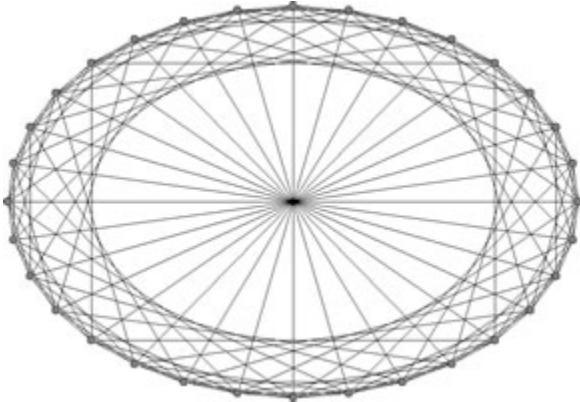


# Chord

## Description

Chord is another self-organizing structured P2P system model. This algorithm provides an implementation of the Chord primitive for the modeling of peer-to-peer networks and to perform search on such topologies. Chord provides a flexible, high-performance lookup primitive upon which functionality for efficient data location, anonymity, search, server selection, authentication etc. can be layered.

Chord uses consistent hashing to map nodes onto a circular identifier space (see Figure below). Nodes in a Chord graph have, as an additional attribute, a coordinate in a 1-dimensional virtual space (called a ring). When a new node N is added to the graph, the attributes of the nodes adjacent to N on the ring are used to add edges between N and k other nodes distributed in the space. Every node also keeps track of its predecessor and successor nodes. This provides significant advantage over standard consistent hashing which requires every node to track almost all other nodes.



## Pros & Cons

Applicable to network graphs only. Since placement of system resources at nodes is strictly controlled in a centralized fashion, network evolution has costly overhead. Centralized control limits network robustness and node autonomy.

Chord model is advantageous for building systems where controlled resource placement is a high priority, such as distributed file storage. Chord model provides sub-linear search mechanisms.

## Applications

The algorithm can be used to generate Chord structured P2P graphs. These graphs can then be visualized and analyzed using Pajek or any other network visualization program.

### Implementation Details

Pseudo code:

```
n.find_successor(id) if (id < (n, successor))
return successor
else // forward the query a round the circle
return successor.find_successor(id)
```

## Usage Hints

In the IVC Software Framework, simply select Modeling > P2P > Structured > CHORD to generate a network. Recommended number of edges =  $\log M$ , where M is the size of the network. Use Analysis > Search > CHORD Search 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

1. I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications. IEEE & ACM Trans. on Networking, 11(1): 17-32, February 2003.

## Links

[Source Code](#)