UNIVERSITY OF CALIFORNIA BERKELEY

Numerical Algorithms Applied to Computational Quantum Chemistry
Class Project Guidelines and Suggestions.

Prof. Daniel Levine, Prof. Martin Head-Gordon and Rebecca Tomann

Due Dec. 15, 2023

1 GENERAL INSTRUCTIONS.

To finish the semester, we would like you to either further develop the SCF program you wrote for CNDO/2 or the non-iterative extended Huckel method that you developed, or write something entirely new. Here is your chance to go beyond structured homeworks and exercise your creativity in some scientific programming! This should be the equivalent of 1-2 homework sets in terms of effort, but we know different people have different time constraints, so we'll work with you to try to make sure that you are choosing something achievable.

We do want each of you to produce something that you are proud of, that reflects what you've learned throughout the degree so far, and takes you further. Also, since this class has had problem sets that build cumulatively on each other, by building further with the project, you may even be able to take your code and use or further develop it in Prof. Demmel's parallel computing class next semester!

Here is the schedule for our class project:

1. **November 18:** Select a topic, in consultation with Rebecca and myself – see choices below.
2. **December 7:** Preliminary progress report. This will be a 10 minute discussion with Rebecca and myself on December 7 to describe your initial progress and plans.
3. **December 15:** Present your theory, design, code, tests, and results to the class in a 20 minute talk, allowing at least 5 minutes for questions. We will do presentations on Dec. 15 and possibly the day before, and conceivably another day if needed...

2 CHOOSING A PROJECT.

The main idea of the project is for you each to do something different and individual. The topic can be something drawn from the class topics, or an extension of them. Ideally it should be something that interests you, and makes you want to go further! A list of potentially suitable topics are given below. They are not all equally difficult, so a rough estimate of how many homework sets each one is equivalent to is given. Regardless of the level of difficulty, we will award bonus points for outstanding achievements in the project! You can either choose one of the topics below, or use the list to get you thinking about your own project suggestion. Regardless, we want everyone to have their own project, so once a topic is selected by someone, it will no longer be available.

1. Extend your code to perform vibrational analysis to obtain the vibrational frequencies and the normal modes of a molecule at its optimized geometry: see ICC pp. 555-556. You will evaluate the hessian matrix by finite differences, transform to mass-weighted coordinates, and diagonalize. The resulting vibrational frequencies can be compared against experimental infrared results. (1 – 2 homeworks)

2. Improve the non-linear optimization aspects of your code. First implement the DIIS method to accelerate convergence of your SCF equations. Then improve your geometry optimizer by implementing a quasi-Newton method such as BFGS to search for stable structures. (1 – 2 homeworks)

3. Implement your own replacement for Armadillo for the matrix operations that we need. The main challenge will be writing your own diagonalizer! There is a good discussion of methods that can yield all the eigenvalues and eigenvectors of a a symmetric matrix in Numerical Recipes and elsewhere. In addition you must also build suitable code infrastructure. (1 – 2 homeworks)

4. Extend your Huckel code to include the pairwise atomic corrections needed to obtain high accuracy for hydrocarbons (see the paper by Voityuk posted under reading material), and implement the gradient. Get some results and compare to your CNDO/2 code for accuracy of structures and relative energies.

5. Experiment with extending the CNDO/2 model to include overlap. Make a new version of your code for energies (and possibly gradients) that explicitly accounts for overlap. See how results for relative energies (and structures if you do the gradient) compare. Maybe you can even try adjusting the parameters to improve accuracy for a given class of molecules such as the hydrocarbons (look at the Voityuk paper mentioned in the project above, for instance).

6. Implement a (bare-bones) version of Hartree-Fock using integrals computed by recurrence relationships (Prof. Levine can provide more details on these). Compare the integral quantities obtained with those you have already implemented. Compare thermochemistry (energy differences) of a few molecules with your CNDO/2 code.

7. Make your CNDO/2 code efficient enough to treat very large molecules by eliminating overhead and optimizing the construction of **h** and **f** so that the computation time is dominated by the linear algebra of diagonalizing **f**. Demonstrate with timings and profiling on molecules that range up to 1000 atoms. (1 – 2 homeworks)

8. Feel free to suggest anything else that interests you in scientific programming that connects to our class! We will try to stop you from doing something where you may not get results in time, or help you rescope appropriately.