EASYPARM

User manual

Version 1.00 – September 2024

Abdelazim M. A. Abdelgawwad

abdelazim.abdelgawwad@uv.es

Institut de Ciència Molecular, Universitat de València, P.O. Box 22085, València 46071, Spain

Table of Contents

1. Introduction

easyPARM is a computational tool with two primary functions. First, it derives bond-stretching and bond-angle bending force field parameters for metal-containing molecular systems using the Seminario method.[1] This ensures accurate parameter generation tailored for the AMBER software suite[2], based on the Hessian matrix produced by frequency calculation in Gaussian software.[3]

Second, easyPARM enables charge restraints on specific atoms using the Restrained Electrostatic Potential (REsP) fitting approach.[4] This helps achieve a more accurate electrostatic potential around the molecule while keeping charge distribution physically meaningful.

With these capabilities, easyPARM is an efficient tool for accurately modeling metal-containing systems, ensuring both structural and electrostatic precision.

The easyPARM code was written by Abdelazim M. A. Abdelgawwad under supervision Dr. Antonio Francés-Monerris. This code version 1.00 is free: you can redistribute it and/or modify it under the terms of Creative Commons Attribution 4.0 International.

1.1 Key Features

- Generates parameters for metal-containing systems using a combination of the general AMBER force field (GAFF) or the AMBER force field and the missing parameters for bond stretching and angle bending involving metals and linked atoms with Seminario method
- Capable of generating parameters for entire structures in non-metal systems using either GAFF or AMBER
- Outputs essential files (frcmod, pdb, mol2, and lib) for initiating simulations or preparing metal-containing systems

1.2 Citation

Please cite the following references if you use the easyPARM code:

- Paper/preprint
- Github (https://github.com/Abdelazim-Abdelgawwad/easyPARM.git)

2. Installation

2.1 Prerequisites

To successfully install and run easyPARM, ensure that the following prerequisites are met:

- Python 3 or higher: easyPARM requires Python version 3 or above to function properly. You can download the latest version of Python from the official Python website.
- periodictable package: This package provides access to the periodic table of elements, which is essential for the metal-specific calculations in easyPARM. To install it, use the following command in your terminal or command prompt:

```
pip install periodictable
```

- scipy package: SciPy is a scientific computing library that easyPARM utilizes for various numerical calculations, including those related to force fields and molecular mechanics. To install SciPy, use the following command:

```
pip install scipy
```

Additional package requirements may arise during execution. If prompted, install them via pip or contact us for assistance.

Once the necessary dependencies are installed, you'll be ready to run easyPARM to generate the required force field parameters.

2.2 Running easyPARM

Choose one of the following methods:

1. **Run the script directly**: Navigate to the directory containing the easyPARM script and execute the following command.

```
./easyPARM.sh
```

2. **Set up an alias for easy access**: To run easyPARM from any location in your terminal, you can create an alias by adding the following line to your .bashrc file:

```
alias easyPARM='/full/path/to/easyPARM.sh'
```

3. Usage Guide

3.1 Required Input Files

1. Checkpoint File (.chk): Contains the Hessian matrix from a Gaussian calculation.
2. Optimized Structure (XYZ Format): Contains the optimized molecular geometry.
3. Gaussian Charge Output: Contains charge distribution information (ESP charge).

3.2 Step-by-Step Tutorial

1- AMBER Configuration

The code will prompt you to confirm whether AMBER is already loaded or if you need to specify its path. You will see the following prompt:

Select your option:
1- Amber is already loaded
2- Provide the Amber path
Enter your choice (1 or 2):

- If you can use the module load amber command, select option 1.
- If you prefer to manually provide the AMBER installation path, select option 2.

Option 1:

Enter your choice (1 or 2): 1
Amber is assumed to be already loaded. Skipping sourcing.

Option 2:

Enter your choice (1 or 2): 2
Please provide the path for Amber:

2- System Charge.

Please provide the total charge:

3- mol2 File Generation

After setting the total charge, the code will prompt you to select the input format for generating the mol2 file. You'll be presented with the following options:

Please select the input format:
1- Gaussian output (log)
2- Resp (gesp)
3- PDB structure (pdb)
Enter the number corresponding to the input format:

Next, you'll need to select the charge method for the system. The recommended method is **RESP**, but other options are available:

Please select the charge method (recommended: RESP):
1- RESP (resp)
2- Mulliken (mul)
3- ESP (esp)
4- AM1-BCC (bcc)
Enter the number corresponding to the charge method:

Finally, you'll be asked to select the atom type for the system. The recommended choices are **GAFF** or **AMBER**, but you can also choose **Sybyl**:

Please select the atom type:
1- Amber Force Field (AMBER)
2- General Amber Force Field (GAFF)
3- General Amber Force Field (GAFF2)
Enter the number corresponding to the atom type:

After selecting the atom type, the code will prompt you to provide the output file that contains the charge information. The accepted file formats include .log, .resp, and .pdb. The prompt will look like this:

Please provide the charge output file (e.g., .log, .resp, .pdb):

4- Seminario Method Setup

You must provide the checkpoint file required to run the **Seminario method**. The checkpoint file should have a .chk extension.

Please provide the checkpoint file (.chk):

5- Checking the availability of Gaussian software

The code will next verify the availability of the Gaussian software, which is necessary for certain computational tasks. You will need to confirm whether Gaussian is already loaded on your system or if its path needs to be specified manually:

Select your option:
1- Gaussian is already loaded (formchk is available)
2- Provide the Gaussian path
Enter your choice (1 or 2):

6- Providing the Optimized Structure for Seminario Method

To proceed with the Seminario method, you will need to provide the optimized XYZ geometry file. The code will prompt you to enter the name of this file:

Please provide the optimized XYZ geometry file:

7- Output Generation

After successfully completing all the preceding steps and ensuring that all required inputs are correctly provided, the software will generate the necessary parameter files for further use. Upon completion, you will receive a set of output files that may include:

Here is your mol2 COMPLEX.mol2
Here is your frcmod COMPLEX.frcmod
Here is your pdb COMPLEX.pdb
Here is your lib COMPLEX.lib

8- Atomic Charge Restraint (Optional)

The program provides the option to restrain the atomic charge on specific atoms. Upon execution, the following prompt will appear:

Would you like to restrain the charge on specific atoms? (Yes or No):

- Selecting No will terminate the operation.
- Selecting Yes will allow you to provide details for one or more atoms

8.1 Specify the Number of Atoms

If you choose to proceed, the following prompt will appear:

How many atoms do you want to restrain?

At this point, you will be required to specify the exact number of atoms for which the charge is to be restrained.

8.2 Providing Atom-Specific Information

You will see the following prompt:

Please provide the atom number and its charge for atom 1 (e.g., 12 -0.834):

You are required to input both the atom number and its corresponding charge for each selected atom.

8.3 Inputting Gaussian Output

You will see the following prompt:

Please enter the name of Gaussian output that contains all the ESP charges:

Here, you must specify the file name containing the Electrostatic Potential (ESP) charges from the Gaussian output.

8.4 Generating the Output

After processing, a new mol2 and library file will be generated, containing the updated atomic charges based on the information you provided.

4. Appendix

Example for Generating the Required Files Using Gaussian

When using Gaussian to generate the required files, note that the choice of the level of theory is up to the user and is not mandatory to match the examples provided. Follow the steps below:

4.1 Structure Optimization and Frequency Calculation

First, optimize the molecular structure and run a frequency calculation. Upon completion, you will obtain the optimized structure in .xyz format, along with the checkpoint file (.chk).

```
%mem=30GB
%nproc=24
%chk=output.chk
#p B3LYP/def2tzvp symmetry=none opt scf=(MaxCycle=60,xqc)
freq(noraman) int(ultrafinegrid)

#Comment

charge multiplicity
coordinates
```

4.2 Charge Calculation Using the RESP Method

Next, calculate the molecular charge using the RESP method in Gaussian. This step will produce the charge output in the .log file.

```
%mem=30GB
%nproc=24
#p B3LYP/gen pseudo=read SCF=tight Pop=(mkuff) iop(6/33=2)
iop(6/42=6) iop(6/50=1)

#Comment

charge multiplicity

coordinates

C N H 0
6-31G*
****

Ru 0
SDD
****

Ru 0
SDD

output.resp

output.resp
```

5. References

(1) Seminario, J. M. Calculation of Intramolecular Force Fields from Second-Derivative Tensors. *Int J Quantum Chem* **1996**, *60* (7), 1271–1277. https://doi.org/https://doi.org/10.1002/(SICI)1097-461X(1996)60:7<1271::AID-QUA8>3.0.CO;2-W.

(2) Case, D. A.; Aktulga, H. M.; Belfon, K.; Ben-Shalom, I. Y.; Brozell, S. R.; Cerutti, D. S.; Cheatham III, T. E.; Cisneros, G. A.; Cruzeiro, V. W. D.; Darden, T. A.; Duke, R. E.; Giambasu, G.; Gilson, M. K.; Gohlke, H.; Goetz, A. W.; Harris, R.; Izadi, S.; Izmailov, S. A.; Jin, C.; Kasavajhala, K.; Kaymak, M. C.; King, E.; Kovalenko, A.; Kurtzman, T.; Lee, T. S.; LeGrand, S.; Li, P.; Lin, C.; Liu, J.; Luchko, T.; Luo, R.; Machado, M.; Man, V.; Manathunga, M.; Merz, K. M.; Miao, Y.; Mikhailovskii, O.; Monard, G.; Nguyen, H.; O'Hearn, K. A.; Onufriev, A.; Pan, F.; Pantano, S.; Qi, R.; Rahnamoun, A.; D.R. Roe; Roitberg, A.; Sagui, C.; Schott-Verdugo, S.; Shen, J.; Simmerling, C. L.; Skrynnikov, N. R.; Smith, J.; Swails, J.; Walker, R. C.; Wang, J.; Wei, H.; Wolf, R. M.; Wu, X.; Xue, Y.; York, D. M.; Zhao, S.; Kollman, P. A. Amber 20. *Amber 20* **2021**, San Francisco.

(3) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. a.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Petersson, G. a.; Nakatsuji, H.; Li, X.; Caricato, M.; Marenich, a. V.; Bloino, J.; Janesko, B. G.; Gomperts, R.; Mennucci, B.; Hratchian, H. P.; Ortiz, J. V.; Izmaylov, a. F.; Sonnenberg, J. L.; Williams; Ding, F.; Lipparini, F.; Egidi, F.; Goings, J.; Peng, B.; Petrone, A.; Henderson, T.; Ranasinghe, D.; Zakrzewski, V. G.; Gao, J.; Rega, N.; Zheng, G.; Liang, W.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Throssell, K.; Montgomery Jr., J. a.; Peralta, J. E.; Ogliaro, F.; Bearpark, M. J.; Heyd, J. J.; Brothers, E. N.; Kudin, K. N.; Staroverov, V. N.; Keith, T. a.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, a. P.; Burant, J. C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Millam, J. M.; Klene, M.; Adamo, C.; Cammi, R.; Ochterski, J. W.; Martin, R. L.; Morokuma, K.; Farkas, O.; Foresman, J. B.; Fox, D. J. G16_C01. 2016, p Gaussian 16, Revision C.01, Gaussian, Inc., Wallin.

(4) Comell, W. D.; Cieplak, P.; Bayly, C. I.; Kollman, P. A. Application of RESP Charges. *journal of American Chemical Society* **1993**, *115* (7), 9620–9631.