

CHEM532 Programming Project #2: Hartree–Fock Theory

Francesco A. Evangelista

March 25, 2014

Introduction

In this programming project you will read the one- and two-electron integrals in the atomic basis and use them to compute the Hartree–Fock energy of a molecule.

The integrals are contained in two files. The file `one-electron` contains the one-electron integrals:

$$H_{\mu\nu} = (\mu|\hat{h}|\nu) = \int d\mathbf{r} \chi_{\mu}^*(\mathbf{r})\hat{h}(\mathbf{r})\chi_{\nu}(\mathbf{r}), \quad (1)$$

where $\hat{h}(1) = \hat{T}_e(1) + V_{eN}(1)$ contains both the kinetic energy and the electron-nuclei Coulomb interaction and $\{\chi_{\nu}(\mathbf{r})\}$ is the basis of atomic orbitals used to perform the computation.

The file `two-electron` contains the two-electron integrals in chemist notation:

$$(\mu\nu|\rho\sigma) = \int d\mathbf{r}_1 d\mathbf{r}_2 \phi_{\mu}^*(\mathbf{r}_1)\phi_{\nu}(\mathbf{r}_1)\frac{1}{r_{12}}\phi_{\rho}^*(\mathbf{r}_2)\phi_{\sigma}(\mathbf{r}_2). \quad (2)$$

In this project we will assume that the atomic orbitals are *real functions*, and therefore the electron integrals are real quantities.

Program outline

1. Read the nuclear-nuclear repulsion energy (in Hartree) from the file `vnn`.

2. Read the one-electron integrals, store them in memory, and print them to the output. The one-electron integrals are arranged as a $K \times K$ matrix, where K is the number of atomic basis functions. For example, in the case of the one-electron operator (**one-electron**):

$$\mathbf{H} = \begin{bmatrix} H_{0,0} & H_{0,1} & H_{0,2} & \cdots & H_{0,K-1} \\ H_{1,0} & H_{1,1} & H_{1,2} & \cdots & H_{1,K-1} \\ H_{2,0} & H_{2,1} & H_{2,2} & \cdots & H_{2,K-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ H_{K-1,0} & H_{K-1,1} & H_{K-1,2} & \cdots & H_{K-1,K-1} \end{bmatrix}, \quad (3)$$

while

$$\mathbf{S} = \begin{bmatrix} S_{0,0} & S_{0,1} & S_{0,2} & \cdots & S_{0,K-1} \\ S_{1,0} & S_{1,1} & S_{1,2} & \cdots & S_{1,K-1} \\ S_{2,0} & S_{2,1} & S_{2,2} & \cdots & S_{2,K-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_{K-1,0} & S_{K-1,1} & S_{K-1,2} & \cdots & S_{K-1,K-1} \end{bmatrix}. \quad (4)$$

The one-electron integrals can be read using the following python function:

```
def read_one_electron_integrals(filename):
    """Read the one-electron integrals contained in the file
       'filename' and return a numpy matrix"""
    ints_lines = open(filename,"r").readlines()
    n = len(ints_lines[0].split()) # determined the size of the matrix
    ints = zeros( (n,n) )
    for i,line in enumerate(ints_lines):
        for j,value in enumerate(line.split()):
            ints[i][j] = float(value)
    return ints
```

3. Read the two-electron integrals and store them in memory.

The file **two-electron** contains a list of the values of μ, ν, ρ, σ , and $(\mu\nu|\rho\sigma)$:

```

0 0 0 0    4.785065404705502
1 0 0 0    0.741380351973408
1 1 0 0    1.118946866342470
1 0 1 0    0.136873385354388
1 1 1 0    0.256633394730974
1 1 1 1    0.817206321526058
...

```

The two-electron integrals can be read using the following python function:

```

def read_two_electron_integrals(filename,nao):
    """Read the two-electron integrals contained in the file
    'filename' and return a 4-dimensional array that stores
    the integrals in chemistry notation as
    ints[m][n][r][s] = (mn|rs).
    This function needs to know the total number of atomic
    orbitals 'nao'."""
    ints_lines = open(filename,"r").readlines()
    ints = zeros( (nao,nao,nao,nao) )
    for line in ints_lines:
        line_split = line.split()
        p,q,r,s = [int(t) for t in line_split[0:4]]
        value = float(line_split[-1])
        ints[p][q][r][s] = ints[q][p][r][s] = value
        ints[p][q][s][r] = ints[q][p][s][r] = value
        ints[r][s][p][q] = ints[r][s][q][p] = value
        ints[s][r][p][q] = ints[s][r][q][p] = value
    return ints

```

4. Check that the integrals are read correctly by computing the sums:

$$a = \sum_{\mu\nu\rho\sigma} [(\mu\nu|\rho\sigma)]^2, \quad (5)$$

and

$$b = \sum_{\mu\nu\rho\sigma} |(\mu\nu|\rho\sigma)|. \quad (6)$$

5. Form the $\mathbf{X} = \mathbf{S}^{-1/2}$ matrix. Begin by diagonalizing the matrix \mathbf{S} :

$$\mathbf{S}\mathbf{L} = \mathbf{L}\mathbf{s}, \quad (7)$$

where \mathbf{L} and \mathbf{s} are respectively the matrices of eigenvectors and eigenvalues of \mathbf{S} . Form $\mathbf{S}^{-1/2}$ as

$$\mathbf{X} = \mathbf{S}^{-1/2} = \mathbf{L}\mathbf{s}^{-1/2}\mathbf{L}^T, \quad (8)$$

where the matrix elements of the matrix $\mathbf{s}^{-1/2}$ are

$$(\mathbf{s}^{-1/2})_{\mu\nu} = \begin{cases} \frac{1}{\sqrt{s_{\mu\mu}}} & \text{if } \mu = \nu \\ 0 & \text{else} \end{cases}. \quad (9)$$

6. Show that the \mathbf{X} matrix is correct by computing the quantity

$$\mathbf{Y} = \mathbf{X}^T\mathbf{S}\mathbf{X}. \quad (10)$$

The matrix \mathbf{Y} should look like the identity matrix (up to numerical errors of the order of 10^{-16} Hartrees).

7. Initialize the computation.

Set the initial density matrix to zero:

$$D_{\mu\nu} = 0 \text{ for } \mu, \nu = 0, \dots, K-1. \quad (11)$$

Set the iteration count $k = 0$.

Set the energy at iteration $k = 0$: $E^{(0)} = 0$.

8. Form the G matrix and increase the iteration counter k .

$$G_{\mu\nu} = \sum_{\rho\sigma}^K D_{\rho\sigma} [2(\mu\nu|\rho\sigma) - (\mu\rho|\nu\sigma)] \quad (12)$$

and

$$k \leftarrow k + 1. \quad (13)$$

9. Form the Fock matrix and compute the total energy.

$$F_{\mu\nu} = H_{\mu\nu} + G_{\mu\nu}, \quad (14)$$

The total energy for cycle k can be expressed as:

$$E^{(k)} = V_{\text{NN}} + \sum_{\mu\nu}^K D_{\mu\nu} (H_{\mu\nu} + F_{\mu\nu}). \quad (15)$$

10. Transform the Fock matrix to the basis of orthonormal atomic orbitals:

$$\mathbf{F}' = \mathbf{S}^{-1/2} \mathbf{F} \mathbf{S}^{-1/2}. \quad (16)$$

11. Find the eigenvalues and eigenvectors of \mathbf{F}' . Solve the eigenvalue problem:

$$\mathbf{F}' \mathbf{C}' = \mathbf{C}' \boldsymbol{\epsilon}. \quad (17)$$

12. Transform the coefficient matrix \mathbf{C}' to the atomic orbital basis.

$$\mathbf{C} = \mathbf{X} \mathbf{C}'. \quad (18)$$

13. Form the density matrix.

$$D_{\mu\nu} = \sum_i^{N/2} C_{\mu i} C_{\nu i}. \quad (19)$$

14. Compute the energy error as the difference between the energy at iteration k and $k - 1$:

$$E^{(k)} - E^{(k-1)} = \Delta E^{(k)}. \quad (20)$$

At the same time compute the root mean square error of the density matrix:

$$|\Delta \mathbf{D}^{(k)}| = \left\{ \sum_{\mu\nu}^K [\mathbf{D}_{\mu\nu}^{(k)} - \mathbf{D}_{\mu\nu}^{(k-1)}]^2 \right\}^{1/2}. \quad (21)$$

15. Repeat steps 8–13 until convergence. At convergence, the absolute value of this quantity must be smaller than the energy convergence threshold $\tau_E = 10^{-9}$ Hartree:

$$|\Delta E^{(k)}| < \tau_E, \quad (22)$$

and at the same time $|\Delta \mathbf{D}^{(k)}|$ must be less than the density matrix convergence threshold $\tau_D = 10^{-5}$. Stop the iterative process **if both these conditions are satisfied**.

Compare your results with the sample output file. If the converged energy matches, pat yourself on the back, you just coded Hartree–Fock theory!