Documentation

This document provides information about the software that was used to investigate non-Poisson continuous time random walks in our paper [6]. We discuss the requirements necessary to run the software in section I and provide an outline of the components of the software in section II.

I. REQUIREMENTS & SETUP

The software is based on a number of libraries that need to be installed before the software can be run. In particular, you will need to have the following software installed on your machine

- Python 2.6 or later (but not the 3.* version of python which is not backwards compatible) [4],
- Numpy and Scipy [5],
- NetworkX [3],
- (MatPlotLib [2] is required to run the example).

The easiest way to install all of the above is to use the free Enthought Python Distribution [1] which is available for all major operating systems. After having installed the Enthought package you can install NetworkX by issuing the command

```
sudo easy_install networkx
```

We have implemented parts of the software in C because evaluating the probability distribution functions associated with the waiting times can be computationally intensive. You need to compile the C source before being able to run the software by executing the following steps

1. Issue the command
   `python _distributionssetup.py`
2. Navigate to the build directory and copy the compiled file into the source directory such that it is visible to the python implementation. For example, on a Mac OSX system the compilation process creates a file `_distributions.so` in the folder `<source directory>/build/lib.macosx-10.5-i386-2.7/`. The file `_distributions.so` needs to be copied into the source directory.

You should be ready to go!

II. COMPONENTS

Each file contains commentary to explain its functionality. Thus, we will only give a brief summary of each file's content here.

`_distributions.c` contains C implementations to evaluate probability distribution functions and cumulative distribution functions quickly.

`_distributionssetup.py` is a helper script that you can use to compile the C source. It will create a library file which you need to copy to the source directory.

`distributions.py` is a python wrapper for the C library and defines probability distribution functions as classes.

`walker.py` contains the main functionality of the software which can be categorized into two groups

- Monte-Carlo simulations that approximate the walker density on networks by generating a large number of random walks,
- Calculation of effective transition matrices and resting times to obtain steady state solutions.

`example.py` is a self-explanatory example. It creates a toy network of three nodes, approximates the walker density through simulations and obtains the steady-state solution explicitly.

[1] Enthought python distribution. URL `http://www.enthought.com/products/epd_free.php`.

[2] matplotlib. URL `http://matplotlib.org/`.

[3] Networkx. URL `http://networkx.lanl.gov/`.

[4] Python programming language. URL `http://python.org/`.

[5] Scipy. URL `http://scipy.org/`.

[6] Till Hoffmann, Mason A. Porter, and Renaud Lambiotte. Generalized master equations for non-poisson dynamics on networks. *Phys. Rev. E*, 86:046102, October 2012. doi: 10.1103/PhysRevE.86.046102. URL `http://link.aps.org/doi/10.1103/PhysRevE.86.046102`.