# PRU

Description

The PRU model for unstructured Peer-to-Peer systems, proposed by Pandurangan et. al, is based on a simple network growth policy that ensures low graph diameter. In these graphs, nodes have a boolean attribute inCache, indicating their role in network evolution. The model has as parameters node degree K, minimum degree L, and maximum degree U. The graph starts with K nodes with attribute inCache = True. Each of these nodes has L edges incident on them from randomly chosen nodes within the group. When a new node N is introduced into the graph its inCache value is False, and edges are added between it and L randomly selected inCache nodes. If this addition causes any inCache node N_C to have more than U edges, N_C has its inCache value set to False, and a non-inCache node in the system is chosen to become inCache.

A typical network layout is given below. Nodes with attribute inCache = True are shown in black.
Links

- Source Code: ...
- Binaries: ...
- Update Site: ...
- Home Page: ...
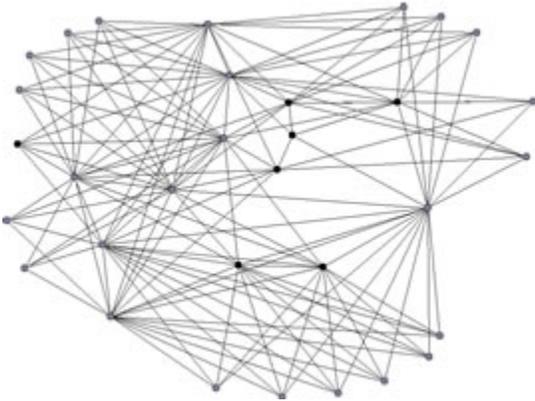
Pros & Cons

Applicable to network graphs only. PRU model imposes minimal network growth policies. It focuses on growing a network with the desirable low diameter of small world systems using only local information. It strives for complete decentralization of network growth and is topologically more robust. But the search cost on PRU network models is high.
Applications

These algorithms can be used to generate unstructured P2P graphs based on PRU topology. These graphs can then be visualized and analyzed using Pajek or any other network visualization program.

Implementation Details

Pseudo code:

```
Add nodes to the graph. for i(0..lowerBound) {

 Add node[i] to cache.
 Connect node[i] with other nodes in the cache.

} for i(lowerBound..networkSize) {

if (i < cacheSize){
  Connect node[i] to (random[i] * lowerBound) nodes.
  Add node[i] to cache.
 }
else{
  Connect node[i] to a random node in the cache.
 }

}
```

Usage Hints

In the IVC Software Framework, simply select Modeling > P2P > Unstructured > PRU to generate a network. Recommended InCache Nodes = M/4, Minimum Degree = log M, Maximum Degree = 3log M + 3. Use any search algorithms via the Analysis > Search menu to analyze the network.

References

1. Pandurangan, G., Prabhakar Raghavan, and Eli Upfal.Building Low-Diameter Peer-to-Peer Networks. IEEE J. Select. Areas Commun., 21(6):995-1002, Aug. 2003.