
PYPOWER Documentation

Release 4.0.0

Richard Lincoln

July 07, 2011

CONTENTS

1	Introduction	3
2	License and Copyright	5
3	Installation	7
4	Usage	9
5	Indices and tables	11

Contents:

INTRODUCTION

PYPOWER is a power flow and Optimal Power Flow (OPF) solver. It is a port of [MATPOWER](#) by PSERC (Cornell) to the [Python](#) programming language. Current features include:

- DC and AC (Newton's method & Fast Decoupled) power flow and
- DC and AC optimal power flow (OPF)

LICENSE AND COPYRIGHT

Copyright (C) 2009-2011 Power System Engineering Research Center

Copyright (C) 2010-2011 Richard Lincoln

PYPOWER is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

PYPOWER is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with PYPOWER. If not, see <http://www.gnu.org/licenses/>.

INSTALLATION

PYPOWER depends upon:

- [Python](#) 2.5 or later and
- [SciPy](#) 0.7 or later.

It can be easily installed using [setuptools](#):

```
$ easy_install PYPOWER
```

Alternatively, [download](#) and unpack the tarball and install:

```
$ tar xzf PYPOWER-4.0.tar.gz  
$ python setup.py install
```

On UNIX systems, use `sudo` for the latter command if you need to install the scripts to a directory that requires root privileges:

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

The development [Git](#) repository can be cloned from [GitHub](#):

```
$ git clone http://github.com/rwl/PYPOWER.git
```


USAGE

Installing PYPOWER creates `pf` and `opf` commands. To list the command options:

```
$ pf -h
```

PYPOWER includes a selection of test cases. For example, to run a power flow on the IEEE 14 bus test case:

```
$ pf -c case14
```

Alternatively, the path to a PYPOWER case data file can be specified.:

```
$ pf /path/to/case14.py
```

The `opf` command has the same calling syntax. For example, to solve an OPF for the IEEE Reliability Test System and write the solved case to file:

```
$ opf -c case24_ieee_rts --solvedcase=rtsout.py
```

For further information please refer to the [API documentation](#).

INDICES AND TABLES

- *Index*
- *Module Index*
- *Search Page*