

# Agenda

Patryk Kozlowski

September 26, 2023

## 1 Introduction

It has been over 2 years since I have done some serious academic work. I have gotten comfortable with the AI tools and I am excited to put them to use!

## 2 Academic Advising

To refresh your memory, I am taking the senior thesis with you, the computational chemistry lab class with Professor Goddard (time to use pyscf!), and Ph127a with Professor Motrunich (Statistical Physics of Interacting Systems, Phases, and Phase Transitions).

## 3 Senior Thesis

### 3.1 Content

We previously spoke about this being a programming project. However, I know that it can also be done through reading primarily. While I do like the voice coding, will I be missing out on things if I am not doing reading?

### 3.2 Context of the FCI starter project

Back in August, I was thinking of devoting some time into looking into the pyscf implementation of FCI. However, I was asking:

I want to spend a little time now learning about how pyscf works ... to learn what an optimal implementation of FCI could look like. Do you have any recommendations as to what I should bring my attention to? Even though this is only an example, `pyscf/examples/fci/01-given_h1e_h2e.py` seems similar to the structure of what I implemented with `h1e.npy` and `h2e.npy`. `pyscf/fci/cistring.py` does have a lot of functions whose function I recognize, which may be a sign that I learned something :)

To which Rui responded:

Do you really need to learn pyscf algorithm for FCI? It's not like it's a bad choice, but it still feels to me you have something easier and more significant you can learn.

I have been wondering what is the meaning of this last sentence for me? Is there a bridge between FCI and real quantum chemistry development or is it recommended to just try and jump right into it?

### **3.3 How to proceed**

Ideas?

#### **3.3.1 Addendum to what he said**

Optimization of the code requires much more effort than having a good mathematical representation of the algorithm - e.g. mastering C/C++ codes, knowledge of memory management, SIMD techniques, math libraries, latencies, thread parallelization, etc., etc., etc. and etc. No intention to scare you, but don't underestimate the difficulty of reading the optimized code. Usually it would be 1-3 months without any positive feedback.