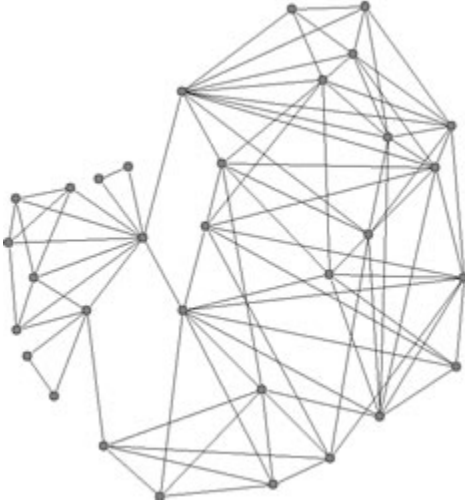


# Hypergrid

## Description

The Hypergrid model for Peer-to-Peer (P2P) systems builds a graph topology that enforces low graph diameter and fixed node degree. The Hypergrid graph grows as a simple  $k$ -ary tree with nodes on the leaf level of the tree having their  $k-1$  free edges randomly connected to other nodes on the same level in the tree with degree less than  $k$ .

Initially graph consists of a single node with no edges. For a new node  $N$  to join the graph, it looks for a parent node in the graph which has free edges. A horizontal edge between the parent node and another node, on the same level as that of parent node, is removed. The new node then connects to this parent node and also to  $k-1$  other nodes on the same level in the graph. A typical hypergrid network layout is given above.



## Pros & Cons

Applicable to network graphs only. Hypergrid 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 Hypergrid network models is very high.

## Applications

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

## Implementation Details

Pseudo code:

1. Identify the innermost member maintaining "horizontal" connections. 2. Request one of these links to be terminated and reallocated (becoming "vertical" in the process)
3. Attempt to initiate  $k-1$  horizontal links with other nodes belonging to the same layer and advertising a spare connection.

## Usage Hints

In the IVC Software Framework, simply select Modeling > P2P > Unstructured > Hypergrid to generate a network. Recommended degree =  $2\log M + c$ , where  $M$  is the network size and  $c$  is a constant ( $c < 6$ ). Use any search algorithms via the Analysis > Search menu to analyze the network.

## Acknowledgments

This package was developed by George Fletcher and Hardik Sheth. Hardik Sheth integrated the code into the IVC Software Framework

## References

- Saffre, Fabrice and Robert Ghanea-Hercock. Beyond Anarchy: Self-Organized Topology for Peer-to-Peer Networks. Complexity, 9(2):49:53, 2003.

## Links

[Source Code](#)