

Wailord: Parsers and Reproducibility for Quantum Chemistry

Rohit Goswami*
Science Institute, University of Iceland
Quansight Austin, TX, USA
*rog32@hi.is

Introduction

Computers are meant to provide insights, not numbers. To this end however, the ability to phrase chemical questions in a manner best suited to the efficient and reproducible workflows is of paramount importance.

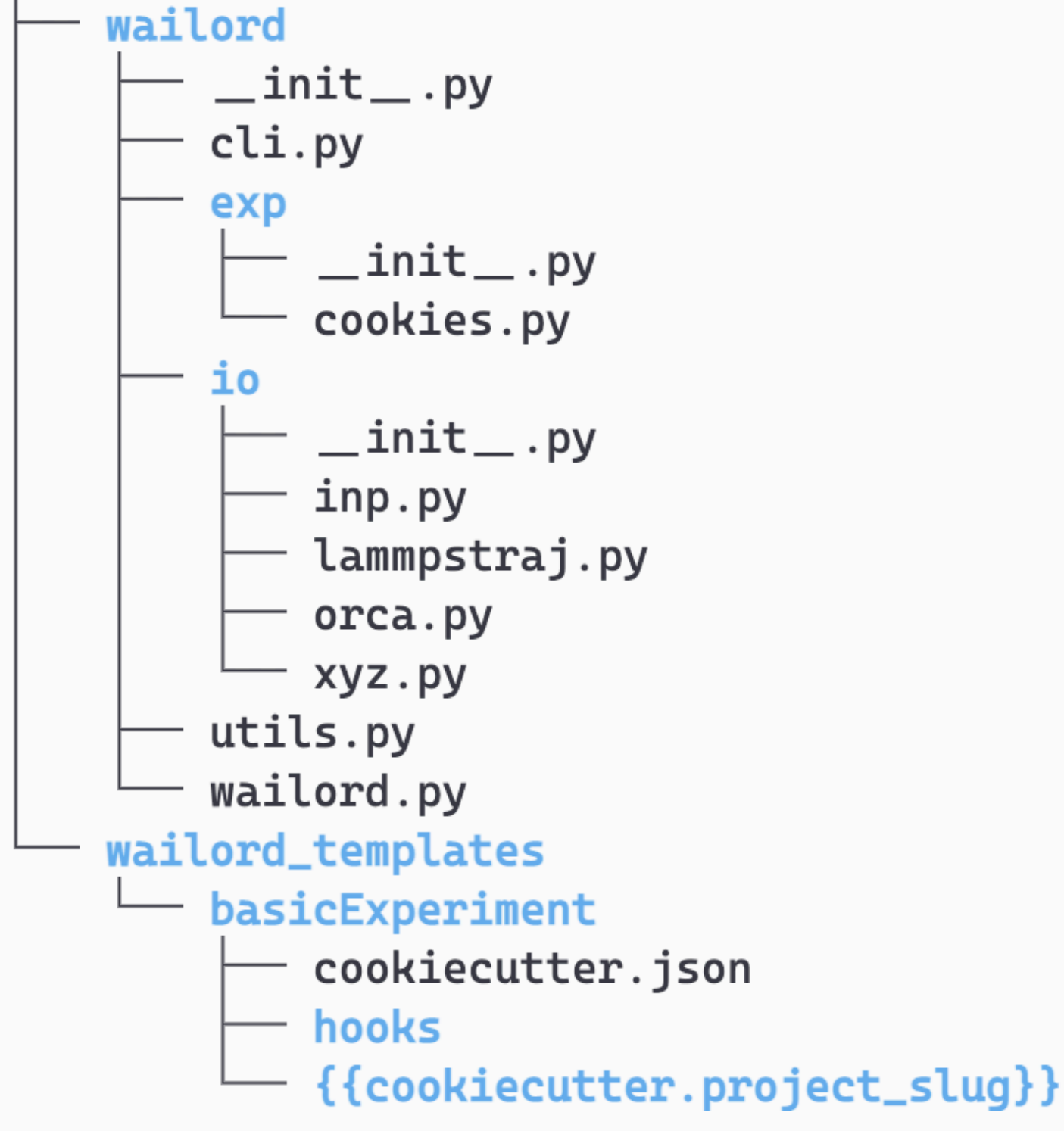
Design Principles

Data generation involves set of known configurations (say, xyz inputs) and a series of common calculations whose outputs are required. Computational chemistry packages tend to be focused on acceleration and setup details on a *per-job* scale. wailord, in contrast, considers the outputs of simulations to form a tree, where the actual run and its inputs are the leaves, and each layer of the tree structure holds information which is collated into a single dataframe which is presented to the user.

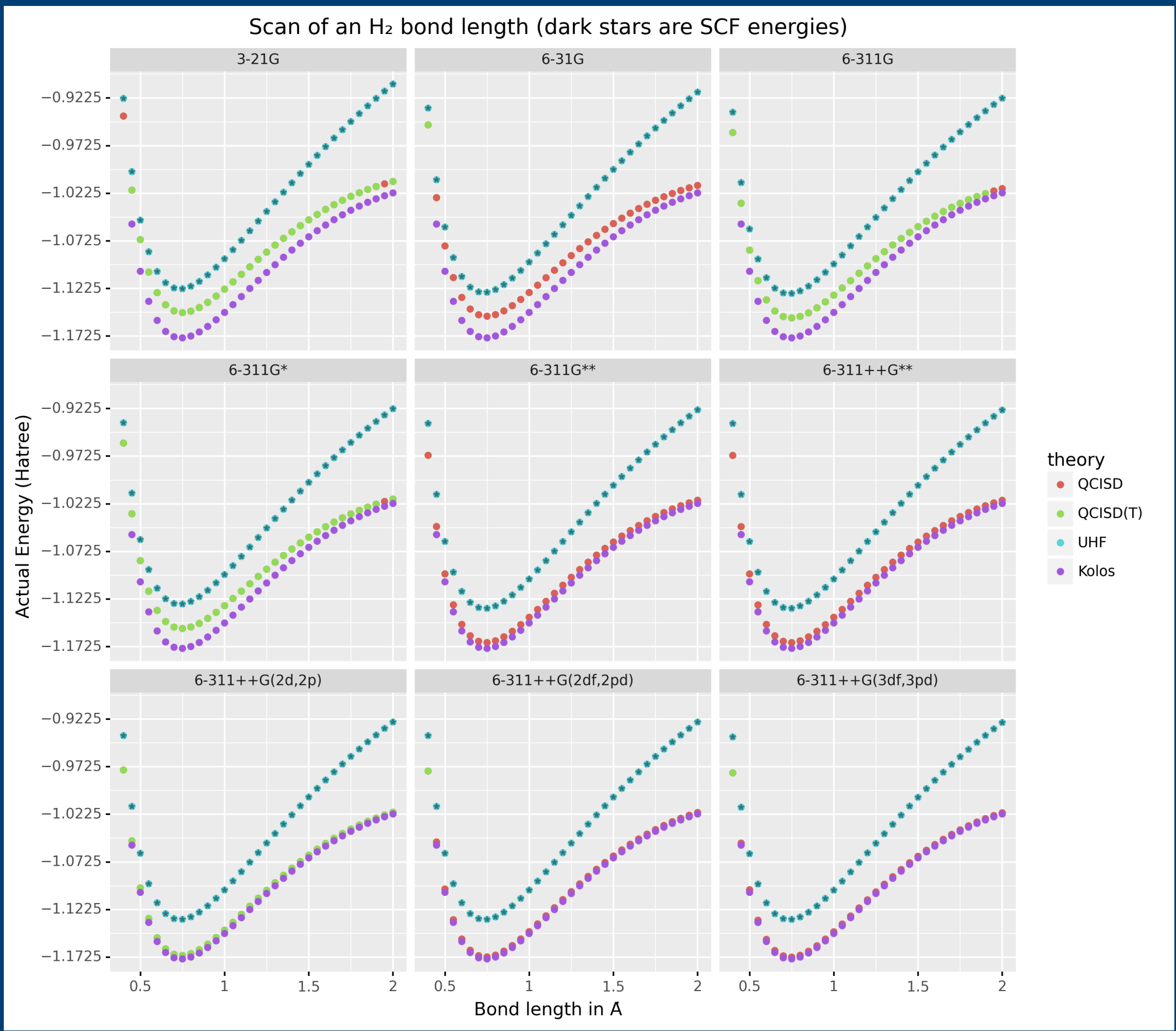
wailord leverages tools and methodologies from programming language design leading to a robust and modular tool-chain.

ORCA (version 4) forms the computational core of the package.

Each segment is unit-tested to ensure reproducible results, and the design is modular enough to be expanded into more general purpose usage as well.



The appropriate level of abstraction for computational chemistry workflows is that of a domain specific programming language (DSL).



† Data → parsimonious

† Outputs → pandas dataframes

† Units → pint

† Experiments defined as cookiecutter templates with metadata

† Reliable output structured without user intervention



Take a picture to download the full paper

Partially supported by the Icelandic Research Fund, grant number 217436052.

Parsing Structures

A grammar can be expressed as a series of tokens (terminal symbols) and non-terminal (syntactic variables) symbols along with rules defining valid combinations of these.

```
2
H -2.8 2.8 0.1
H -3.2 3.4 0.2

grammar_xyz = Grammar(
    r"""
    meta = natoms ws coord_block ws?
    natoms = number
    coord_block = (aline ws)+
    aline = (atype ws cline)
    atype = ~"[a-zA-Z]" / ~"[0-9]"
    cline = (float ws float ws float)
    float = pm number "-" number
    pm = ~"[+-]"?
    number = ~"\\d+"
    ws = ~"\\s*"
    """
)
```

Generating Inputs

Each "experiment" consists of multiple single-shot runs; each of which can take a long time. A top level experiment is defined as:

```
project_slug: methylene
project_name: singlet_triplet_methylene
outdir: "./lab8"
desc: An experiment to calculate singlet and triplet
states differences at a QCISD(T) level
author: Rohit
year: "2020"
license: MIT
orca_root: "/home/orca/"
orca_yaml: "orcaST_meth.yml"
inp_xyz: "ch2_631ppg88_trip.xyz"
```

Where each run is controlled individually.

```
qc:
  active: True
  style: ["UHF", "QCISD", "QCISD(T)]
  calculations: ["OPT"]
  basis_sets:
    - 6-311++G**
  xyz: "inp.xyz"
  spin:
    - "0 1" # Singlet
    - "0 3" # Triplet
  extra: " !NUMGRAD"
  viz:
    molden: True
    chemcraft: True
  jobscript: "basejob.sh"
```

With a directory tree generated by:

```
waex.cookies.gen_base(
    template="basicExperiment",
    absolute=False,
    filen="./lab8/expCookieST_meth.yml",
)
```

Finally pandas data frames can be extracted from the outputs and analysis may be carried out say, in a jupyter notebook.

```
mdat = waio.orca.genEBASet(Path("buildOuts") /
\ "methylene",
deci=4)
print(mdat.to_latex(index=False,
caption="CH2 energies and angles \
at various levels of theory, with NUMGRAD"))
```

References

[1] A. V. Aho and A. V. Aho, Eds., *Compilers: Principles, Techniques, & Tools*, 2nd ed. Boston: Pearson/Addison Wesley, 2007, 1009 pp.

[2] G. K. Sandve, A. Nekutenko, J. Taylor, and E. Hovig, "Ten Simple Rules for Reproducible Computational Research," *PLOS Computational Biology*, vol. 9, no. 10, e1003285, Oct. 24, 2013.

[3] R. D. Peng, "Reproducible Research in Computational Science," *Science*, vol. 334, no. 6060, pp. 1226–1227, Dec. 2, 2011. PMID: 22144613.

[4] N. M. O'boyle, A. L. Tenderholt, and K. M. Langner, "Cclib: A library for package-independent computational chemistry algorithms," *Journal of Computational Chemistry*, vol. 29, no. 5, pp. 839–845, 2008.

[5] F. Neese, F. Wennmohs, U. Becker, and C. Riplinger, "The ORCA quantum chemistry program package," *The Journal of Chemical Physics*, vol. 152, no. 22, p. 224108, Jun. 12, 2020.

[6] W. Kolos and L. Wolniewicz, "Improved Theoretical GroundState Energy of the Hydrogen Molecule," *The Journal of Chemical Physics*, vol. 49, no. 1, pp. 404–410, Jul. 1, 1968.