Numerical Algorithms Applied to Computational Quantum Chemistry

Homework 4: Implementation of Complete Neglect of Differential Overlap 2 CNDO/2 Method for Molecular Calculations

---

Link to Github Repo: https://github.com/Mattwaseem/Chem279-Numerical-Algos-HW4
Repo: Publicly Available & TA invited.
Username: **MattWaseem**

—

Assignment Overview:

The goal here was to extend concepts from previous homework (EHT, Matrix Overlap calculations etc..) to implement the new approach CNDO method. This is a semi-empirical Hartree Fock calculation. The key steps for this assignment included:

1. Parsing molecule inputfile (code recycled from previous homework)
2. Build the CNDO/2 Fock Matrix
3. Solve Self Consistent Field equation (equation 1.4)
4. Calculate the total energy

In similar manner to previous homeworks the assignment was organized in two sections with a supplementary section.

Problem 1: Correctly Building the CNDO/2 Fock Matrix

Problem 2: Calculating the SCF equation with the open shell CNDO/2

Problem 3: Use the CNDO/2 on other more complex molecules such as N2 and O2

The best way to approach these problems was to modularize the code, use previous homework code, so that subsequent tasks can become easier to accomplish. To begin below is the directory structure for the assignment. For the purpose of this discussion files within each directory will not be listed, please refer to my github for file specific details. This is to get a broad overview picture of the directory structure and for ease of navigation.

```
.
├── build
├── calculated_outputs
├── include
├── sample_input
├── sample_output
├── src
└── CMakeLists.txt
```

How to Compile and Run:

1. Ensure you have armadillo library installed in your system
2. Also make sure you have the input files under the input directory.
3. CD to build directory (if there is not one, mkdir build, please).
4. Once in build directory, run the following command to use CMakeList.txt to compile code:
    a. Cmake ..
    b. Make
    c. Make run
5. Results from the terminal should be generated in the calculated_output directory for each input file. Each Input file will have a corresponding output file with a similar name.
6. Once, satisfied with response, you can run the following to clean all files and artifacts:
    a. Make clean-all

Problem 1: Building The CNDO/2 Fock Matrix for Simple Molecule

To implement the CNDO/2 method, the first step was to build the Fock matrix by correctly parsing the input files to obtain the molecular coordinates. The input file parser was designed to handle both closed-shell and open-shell molecules. After parsing the files, the correct number of atomic orbitals was determined based on how each atom contributes a specific number of atomic orbitals corresponding to their valence electrons.

In the CNDO/2 method, a minimal basis set consisting only of valence electrons is used. The total number of basis functions (**N**) was calculated by summing the contributions from each atom. The atomic orbital information was stored to facilitate the calculation of overlap integrals (using the overlap matrix code from previous homework) and Fock matrix elements.

The overlap matrix elements between atomic orbitals were computed. Calculating the overlap integrals was essential for solving the generalized eigenvalue problems in the SCF procedure in

the second part of the assignment. The density matrices for alpha and beta spins were also initialized and used as the starting point for the iterative SCF process.

Lastly, the CNDO/2 Fock matrix was built using several empirical parameters and approximations. The diagonal elements were calculated using the following formula:

$$f_{\mu\mu}^{\alpha} = -\frac{1}{2}\left(I_{\mu} + A_{\mu}\right) + \left[\left(p_{AA}^{\mathrm{tot}} - Z_A\right) - \left(p_{\mu\mu}^{\alpha} - \frac{1}{2}\right)\right]\gamma_{AA} + \sum_{B \neq A}\left(p_{BB}^{\mathrm{tot}} - Z_B\right)\gamma_{AB} \qquad (1.4)$$

where:

- Fμμα is the diagonal element of the alpha-spin Fock matrix.
- IμI and Aμ are the ionization potential and electron affinity of orbital muμ.
- Pμμα is the diagonal element of the alpha-spin density matrix.
- P_AAtot is the total electron density on atom A.
- ZA is the valence atomic number of atom A.
- γAA is the two-electron repulsion integral for atom A.

The off Diagonal elements were calculated using the following:

$$f_{\mu\nu}^{\alpha} = \frac{1}{2}\left(\beta_A + \beta_B\right) s_{\mu\nu} - p_{\mu\nu}^{\alpha}\gamma_{AB} \qquad (1.5)$$

where:

- βA and Bβ are the bonding parameters for atoms A and B.
- Sμv is the overlap matrix element between atomic orbitals muμ and nuv.
- Pμvα is the off-diagonal element of the alpha-spin density matrix.
- γAB is the two-electron repulsion integral between atoms A and B.

The Gamma values were determined based on:

$$\gamma AB = \frac{14.397}{R_{AB} + \frac{1}{2}(1/UA + 1/UB)} \qquad (1.6)$$

Finally, unit conversions were performed to obtain all energies in electron volts (eV) from atomic units. The Fock matrix was built, and the calculated outputs were written to corresponding output files with names similar to the input files (e.g., `H2.txt` → `H2_calculated_output.txt`).

Overall, this process established the foundational steps in performing semi-empirical Hartree-Fock calculations, bridging the molecular input data with the SCF process to calculate the total energies of the system.

Problem 2: Evaluating The Open Shell CNDO/2 Self Consistent Field Equation

The Fock Matrix from problem one was used to build the code for f alpha and f beta in order to develop the self consistent field (SCF) algorithm for the CND/2 method. This did involve solving the SCF equation in the atomic orbital (AO) basis for both the alpha and beta spins:

$$\mathbf{f}^{\alpha}\mathbf{c}^{\alpha} = \mathbf{c}^{\alpha}\boldsymbol{\epsilon}^{\alpha} \tag{2.1}$$

$$\mathbf{f}^{\beta}\mathbf{c}^{\beta} = \mathbf{c}^{\beta}\boldsymbol{\epsilon}^{\beta} \tag{2.2}$$

Where:

- Fα and fβ are the Fock matrices for alpha and beta spins, respectively.
- Cα and cβ are the molecular orbital (MO) coefficient matrices.
- Episilon-α and ϵβ are the diagonal matrices of orbital energies (eigenvalues).

Implementation of the SCF algorithm was done by first taking an initial guess. The density matrices for alpha and beta spins were initialized to zero matrices. The Fock Matrix from the previous problem was constructed using the f alpha and f beta spin for the current density matrices. The eigenvalue problem was solved to get the molecular orbital coefficient and eigenvalues using equation 2.1 and 2.2.Once the eigenvalue problem was solved using Armadillo library the density matrices were updated based on the following:

$$P^{\alpha}_{old} = P\alpha$$

$$P^{\alpha}_{old} = P^{\beta}$$

New density matrices were assembled by occupying the ppp lowest energy alpha MOs and the qqq lowest energy beta MOs, using:

$$p_{\mu\nu}^{\alpha} = \sum_{i}^{p} c_{\mu i}^{\alpha} c_{\nu i}^{\alpha} \tag{1.1}$$

$$p_{\mu\nu}^{\beta} = \sum_{i}^{q} c_{\mu i}^{\beta} c_{\nu i}^{\beta} \tag{1.2}$$

$$p_{\mu\nu}^{\text{tot}} = p_{\mu\nu}^{\alpha} + p_{\mu\nu}^{\beta} \tag{1.3}$$

The maximum magnitude of the change in the density matrices was calculated and the total energy was calculated upon convergence, where the total energy was calculated using:

$$E_{\text{CNDO/2}} = \frac{1}{2}\sum_{\mu\nu} p_{\mu\nu}^{\alpha}\left(h_{\mu\nu} + f_{\mu\nu}^{\alpha}\right) + \frac{1}{2}\sum_{\mu\nu} p_{\mu\nu}^{\beta}\left(h_{\mu\nu} + f_{\mu\nu}^{\beta}\right) + \sum_{A}\sum_{B<A}\frac{Z_A Z_B}{R_{AB}} \tag{2.5}$$

The implementation of the open shell CNDO/2 was successful. However, the output of the total energies did not exactly correlate with the expected energy as what the sample output files had suggested. This shows that there are possible errors in the calculation error in my implementation for which I did not fully have a chance to reconcile and fix. Thus far, I think the implementation was successful and the total energy was calculated, though it can use some improvements.

Problem 3: GOING FURTHER Building CNDO/2 for N2 and O2:

The extension of the CNDO/2 code to N_2 and O_2 molecules provided valuable insights into my understanding of the chemical properties and validated the code's capability to handle different electron configurations. Although the energy total did not match the experimental results on the web, the overall ability to build the CNDO/2 for other systems shows the flexibility of my current implementation. The simulations show key characteristics such as bond strength, bond order, electronic structure, and magnetic properties.

- **N_2 Simulation:**
  - Strong triple bond and high stability were reflected in the results with a higher energy output in the N2_calcualted_output.txt file.
  - The code accurately represented a closed-shell diatomic molecule.
- **O_2 Simulation:**
  - Open-shell configuration and paramagnetism were correctly modeled.
  - The simulation demonstrated the code's ability to handle unpaired electrons.

Overall, the CNDO/2 code proved to be a useful tool for exploring molecular properties and behaviors, providing a solid foundation for further computational chemistry studies.