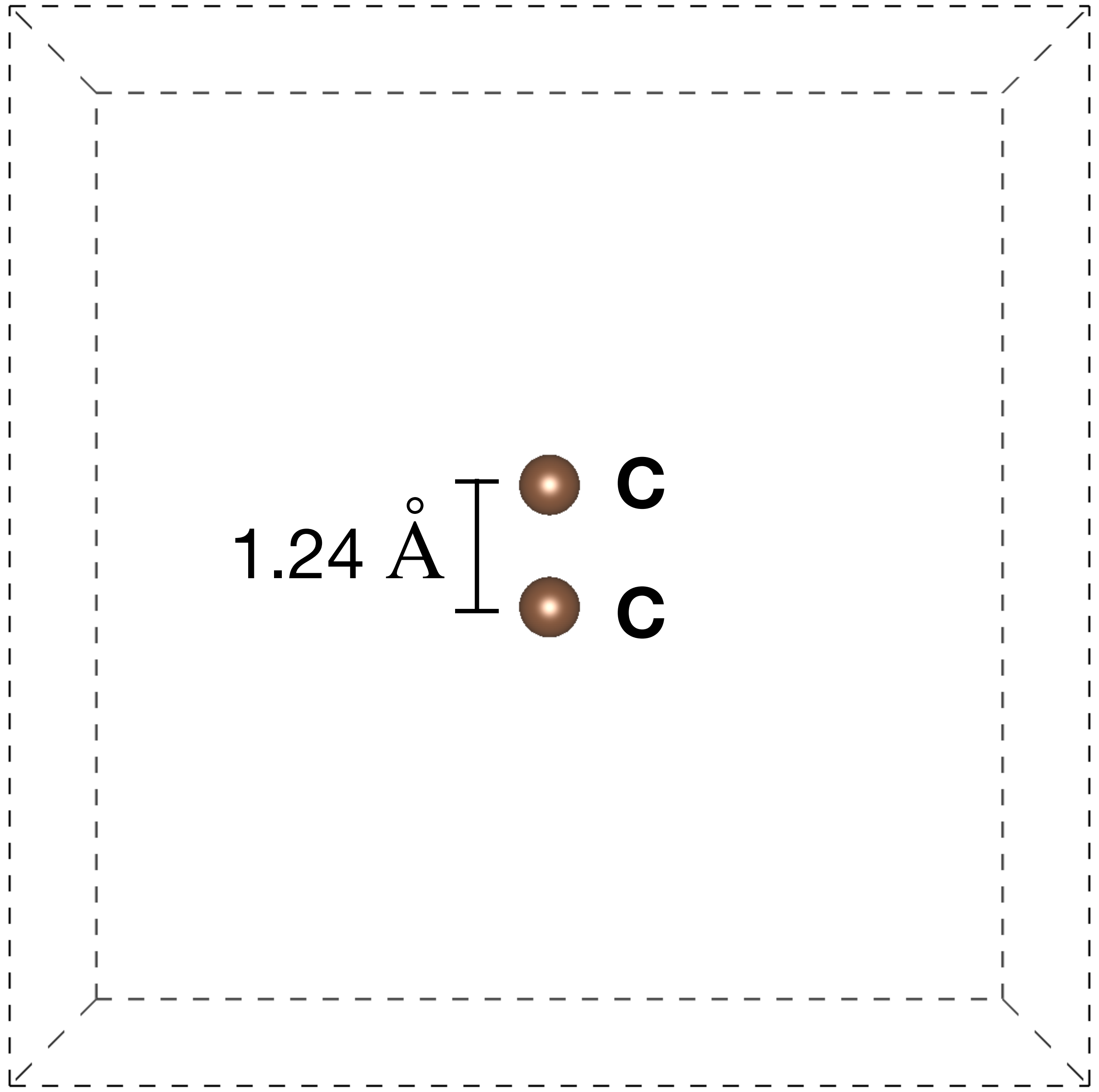
```
     udC_23                 6.747054e-05    1         1  ON    23
     udC_24                -1.487620e-06    1         1  ON    24
     udC_25                -8.660394e-07    1         1  ON    25

 Reading variational parameters from opt.s099.vp.h5
 Adding psi0 TrialWaveFunction to the pool

Hamiltonian and observables
---------------------------
 Name:  h0

 QMCHamiltonian::addOperator Kinetic to H, physical Hamiltonian

  Pseudo Potential
  ----------------
   Name: PseudoPot   Wavefunction : psi0
```

Tutorial: Carbon dimer

- Clone: https://github.com/QMCPACK/qmcpack_workshop_2023
- Go to: day2_orbital_optimization
- Read README

- Tutorial problems:

1. Converge energy w.r.t. basis
2. Binding curve (likely on your own time!)



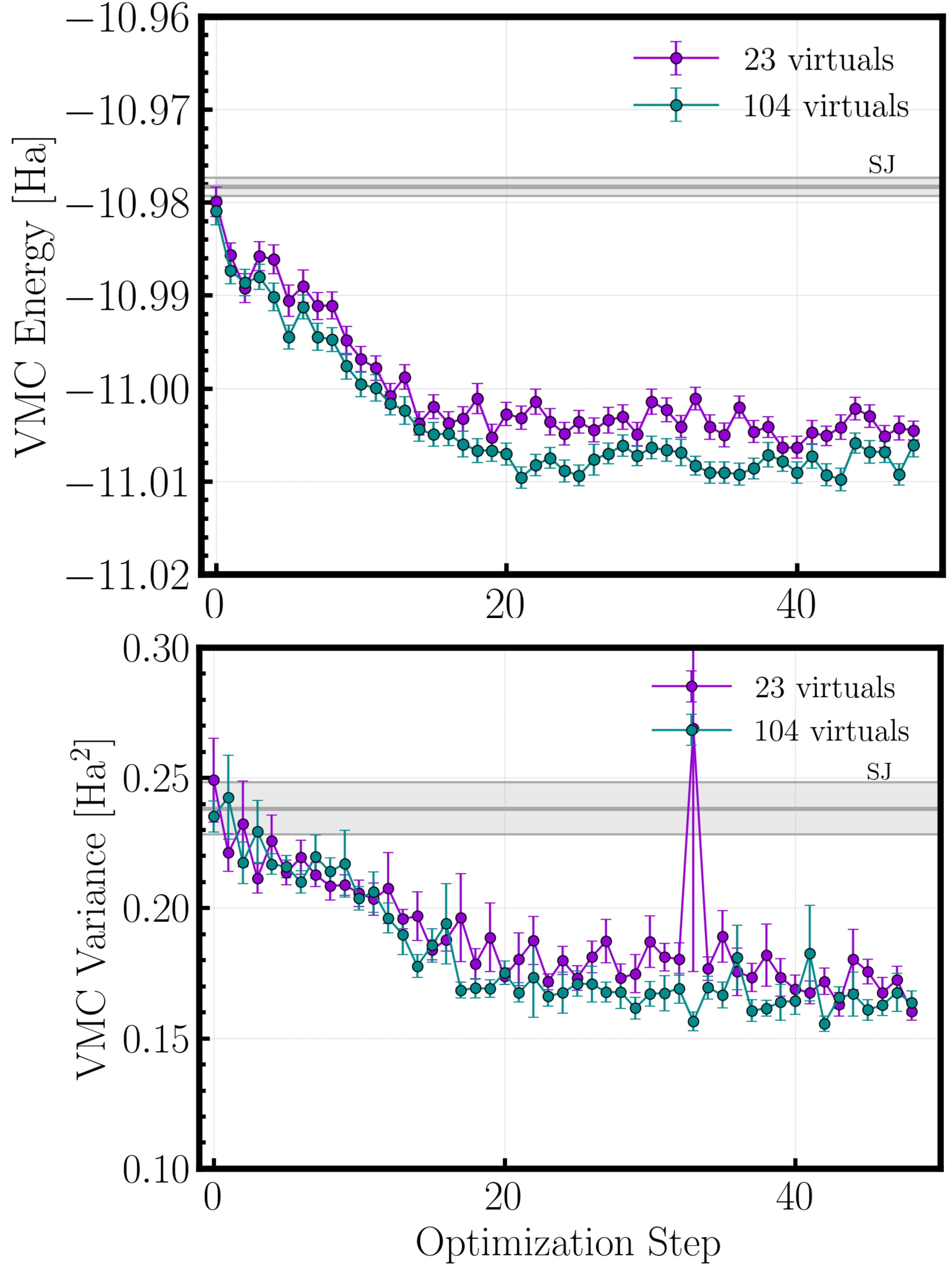
Carbon dimer geometry for problems 1 & 2.

Tutorial: 0_convergence

`day2_orbital_optimization/0_convergence`

1. Go to ‘0_orbital_generation’, run `psycf`
2. Go to ‘1_qmc_opt’, run `qmcpack` in each norbs_xyz directory
3. Compare results (see example on right)
4. Go to ‘2_qmc_production’, copy in optimized parameters, run `qmcpack`

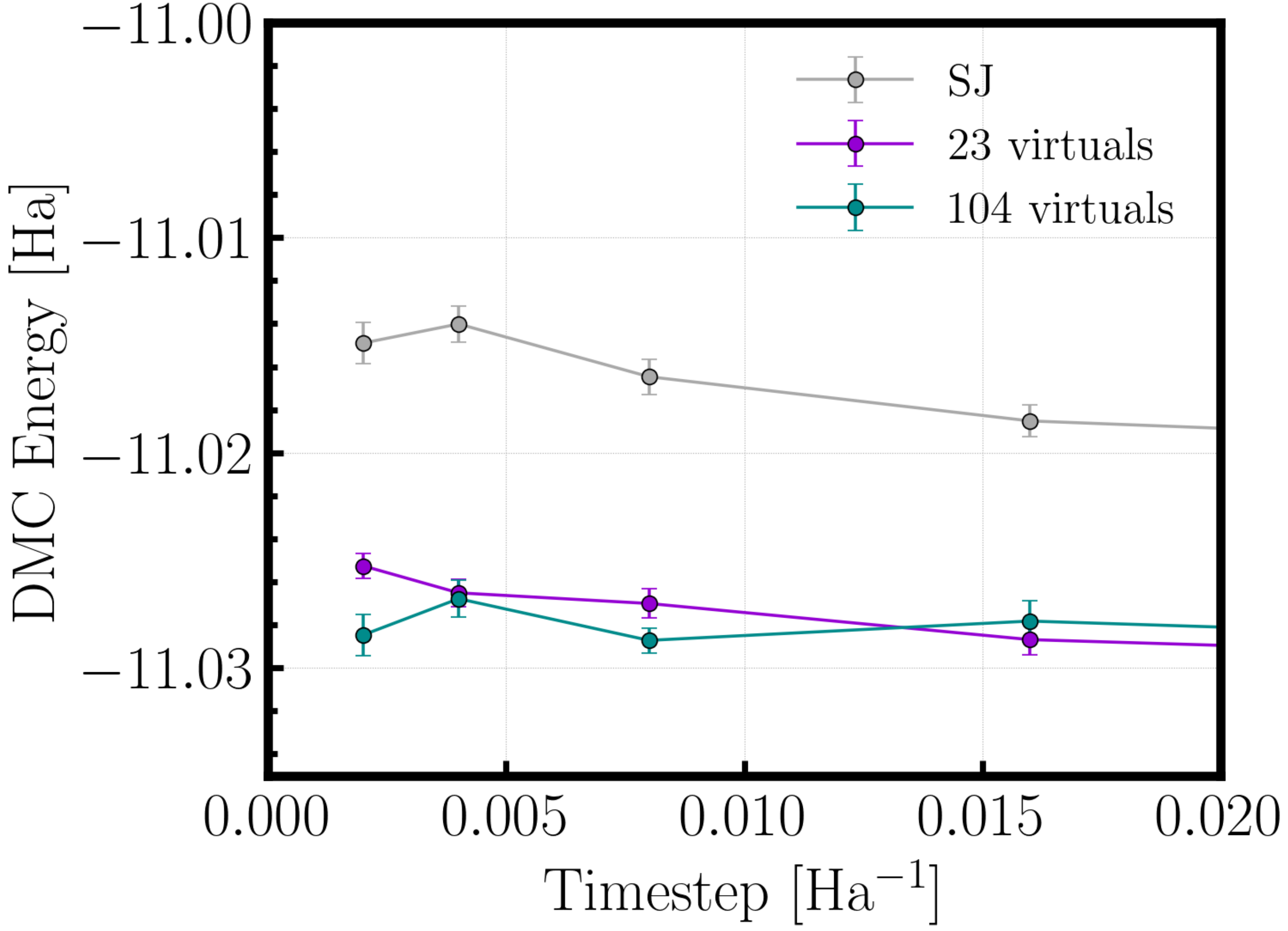
You should find that more virtuals yields a lower energy - but only up to a point.



Tutorial: 0_convergence

```
carbon-dimer/ecp/0_convergence
```

1. Go to ‘0_orbital_generation’, run `psycf`
2. Go to ‘1_qmc_opt’, run `qmcpack` in each norbs_xyz directory
3. Compare results (see example on right)
4. Go to ‘2_qmc_production’, copy in optimized parameters, run `qmcpack`



You should find that more virtuals yields a lower energy - but only up to a point.

Tutorial: 0_convergence

`carbon-dimer/ecp/0_convergence`

Suggested homework:

1. Adjust `minWalkers`, `shift_i` to see how optimization rate changes
2. Adjust `walkers_per_rank`, `steps`, `blocks` to speed up optimization w/ constant error bar
3. Try out binding curve calculation in `day2_orbital_optimization/1_binding_curve`

References

1. Toulouse & Umrigar, JCP 126, (2007)
2. Toulouse & Umrigar, JCP 128, (2008)
3. Helgaker, “Molecular Electronic Structure Theory”, (2013)