ICSScsv

powerd by python

ICSScsv Manual

Extract magnetic shielding tensor from ICSS output, version 2.0

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

Zhe Wang

http://www.wangzhe95.net

Graduate School of Science

Hiroshima University, Hiroshima, Japan

E-mail: wongzit@yahoo.co.jp

[*ICSScsv* Website]

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

1. Overview

1.1 About *ICSScsv*

ICSScsv is a program for extracting magnetic shielding tensor from ICSS calculation output. *ICSScsv* is written in Python 3, thus, users can run *ICSScsv* through Python IDE.

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

1.2 How it Works

ICSScsv reads an output file of ICSS calculation and extracted shielding tensor from it. *ICSScsv* prints the data out in *.csv* file, user can open it with data process software like *Origin*, *Prism* or even *Excel* to plot the ICSS map.

1.3 Testing Platform

ICSSgen has been tested on following platform.

1.3.1 macOS

(1) Mac mini (2020)

CPU: Intel Core i5-8500B 3.00 GHz 6 Cores 6 Threads

Mem: 16 GB 2666 MHz DDR4

GPU: Intel UHD Graphics 630 1536MB

OS: macOS 11.3.1(20E241)

(2) MacBook Air (M1, 2020)

CPU: Apple Silicon M1 8 Cores

Mem: 8 GB

GPU: Apple Silicon M1 8 Cores

OS: macOS 11.3.1(20E241)

1.3.2 Microsoft Windows

(1) Home-built PC I

CPU: Intel Core i7-9700KF 3.60 GHz 8 Cores 8 Threads

Mem: 16 GB 2666 MHz DDR4

GPU: Nvidia RTX 3060 12GB

OS: Windows 10 Education 20H2

(2) Home-built PC II (Physical machine with Windows/Linux dual-boot)

CPU: Intel Core i7-10700 2.90 GHz 8 Cores 16 Threads

Mem: 32 GB 2666 MHz DDR4

GPU: Intel UHD Graphics 630 1536MB

OS: Windows 10 Education 20H2

(3) Mac mini (2020) (Running with Parallels Desktop 16)

CPU: Intel Core i5-8500B 3.00 GHz 6 Cores 6 Threads (2 Cores used)

Mem: 16 GB 2666 MHz DDR4 (4 GB used)

GPU: Intel UHD Graphics 630 1536MB

OS: Windows 10 Education 1909

1.3.3 Linux

(1) Home-built PC I (Running with VMware Workstation Player 16)

CPU: Intel Core i7-9700KF 3.60 GHz 8 Cores 8 Threads (6 Cores used)

Mem: 16 GB 2666 MHz DDR4 (12 GB used)

GPU: Nvidia RTX 3060 12GB

OS: CentOS 8.3

(2) Home-built PC II (Physical machine with Windows/Linux dual-boot)

CPU: Intel Core i7-10700 2.90 GHz 8 Cores 16 Threads

Mem: 32 GB 2666 MHz DDR4

GPU: Intel UHD Graphics 630 1536MB

OS: Red Hat Enterprise Linux 8.3

(3) Mac mini (2020) (Running with Parallels Desktop 16)

CPU: Intel Core i5-8500B 3.00 GHz 6 Cores 6 Threads (3 Cores used)

Mem: 16 GB 2666 MHz DDR4 (4 GB used)

GPU: Intel UHD Graphics 630 1536MB

OS: Ubuntu 20.04, Fedora 34 beta

2. Run *ICSScsv*

2.2 Run with Source Code

If Python IDE is already installed in your computer, you can run *ICSScsv* with the source code. Python 3.7 or newer is recommended. *ICSScsv* may not work normally with Python 2.

For Mac users who want to run *ICSScsv* with source code, please run following command in terminal:

```
python3 /path_to_ICSScsv/ICSScsv_v*_source.py
```

2.2 Run with Executable File

All executable files are packaged in *execufiles.zip*.

2.2.1 Use Packaged Executable File

The pre-packaged executable file "*ICSScsv_v*_mac*" should be running normally on macOS 10.15 or newer with Intel and Apple M1 chip. You can run *ICSScsv* by double click the icon and *ICSScsv* will be running in terminal window.

2.2.2 Package Source Code into Executable File

If 2.2.1 is not work for some reason, you can try following steps to package *ICSScsv* by yourself:

1) Open terminal, execute `pip3 install pyinstaller` to install necessary packages.
2) Assume the source code file is located at "*/home/user/ICSScsv/ICSScsv_v*_source.py*", execute command below.

```
pyinstaller /home/user/ICSScsv/ICSScsv_v*_source.py --onefile
```

3) After that an executable file would be generated in *dist* folder. (Only executable file is needed, you can delete other files generated by *pyinstaller*.)
4) Now you can run *ICSScsv* by double clicking.

2.3 Run on Linux with Executable File

1) Assume the executable file is located at "*/home/user/ICSScsv/execufiles/ICSScsv_v*_linux*", run below command to add executable permission to it.

```
chmod +x /home/user/ICSScsv/execufiles/ICSScsv_v*_linux
```

2) (Optional) Assume the current shell is bash, add below lines to *~/.bashrc* file.

```
alias icsscsv=/home/user/ICSScsv/execufiles/ICSScsv_v*_linux
```

3) After re-entering the terminal, and you can run *ICSSgen* at any dictionary by execute "`icsscsv`" command. (If you passed the step (2), you need to execute the full path to *ICSScsv_v*_linux* for running it.)

2.4 Running on Microsoft Windows with Executable File

Find "*ICSScsv_v*_win.exe*" file in program folder, double click it and *ICSScsv* will be running in command line window.

If the Windows Defender stop the *ICSScsv*, please add the *ICSScsv* to the safe file list. More details please

check:

https://faq.nec-lavie.jp/qasearch/1007/app/servlet/relatedqa?QID=018507

3. How to Use

3.1 Before Running

You need prepare a *Gaussian* (Gaussian Inc.) output file (*.log* or *.out*) of ICSS calculation. You can use *ICSSgen* (http://www.wangzhe95.net/program-icssgen) to create an input file for ICSS calculation.

3.2 Process ICSS Output

*In this section, user inputting is colored in red.

1) Run *ICSScsv*, the *ICSScsv* will request an output file. You can drag the output file into the command window or input the full path to the output file. Then, press enter to submit.

```
Please specify the Gaussian output file path:
(e.g.: /ICSScsv/example/benzene.log)
/Users/path_to_ICSScsv/example/benzene_ICSS_XY_0.log
```

2) Choose which component will be used for ICSS map. Please input the number of the component, and press enter.

```
Choose shielding tensor for ICSS map:
     1 - Isoptropic      2 - Anisotropy
     3 - XX component    4 - YX component    5 - ZX component
     6 - XY component    7 - YY component    8 - ZY component
     9 - XZ component   10 - YZ component   11 - ZZ component
Please input the No.: 11
```

3) A *.csv* file including shielding tensor data would be generated in the same dictionary as the *Gaussian* output file, named with "*xxx_output_component.csv*".

3.3 After Running

Open the *ICSScsv* output *.csv* file with data process software like *Origin*, *Prism*, etc.