Dynamic Benchmark Network Generator - Non-Overlapping Version

Description

This software package includes code to generate sequences of dynamic graphs with embedded communities. The graphs are undirected and unweighted, and the communities are non-overlapping.

This software is largely based on binary network generation tools written by Andrea Lancichinetti and Santo Fortunato. We are very grateful to those authors for making the original tools available:

http://sites.google.com/site/santofortunato/inthepress2

Although this version of the package is not actively supported, additional details may be obtained by contacting derek.greene@ucd.ie

If you find this tool useful, please consider citing the paper:

- D.Greene, D.Doyle, and P.Cunningham, "Tracking the evolution of communities in dynamic social networks," in Proc. International Conference on Advances in Social Networks Analysis and Mining (ASONAM'10), 2010. [PDF] [Supplementary material]

Parameters

Each of the generators has a number of core parameters controlling the network. These are the ones you will need:

```
-seed     [random number generator seed]
-N        [number of nodes]
-s        [number of time steps to generate]
-k        [average degree]
-maxk     [maximum degree]
-muw      [mixing parameter - controls the overlap between communities]
-minc     [minimum for the community sizes]
-maxc     [maximum for the community sizes]
```

In addition each generator produces dynamic graphs containing specific types of community evolution events, and has one or more custom parameters relevant to each event type:

bench_switch: flips memberships between communities at each step

```
-p        [probability of a node switching community membership between time steps]
```

bench_birthdeath: permanently adds/removes communities at each step

```
-birth    [number of community birth events per time step]
-death    [number of community death events per time step]
```

*bench_expand: expands/contracts communities at each step

```
-expand   [number of expansion events per time step]
-contract [number of contraction events per time step]
-r        [rate of expansion/contraction]
```

*bench_hide: temporarily hides a community for a single step

```
-hide     [fraction of communities to hide per time step]
```

*bench_mergesplit: merges/splits communities at each step

```
-merge    [number of merge events per time step]
-split    [number of split events per time step]
```

Example Usage

To generate 5 time steps of 250 nodes with ~20 communities, a low level of inter-community connectivity, no overlapping communities, and 10% membership switching at each step:

```
./bench_switch -s 5 -N 250 -k 10 -maxk 20 -muw 0.2 -p 0.1
```

The *.edges files give the edge lists for the graphs at each step, and the *.comm files given the correct ground-truth communities corresponding to those graphs.