

## I. SECOND-ORDER ORBITAL OPTIMIZATION

To begin our response discussion at the SCF level we will first discuss solving the SCF equations with full Newton steps rather than diagonalization. Where diagonalization builds completely new orbitals at every iteration; this Newton scheme will construct new orbitals by rotating the orbitals of the previous iteration. These rotations will be formed by fundamental linear algebra principles as their gradient and Hessian is well defined. This is often called second-order optimization and, if done exactly, the convergence will be quadratic with respect to the MO gradient<sup>1</sup>, e.g.,  $\text{grad}_{pq}^{n+1} \approx (\text{grad}_{pq}^n)^2$ . As a consequence of the quadratic convergence, the second-order procedure can only be started once all elements of the gradient in the MO basis are less than one, else the error will increase.

The overall goal is to find an orbital rotation matrix,  $\boldsymbol{\kappa}$ , that satisfies the following set of linear equations:

$$\mathbf{E}^{(2)}\boldsymbol{\kappa} = -\mathbf{E}^{(1)} \quad (1)$$

where  $\mathbf{E}^{(2)}$  is our orbital Hessian and  $\mathbf{E}^{(1)}$  is the orbital gradient. The  $\boldsymbol{\kappa}$  matrix is a antisymmetric matrix of non-redundant rotations that describes the unitary transformation  $\mathbf{U}$

$$\mathbf{U} = e^{\boldsymbol{\kappa}} \quad (2)$$

$$\mathbf{C}_{n+1} = \mathbf{C}_n \mathbf{U} \quad (3)$$

Rotations within a given space (inactive or virtual for SCF) are redundant: the only non-redundant orbital rotations for SCF are the inactive-virtual ones.

The orbital gradient for RHF gradient can be written as,

$$E_{pq}^{(1)} = F_{pq} \quad (4)$$

where  $F$  is our generalized Fock matrix<sup>1</sup> in MO indices,

$$F_{pq} = h_{pq} + 2g_{pqrs}D_{rs} - g_{prqs}D_{rs} \quad (5)$$

To derive our orbital-response Hessian we can introduce the one-index and two-index transformed integrals

$$h_{pq}^\kappa = (\kappa_{po}h_{oq} + \kappa_{qo}h_{po}) \quad (6)$$

$$g_{pqrs}^\kappa = (\kappa_{po}g_{oqrs} + \kappa_{qo}g_{pors} + \kappa_{ro}g_{pqos} + \kappa_{so}g_{pqro}) \quad (7)$$

insert these rotated integrals into our Fock matrix,

$$F^\kappa = h_{pq}^\kappa + 2g_{pqrs}^\kappa D_{rs} - g_{prqs}^\kappa D_{rs} \quad (8)$$

expanding these equations we find,

$$\begin{aligned} F_{pq}^\kappa \equiv \mathbf{E}_{pqrs}^{(2)}\kappa_{rs} = & (\kappa_{mp}h_{pn} + \kappa_{np}h_{mp}) + \\ & 2(D_{rs}\kappa_{po}g_{oqrs} + \kappa_{qo}g_{pors} + \kappa_{ro}g_{pqos} + \kappa_{so}g_{pqro}) - \\ & (D_{rs}\kappa_{po}g_{orqs} + \kappa_{qo}g_{pros} + \kappa_{ro}g_{poqs} + \kappa_{so}g_{prqo}) \end{aligned} \quad (9)$$

and then simplified by considering the permutational symmetry of the  $g$  tensor and collapse of several elements to form various Fock matrices,

$$\mathbf{E}_{pqrs}^{(2)}\kappa_{rs} = (F_{qs}\kappa_{rs} + F_{pr}\kappa_{rs}) + \kappa_{rs}(4g_{pqrs} - g_{prqs} - g_{sqrp}) \quad (10)$$

For RHF only the occupied - virtual rotations change the Wavefunction. We can therefore simplify this above once more,

$$\mathbf{E}_{iajb}^{(2)}\kappa_{jb} = (-F_{ab}\kappa_{jb} + F_{ij}\kappa_{jb}) + \kappa_{jb}(4g_{iajb} - g_{ijab} - g_{bjai}) \quad (11)$$

Our Eqn. 1 therefore becomes,

$$\mathbf{E}_{iajb}^{(2)}\kappa_{jb} = F_{ab} \quad (12)$$

The easiest way to solve this is by direct inversion,

$$\kappa_{jb} = F_{ab}[\mathbf{E}_{iajb}^{(2)}]^{-1} \quad (13)$$

Solving Eqn. 12 either through direct inversion or through an iterative method is known as second-order SCF (SOSCF). Returning to our "physicist's water molecule", the convergence pattern for SOSCF is demonstrated in Fig. 1. We can observe that three regular DIIS steps were

required before all elements of the gradient were less than one. The RHF wavefunction is then converged to within machine precision in terms of energy and density within 4 SOSCF iterations. SOSCF is quite beneficial in terms of the number of iterations, but is not currently competitive in terms of cost. However, when solved exactly, the cost of inverting the full orbital Hessian is  $o^3v^3$ , unless  $o$  is very small, we have effectively changed the cost of SCF from  $N^4$  to  $N^6$ , or roughly the cost of CCSD.

FIG. 1. A RHF computation of "physicist's water molecule" starting with a core Hamiltonian guess in the cc-pVDZ basis set utilizing DIIS and SOSCF convergence. For the SOSCF step, the Eq. 1 has been solved exactly by inverting the full orbital Hessian. The right most abbreviations indicate the type of step taken in each iteration.

RHF Iteration	1: Energy = -68.98003273414295	dE = -6.898E+01	dRMS = 1.165E-01	
RHF Iteration	2: Energy = -69.64725442845806	dE = -6.672E-01	dRMS = 1.074E-01	DIIS
RHF Iteration	3: Energy = -75.79192914624532	dE = -6.144E+00	dRMS = 2.892E-02	DIIS
RHF Iteration	4: Energy = -75.97218922804181	dE = -1.802E-01	dRMS = 7.564E-03	DIIS
RHF Iteration	5: Energy = -75.98973929215161	dE = -1.755E-02	dRMS = 3.049E-04	SOSCF
RHF Iteration	6: Energy = -75.98979578473095	dE = -5.649E-05	dRMS = 1.231E-06	SOSCF
RHF Iteration	7: Energy = -75.98979578551825	dE = -7.873E-10	dRMS = 1.901E-11	SOSCF
RHF Iteration	8: Energy = -75.98979578551835	dE = -9.948E-14	dRMS = 2.395E-15	SOSCF

To reduce the cost we can solve Eqn. 12 through iterative means and to avoid the costly 4-index ERI transformation we can solve it in the AO basis. Consider the left hand side of Eqn. 12 again,

$$F_{ia}^\kappa \equiv \mathbf{E}_{iajb}^{(2)} \kappa_{jb} = (-F_{ab} \kappa_{jb} + F_{ij} \kappa_{jb}) + \kappa_{jb} (4g_{iajb} - g_{ijab} - g_{bjai}) \quad (14)$$

If we are careful we can compute this product using the conventional AO-based  $J$  and  $K$  routines:

$$\vartheta_{\lambda\sigma} = C_{j\lambda} \kappa_{jb} C_{b\sigma} \quad (15)$$

$$\begin{aligned} F_{ia}^\kappa = & (F_{ij} \kappa_{ja} - \kappa_{jb} F_{ba}) \\ & + C_{i\mu} (4J[\vartheta_{\lambda\sigma}]_{\mu\nu} - K[\vartheta_{\lambda\sigma}]_{\mu\nu} - K[\vartheta_{\lambda\sigma}]_{\nu\mu}) C_{a\nu} \end{aligned} \quad (16)$$

The cost of solving Eq. 12 is now an iterative  $N^4$ th when the left hand side is computed from Eq. 16 each of these contractions (often called a microiteration) is equivalent to the cost of a normal RHF iteration. The term macroiteration refers to a step in which the overall SCF energy is computed, this step collects all Fock builds, microiterations, and orbital rotations.

Our "physicist's water molecule" was again optimized using iterative SOSCF in Fig. 2. We observe that while the convergence is significantly faster than for the DIIS acceleration alone, the

FIG. 2. A RHF computation of "physicist's water molecule" starting with a core Hamiltonian guess in the cc-pVDZ basis set utilizing DIIS and SOSCF convergence. For the SOSCF step, Eq. 12 has been solved iteratively through a conjugate gradient method limited to four microiterations. The rightmost abbreviations indicate the type of step taken on each iteration.

RHF Iteration	1: Energy = -68.98003273414295	dE = -6.898E+01	dRMS = 1.165E-01	
RHF Iteration	2: Energy = -69.64725442845806	dE = -6.672E-01	dRMS = 1.074E-01	DIIS
RHF Iteration	3: Energy = -75.79192914624532	dE = -6.144E+00	dRMS = 2.892E-02	DIIS
RHF Iteration	4: Energy = -75.97218922804181	dE = -1.802E-01	dRMS = 7.564E-03	DIIS
RHF Iteration	5: Energy = -75.98970327666461	dE = -1.751E-02	dRMS = 5.908E-04	SOSCF
RHF Iteration	6: Energy = -75.98979576713703	dE = -9.249E-05	dRMS = 8.777E-06	SOSCF
RHF Iteration	7: Energy = -75.98979578513478	dE = -1.799E-08	dRMS = 8.237E-07	SOSCF
RHF Iteration	8: Energy = -75.98979578551806	dE = -3.832E-10	dRMS = 1.648E-08	SOSCF

overall convergence is much slower than when Eq. 12 was solved exactly. This is due to the fact that the number of microiterations used is insufficient to solve Eq. 12 exactly. More microiterations can be utilized, but a balance between macro- and microiterations must be computationally efficient. However, it should be noted that even though each microiteration of SOSCF is equivalent to the cost of a normal RHF step, the overall computational effort expended for SOSCF is generally greater than the DIIS methods for a given convergence criterion. SOSCF should therefore only be utilized for difficult to converge SCF cases.

## II. DIPOLE POLARIZABILITIES

The above is a form of orbital response for optimizing the SCF Wavefunction. Response can also be taken "how does my Wavefunction respond to an outside force"? When answering such questions we obtain useful molecular properties.

First let us compute the  $z$  dipole of a given molecule,

```
dipoles = mints.ao_dipole()
electric_dipole_z = 2 * np.vdot(tmp_dipoles[2], D)
nuclear_dipole_z = mol.nuclear_dipole()[2]
dipole_z = electric_dipole_z + nuclear_dipole_z
```

this procedure can be repeated for the  $x$  and  $y$  components as well.

To compute linear response terms we need gradient-like vectors  $C$  and  $D$ , in addition to the Hessian,

$$\langle\langle C, D \rangle\rangle = (C^{(1)})^T (\mathbf{E}^{(2)})^{-1} (D^{(1)})$$

The result is effectively the average value of property  $C$  under a perturbation on the Wavefunction by  $D$ . Since the Hessian is symmetric the response is the same as the average value of property  $D$  under perturbation by  $C$ .

To compute dipole polarizabilities,  $C$  and  $D$  can be replaced by the dipole tensors built above and the Hessian is the standard orbital response Hessian built in the second-order SCF. Keep in mind you need to compute all possible pairs of  $C$  and  $D$  where each span the  $x$ ,  $y$ , and  $z$  components, you will end up with 9 results in total  $xx$ ,  $xy$ ,  $xz$ ,  $yy$ , etc.

## REFERENCES

<sup>1</sup>T. Helgaker, P. Jørgensen, and J. Olsen, *Molecular electronic-structure theory*. John Wiley & Sons Inc, 2000.