O2I.g

powerd by python

O2I.g Manual

Create Gaussian input file from Gaussian output file

---------- Developed and Edited by ----------

Zhe Wang

https://www.wangzhe95.net

Graduate School of Science

Hiroshima University, Hiroshima, Japan

E-mail: wongzit@yahoo.co.jp

[Website]

https://www.wangzhe95.net/program-o2ig

1. Overview

O2I.g is a Python program for generating *Gaussian* input file from the output file. O2I.g are open-source, free, high-efficient, and user-friendly. We provide the executable files for macOS/Linux and Microsoft Windows, and the source code.

O2I.g can be download at from author's website (https://www.wangzhe95.net/program) and author's GitHub homepage (https://github.com/wongzit).

2. Install and Usage

2.1 Run with Source Code

If Python IDE is already installed in your computer, you can run the program with the source code. Python 3.9 is recommended. You can download the newest version of Python from its homepage (https://www.python.org).

For macOS and Linux users who want to run with source code, please run following command in terminal:

```
python3 /path_to_program/program_src.py
```

For Windows users, please execute following command in PowerShell or Command Prompt (*cmd.exe*):

```
py3 / path_to_program/program_src.py
```

2.2 Run with Executable File

All executable files (if available) are pre-packaged in *execufiles* folder.

For all platforms, users can run the program with the executable files by double click. For macOS/Linux users, you may need to add permission to the executable file before running for the first time. Assume the executable file is located at "*/home/user/program/execufiles/program_linux*", run below command to add executable permission to it.

```
chmod +x /home/user/program/execufiles/program_linux
```

If the pre-packaged executable files do not work normally, please try to run with source code, or package from source code with packaging tools like *pyinstaller*.

2.3 Common Issues

If the packaged programs cannot work due to system security problem, please refer to the "Mac/Windows Users Must Read" file for solution.

3. How to Use

In this section, user-inputted commands are colored in red.

3.1 Usage of

1) Run O2I.g and drag the *Gaussian* output file to the command line window. O2I.g will read necessary information from the output files.

```
Please specify the Gaussian output file path:
(e.g.: /O2I.g/example/DR3b_CS2021.log)
/Users/wangzhe/Desktop/DR3b_CS2021.log
```

2) Choose the input file style by inputting the menu number. About different in each style, please refer to section **3.2**.

```
------- Please choose the new input file type -------
   1 - Coordinates only (default)
   2 - Read routine lines from output file
   3 - Leave routine lines section blank
   4 - Read routine lines from additional input
Please input style number: 4
```

3) The input file will be saved as *.gjf* at same dictionary as the *Gaussian* output file. The file name is ended with "_new.gjf".

3.2 Template in O2I.g

O2I.g provides 4 options of templates: (1) Coordinates only; (2) read routine lines from output file; (3) leave routine lines section blank; (4) read routine lines from additional input.

(1) Coordinates only

This option only save the Cartesian coordinates of last geometry in the output files. Users need to add the routine lines section before submit to *Gaussian* calculations.

(2) Read routine lines from output file

This option will use the same routine lines as the original output file.

(3) Leave routine lines section blank

This option will leave the routine section in blank. Users need to specify the number of CPU cores, memory, check point file path, calculation methods, charge and spin multiplicity before submit to *Gaussian* calculation.

(4) Read routine lines from additional input

To use this option, users need to prepare an additional O2I.g input file, including number of CPU cores, memory, check point file path, calculation methods, charge and spin multiplicity. An example of O2I.g input file is shown in following:

```
--------- CPU Core ---------
8

---------- Memory ----------
10GB

--------- Chkfile ----------
o2ig_test_file.chk

------ Routine Section -----
#p uwb97xd/6-31g(d) opt freq guess=(always,mix)

---------- Charge ----------
0

----------- Spin -----------
1
```