
rexfw Documentation

Release 1.0

Simeon Carstens

Jun 03, 2019

Contents:

1	First steps with rexfw	1
1.1	What is rexfw about?	1
1.2	Setting up rexfw	1
1.3	Contact	2
1.4	License	2
2	Indices and tables	3

1.1 What is rexfw about?

reffw is a simple (for me, at least...) Python 2.7 framework to enhance sampling in MCMC simulations using the Replica Exchange algorithm (RE; Swendsen & Wang, Phys. Rev. Lett. 1986). All you need to do is plug in your own probability distributions and MCMC samplers, which have to expose a simple interface to be compatible with this package. It can be easily extended to RE-related methods such as Replica Exchange with Nonequilibrium Switches (RENS; Ballard & Jarzynski, PNAS 2009).

This is a rather naive, synchronous, and probably not super efficient implementation of RE which allows only one replica per process. It relies on MPI for communication between processes and can thus easily be run both on a cluster and on single machines.

I use it to enhance sampling from complex posterior distributions arising in my research on Bayesian inference of chromosome structure (Carstens et al., PLOS Comput. Biol. 2016). I also implemented various RENS flavors to assess their performance in the above-mentioned sampling problems. These RENS implementations are also contained in the package, but require the Computational Structural Biology toolbox (CSB, [download](#)) and are not unit-tested (many components of it have unit tests in CSB, though).

1.2 Setting up rexfw

Given only three dependencies (a MPI implementation, numpy and mpi4py), installing rexfw is as easy as cloning this repository and then running the setup script:

```
$ python setup.py install
```

possibly with `--user`, if you don't have administrator privileges and instead need to install rexfw in your home directory.

To get started using rexfw, I recommend you to take a look at the file `reffw/tests/normal.py` file which is a commented script using rexfw to sample from a normal distribution. Check the imports in this script for further comments, especially the probability distribution and sampler interfaces. The example can be run with:

```
$ cd rexfw/test
$ mpirun -n 6 python normal.py
```

This will use six cores (one master process, five replicas) and write all simulation output to `/tmp/normaltest_5replicas/`.

Furthermore, there is a complete API documentation to be found in the `doc` directory and online at rexfw.readthedocs.io.

The core of the package is unit-tested; you can run the tests with:

```
$ cd rexfw/test
$ python run_tests.py
```

Not unit-tested are the `statistics` and the `convenience` submodules, which include functionality to print sampling / RE statistics to stdout / files and convenient functions for setting up a standard RE environment.

1.3 Contact

Problems? Questions? Don't hesitate to drop me (Simeon) a message - I'm happy to help!

1.4 License

rexfw is open source and distributed under the OSI-approved MIT license:

```
Copyright (c) 2018 Simeon Carstens
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
```

```
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`