Scalable Egalitarian Networks and the Nested Clique

Edward L. Platt and Daniel M. Romero – University of Michigan School of Information

Egalitarian Networks

In an ***egalitarian*** network, there are no special positions in the network structure. The most egalitarian networks are ***vertex-transitive***: all nodes have the exact same structural properties. Egalitarian networks can be highly resilient, accomodate large flows, and allow innovations to diffuse widely. These properties can be desirable in a range of contexts, including: social networks, peer-to-peer networking, security, and parallel processing. We present the ***nested clique***, a novel egalitarian network topology.

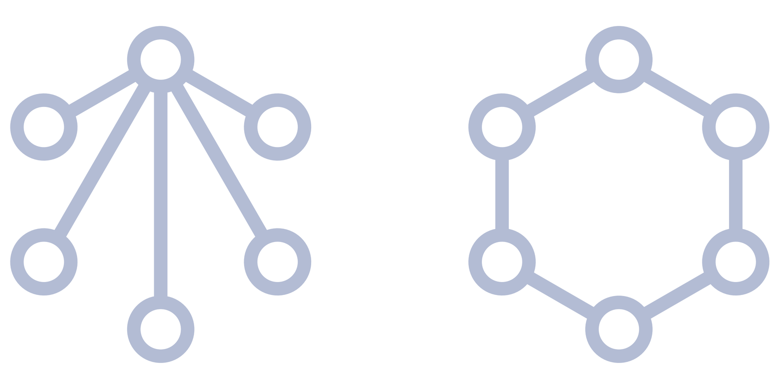

Figure 1. Examples of a non-egalitarian network with a single high-centrality node (left) and an egalitarian network (right).

Does It Scale?

In practice, networked systems are often limited in how large they can grow. Nodes may be limited in the number of edges they can have. In social networks for exmple, Dunbar's number [1] limits the number of relationships any individual can have. Network size can also be limited by the requirement that short paths exist between all nodes. Networks that can grow large despite these limitations are said to be ***scalable***.

Node Degree

In scalable networks, the degree of each node increases slowly (or not at all) as new nodes are added to the network.

Path Length

In scalable networks, the average and maximum path length grows slowly as new nodes are added to the network.

Nested Clique Topology

We present a novel network topology, which we call the ***nested clique***. The nested clique is vertex-transitive, and therefore egalitarain. It is also sparse and low-diameter, making it scalable. The structure of the nested clique also lends itself to a natural routing algorithm, which allows short paths to be found between any pair of nodes. The nested clique can be compared to other sparse, low-diameter, vertex-transitive topologies, such as the hypercube and butterfly.

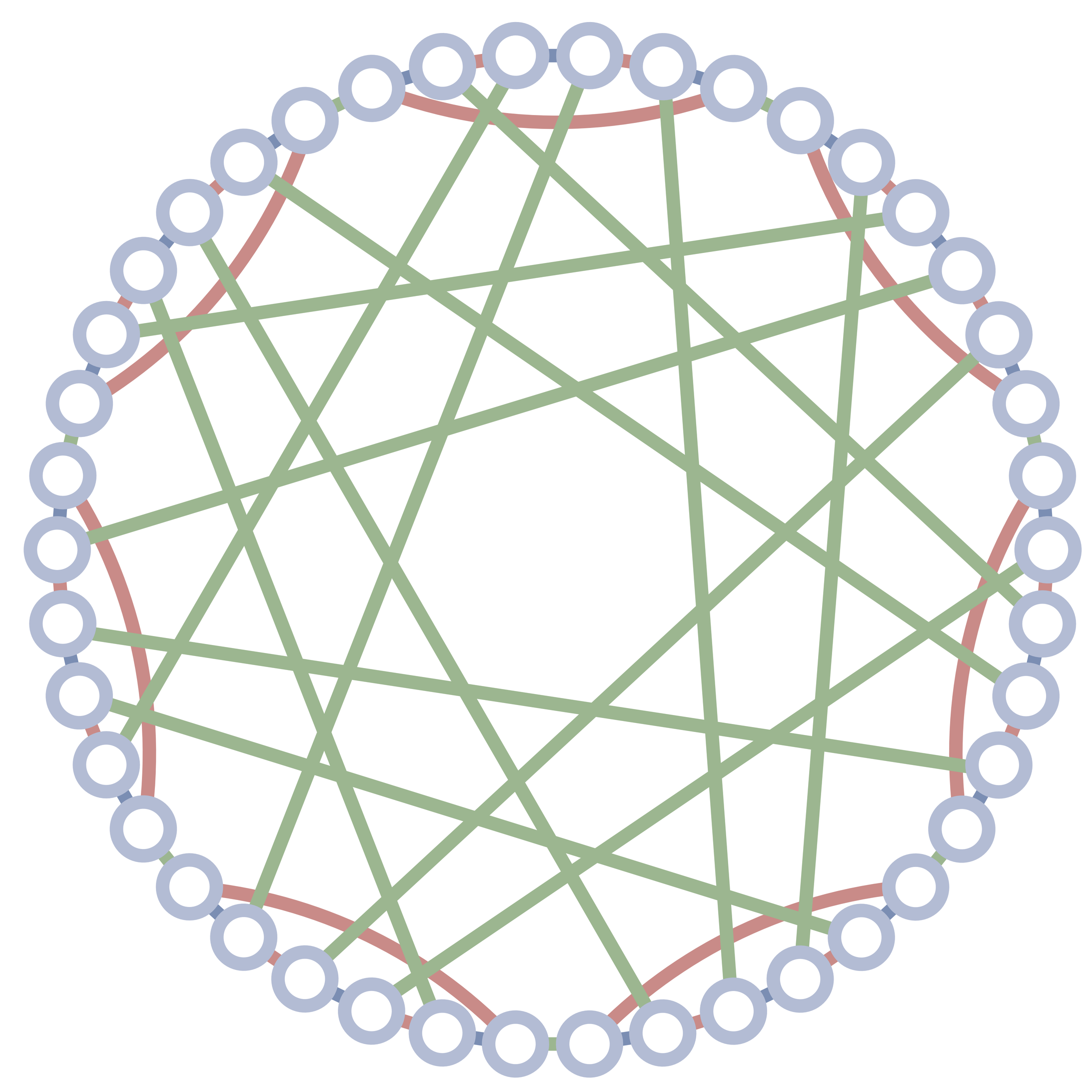

Figure 2. A degree-3 nested clique, composed of seven degree-2 nested cliques. Each degree-2 community has exactly one link (green) to each of the other degree-2 communities.

Recursive Construction

The nested clique is constructed recursively. The (n+1)-degree nested clique is constructed as follows: 1. Begin with an n-degree nested clique, 2. For each node in the orginal, create another n-degree nested clique, 3. For each of these nested cliques, add one link to each of the others, maintaining vertex-transitivity.

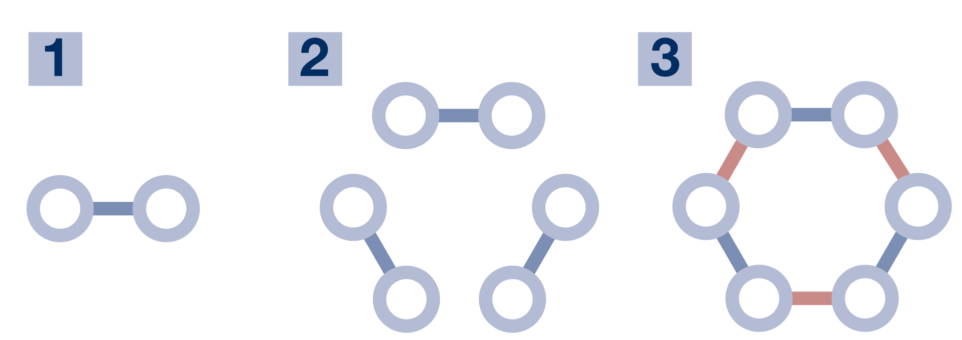

Figure 3. The three stages of constructing a degree-2 nested clique from a degree-1 nested clique. Two copies of the original are created, and then exactly one link (red) is added between each pair of degree-1 nested cliques.

Comparison

We compare the network properties of the nested clique to two other sparse, low-diameter, vertex-transitive networks: the hypercube [2] and the butterfly [3]. These topologies are commonly used in parallel processing and peer-to-peer networking [4].

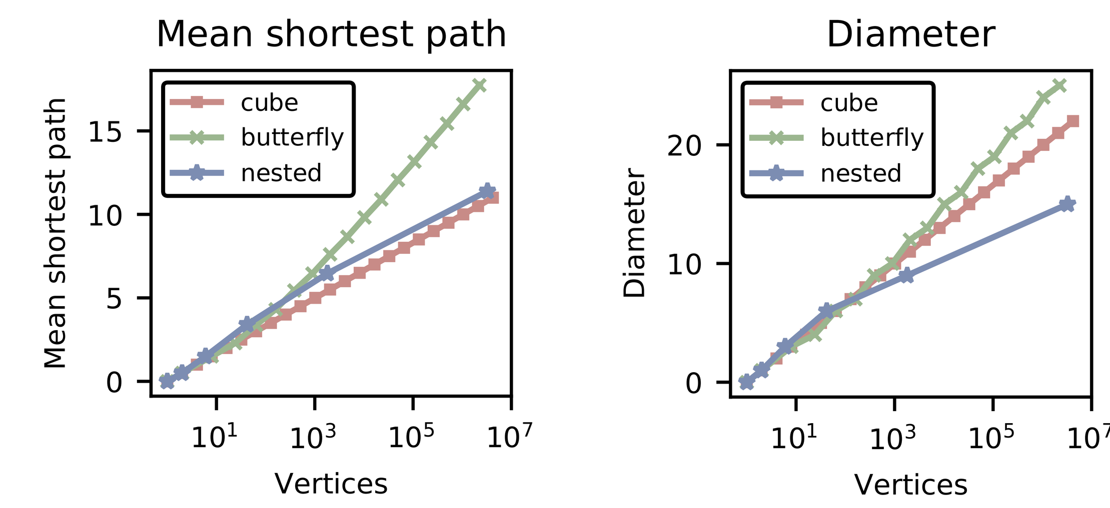

Figure 5. The nested clique topology has both shorter mean path length and smaller diameter than the butterfly. It has smaller diameter than the hypercube, with comparable mean shortest path.

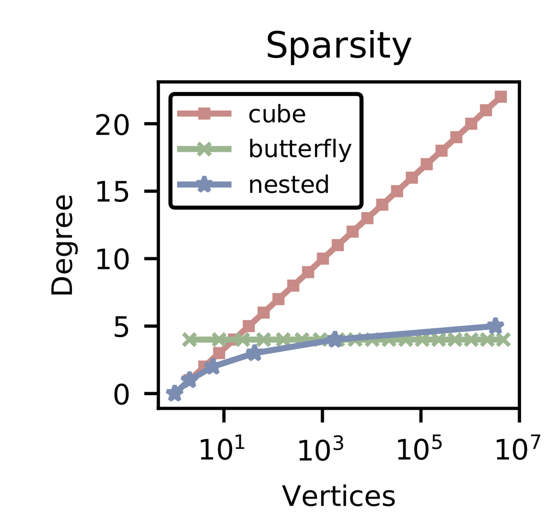

Figure 6. In a nested clique with N nodes, all nodes have degree O(log log N), making it much sparser than the O(log N) cube, and slightly less sparse than the O(1) butterfly topology.

Routing Algorithm

The recursive structure of the nested clique lends itself to a recursive routing algorithm which can find an O(log N) length path between any two nodes. Assuming a routing algorithm on the n-degree nested clique, the routing algorithm for the (n+1)-degree nested clique is constructed as follows:

1. Within the source's n-degree component, construct a path to the unique node having a connection to the target's n-degree component.
2. Add an edge from the source's n-degree component to the target's n-degree component.
3. Within the target's n-degree component, construct a path to the target node.

Applications

Scalable egalitarian networks are desirable for a wide range of applications. The uniform distribution of node degrees can improve the performance of groups in social learning tasks [5] and limit critical points of failure in communication networks [6]. The regular structure and efficient routing algorithm of the nested clique make it well-suited for connecting peers in peer-to-peer networks [4] and for connecting CPU cores in parallel processing [2,3].

Wisdom of Crowds

Egalitarian networks can imporove group performance in social learning tasks by reducing the influence of noisy information [5].

Security

With no critical points of failure, egalitarian networks are more resistant to both random failures and targeted attacks [6].

Peer-to-Peer Networking

Egalitarian networks with efficient routing are well-suited for peer-to-peer networks, which do not rely on central servers [4].

Parallel Processing

The symmetry and efficient routing of the nested clique make it a viable option for parallel processing architectures [2,3].

Conclusion

Scalable egalitarian networks are desirable for many applications. Compared to existing topologies, such as the hypercube and butterfly, the nested clique has similar or preferable path lengths and sparsity. For networks too big to support the hypercube's O(log N) node degree but small enough to support a node degree of O(log log N), the nested clique offers shorter path lengths than the butterfly topology. The existence of a routing algorithm guarantees that efficient paths can be found between any pair of nodes, enabling the nested clique to be used in practical applications.

Acknowledgements

The authors would like to thank the University of Michigan School of Information for supporting this work.

References

[1] Dunbar, R. I. (1993). Coevolution of neocortical size, group size and language inhumans. *Behavioral and brain sciences*, 16(04), 681-694.
[2] Siegel, H. J. (1979). Interconnection networks for SIMD machines. *Computer*, 12(6), 57-65.
[3] Kshemkalyani, A. D., & Singhal, M. (2011). *Distributed computing: principles, algorithms, and systems*. Cambridge University Press.
[4] Lua, E. K., Crowcroft, J., Pias, M., Sharma, R., & Lim, S. (2005). A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys & Tutorials*, 7(2), 72-93.
[5] Golub, B., & Jackson, M. O. (2010). Naive learning in social networks and the wisdom of crowds. *American Economic Journal: Microeconomics*, 2(1), 112-149.
[6] Albert, R., Jeong, H., & Barabási, A. L. (2000). Error and attack tolerance of complex networks. Nature, 406(6794), 378-382.