Logic Modeling

Requirements

- numpy
- cython
- pypar

Can download files from github at https://github.com/LoLab-VU/LogicalModel

Downloading and testing code

```
git clone https://github.com/LoLab-VU/LogicalModel.git

python cb_setup.py build_ext

python main_attractor_synch_cython.py -m Models/core_iron_6variables_3states.txt
```

https://github.com/LoLab-VU/LogicalModel&nbsp
Old models directory contains 5 models, new models are in NewModels_2015_8_17.

```


Models/core_iron_6variables_3states.txt
Models/final_continuous_model_21_nodes.txt
Models/func-example.txt
Models/IRP2_overexp_functions_1.txt
Models/IRP2_overexp_functions_2.txt
Models/core_iron_6variables_3states.txt




NewModels_2015_8_17/Ftmt_oe_24.txt
NewModels_2015_8_17/IRP2_oe_24.txt
NewModels_2015_8_17/Mfrn_ko_24.txt
NewModels_2015_8_17/normal_24.txt
NewModels_2015_8_17/normal_25.txt
NewModels_2015_8_17/Q_ko_25.txt

Ftmt_oe_24.txtQ_ko
```

Example Usage

For this example we will use the core_iron_6variables_3states.txt model. The main program is called main_attractor_synch_cython.py. The option are

```
usage: main_attractor_synch_cython.py [-h] [-n NSTATES] [-s START] [-e END]usage:
main_attractor_synch_cython.py [-h] [-n NSTATES] [-s START] [-e END]
                                      [-m MODEL] [-v VERBOSE] [-p PARALLEL]

optional arguments:
  -h, --help            show this help message and exit
  -n NSTATES, --nstates NSTATES
                        provide a number of states
  -s START, --start START
                        starting string to convert to base Nstates
  -e END, --end END     ending string to convert to based Nstates
  -m MODEL, --model MODEL
                        model to run simulation, assumes file to end in .txt
  -v VERBOSE, --verbose VERBOSE (optional)
                        if you want verbose updates (use with single
                        processor)
  -p PARALLEL, --parallel PARALLEL (optional) will use multiprocessing automatically.
                        run in parallel, use 0 or 1
```

The only required arguments are the number of states (-n) and the model file (-m)

```
python main_attractor_synch_cython.py -n 3 -m Model/core_iron_6variables_3states.txt
```

```
pinojc@LoLab-760:~/Projects/LogicalModel$ python main_attractor_synch_cython.py -n 3
 -m Models/core_iron_6variables_3states.txt
Running on single CPU
Started
Computed 729 samples 0.0005 minutes
Attractors  [(1, 0, 2, 2, 0, 0), (1, 2, 0, 0, 1, 1), (1, 1, 1, 1, 1, 1)]
Frequencies  [266, 434, 29]
Total  729
```

Running on multiple processors/across nodes

NOTE*** With above we can use multiprocessing on a single node (running on all processors). Below is to use multiple nodes. Probably better to run multiple files and starting and stopping each rather than trying to get tons of nodes all at the same time.

Then you run using mpirun

```
mpirun -n #proc /path/to/python main_attractor_synch_cython.py -n 3 -p 1
```

```
pinojc@LoLab-760:~/Projects/LogicalModel$ mpirun -np 4 python
main_attractor_synch_cython.py -n 3  -m Models/core_iron_6variables_3states.txt -p 1
Processor 3 initialised on node LoLab-760
main_attractor_synch_cython.py:54: UserWarning: Shared object file
'core_iron_6variables_3states.so' already exists: moving on.
  warnings.warn(warning_string + 'moving on.')
Processor 1 initialised on node LoLab-760
Processor 0 initialised on node LoLab-760
samplesize =  728
split up into 72 segments
Processor 2 initialised on node LoLab-760
Computed 729 samples in 0.0012 minutes
Attractors  ['002211', '111111', '011111']
Frequencies  [266, 29, 434]
Total  729
```