A Heuristics-Based Approach to the Prediction of FODs in Fermi-Lowdin Orbital Self Interaction Correction (FLOSIC) Theory

Angel-Emilio Villegas Sanchez

A thesis presented for the degree of

Master's of Science

Supervised by:

Dr. Juan Ernesto Peralta

Dr. Koblar Alan Jackson

Central Michigan University

February 2024

Table of Contents

Chapter 1

Introduction

1.1 Density-Functional Theory (DFT)

1.1.1 Hohenberg-Kohn Theorems

1.1.2 Kohn-Sham Formulation

1.1.3 Limitations of DFT

1.2 The Self-Interaction Error (SIC)

1.2.1 Perdew-Zunger SIC

1.3 The FLOSIC Method

1.3.1 The Fermi-Lowdin Orbital (FLOs)

1.3.2 The Fermi-Lowdin Orbital Descriptors (FODs)

1.4 The NRLMOL Code

- Describe the code and status of usability

1.4.1 Overview of Code

- Big-picture description of how the code works.

1.4.2 FOD Determination and Practical Considerations

- Describe methods used to input FODs into the FLOSIC program.
- Prime motivation behind this project: an automated and faster process to begin calculations, informed from "empirical" data, capable to extending to larger systems.

1.5 FOD Geometric Motifs

- Describe how FODs arrange themselves similar to LDQ Theory. This (emergent?) behavior makes it

1.5.1 SP3 Arrangements

1.5.2 SP3D Arrangement

1.5.3 Dominance Schema

- Sometimes FODs are closer to atoms that are of lesser atomic number Z than the neighbouring heavier atoms. This could be attributed to shielding caused by the filled shells of periods 3 and above. The behavior is described in Figure 1.1. This will play a special part in the prediction of bonding FODs.

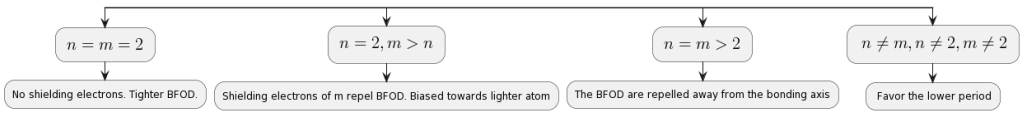

Figure 1.1: dominance

Chapter 2

The FOD-Lego Program

- Note: Different name needed probably

2.1 General Overview

- Give a **Big-Picture** overview as seen in image Figure 2.1

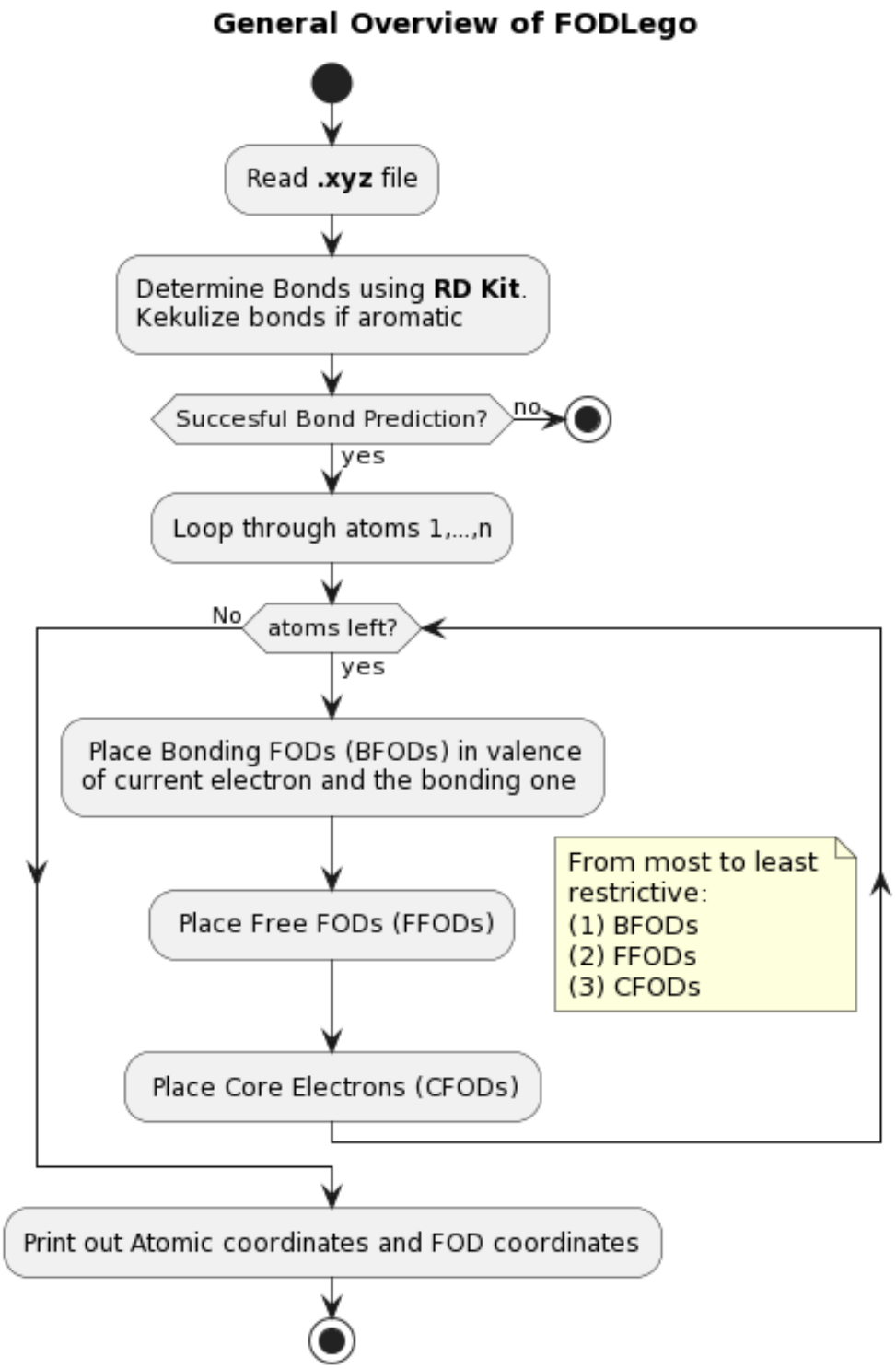

Figure 2.1: overview

2.1.1 A Heuristic-Based Approach to FOD Placement

- How can Heuristics help in computational problems?

2.1.2 Monoatomic Calculations

- Talk about *Dynamic Programming* and how monoatomic geometries could be pieced together in modular fashion (Lego-like).

2.1.3 Reparametrizing the FOD geometries

- sp3 shells stop being perfect tetrahedra and become irregular in the presence of other atoms.
- Describe how to parametrize FODs in irregular tetrahedra for the sp3 shells.

2.1.4 Implementation Details

- Describe code structure and use of object-oriented design
- Describe fundamental as seen in Figure 2.2

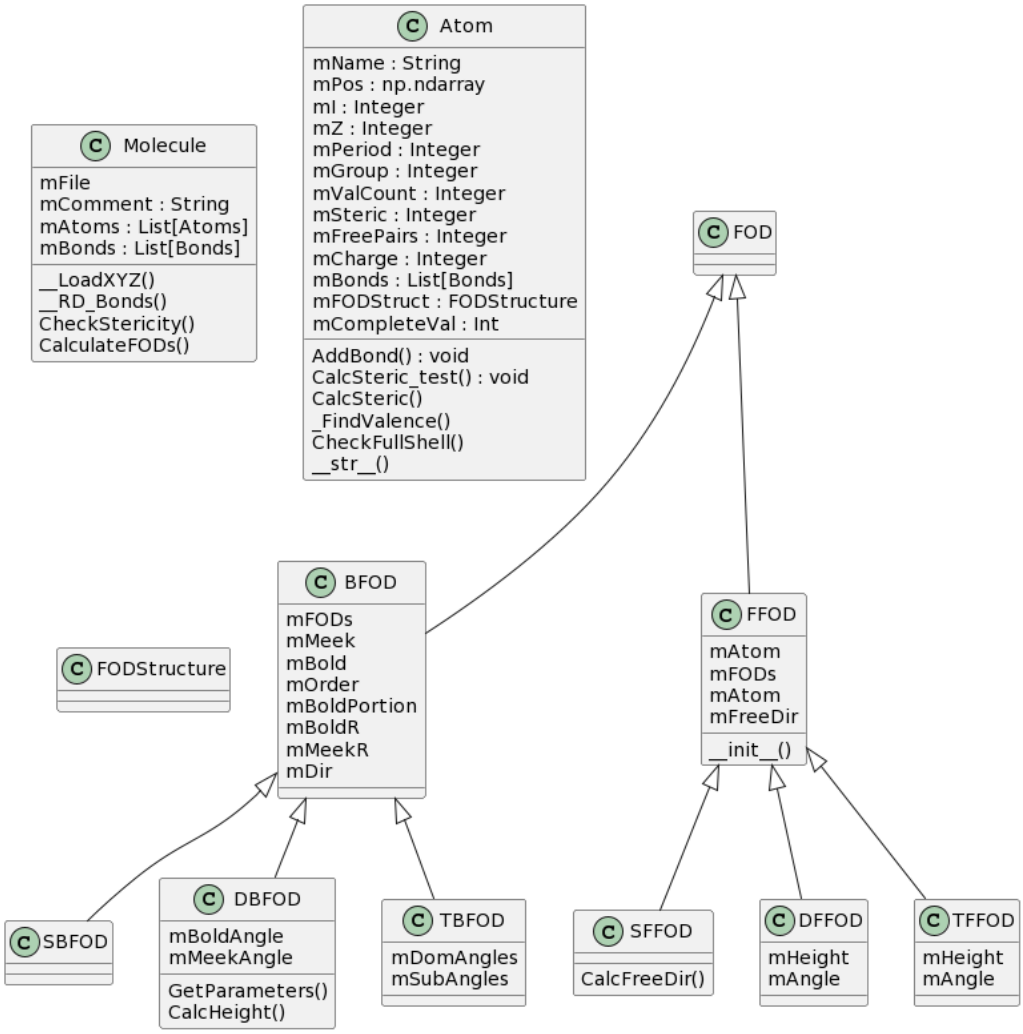

Figure 2.2: classes

2.2 Determination of FODs

- Describe the different determination heuristics for different types of FODs

2.2.1 Bonding FODs (BFOD)

2.2.2 Core FODs (CFOD)

2.2.3 Free FODs (FFOD)

- Sample Diagram and calculation for Double-FFOD (DFFOD) Heuristic in Figure 2.3. The edge distance is applied to as a restriction.

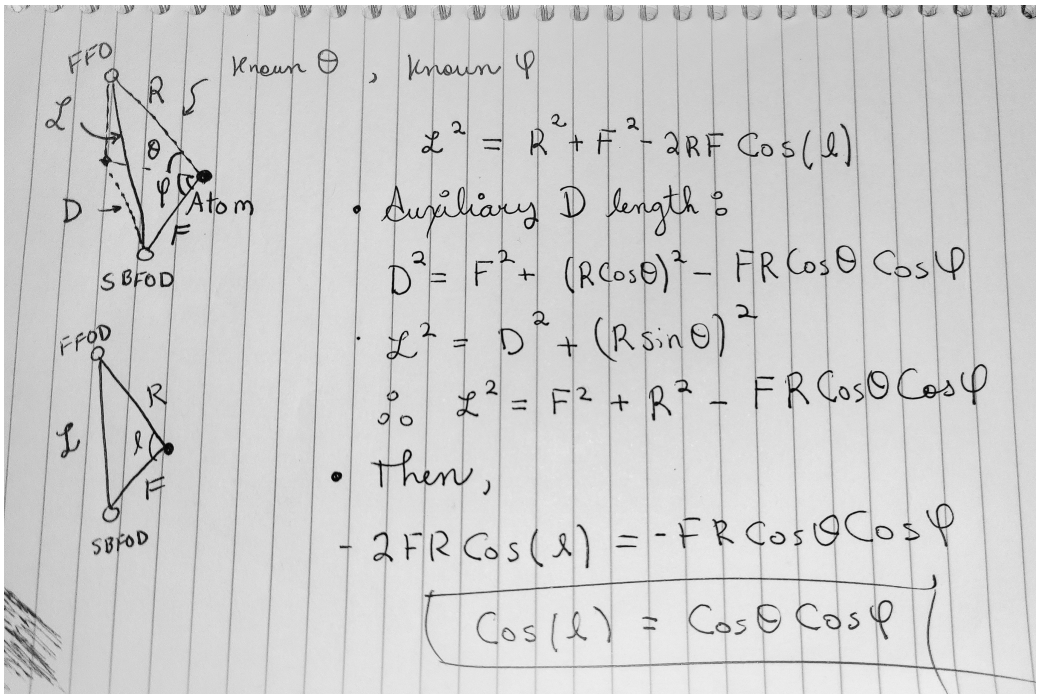

Figure 2.3: draft DFFOD

2.3 A Data-driven Approach to analyzing FODs

2.3.1 Reverse Determination of Parameters

% Use the program to determine the parameters from FLOSIC-optimized FODs. Now we can use angles and relative distances (FOD-FOD distances that are part of a Shell schema) to better characterize FOD arrangement. This is the following implementation used for the reverse determination of parameters

Algorithm 1: Reverse Parameter Determination Basic Sequence

Data: Relaxed FODs R_n

for *FOD in Predicted FODs* **do**

  Save closest R_n to FOD as R_{close}

  Initialize a new FOD with R_{close}

end

2.3.1.1 Free FODs (FFODs)

The free-direction, described in section XYZ, cannot be determined exactly. In order to best compare Free FODs in this parameter-based scheme, I propose using only an average direction formed from the average of all FFOD vectors and use an evaluation function. A cosine distance is too flat near its maximum, so a custom function can offer a more strict metric of our prediction

$$f(\theta) = 1 - \frac{4\theta}{\pi} \tag{2.1}$$

Function Equation **??** yields 1 when there at 0 degrees between the vectors, and 0 when there are 45 degrees between the vectors.

2.3.2 A database of compiled FOD Data

- No current database of FODs. Seeking to create a database
- **TODO!**
  - Design structure of database
  - Using FODLego, then FLOSIC, create optimized structures for a selection of molecules (scope?)

Chapter 3

Validity of FOD Prediction

Explore the predictions done by FODLego

3.1 Comparison of Heuristic FODs to FLOSIC-Optimized FODs

- **TODO:**
  - Create scatter plots for different parameters described in the scheme of previous section.

3.2 Comparison to FOD-MC

3.3 Biasing Predictions after Sampling (?)

- – Assuming data follows certain trends, we can further modify the predictions used in FODLego. For example, the tightening of closed shells in heavier atoms in the presence of the next shell closed.

Chapter 4

Discussion/Conclusions

4.1 Limitations

4.1.1 Assumptions

- Explore assumptions such as First neighbors are mainly considered in this scheme.

4.1.2 Automated Bonding Prediction

4.2 For Future Implementation

4.2.1 Extending modularity to metal FODs?

- Talk about possible parametrization of the *Triaugmented Triangular Prism* composed of 9 FODs.

4.2.2 Spin-Unpolarized Prediction