

I recently read this paper [Low-Resource Eclipse Attacks on Ethereum's Peer-to-Peer Network](#) which explains how Ethereum nodes discover and connect to other nodes. My understanding is that although Ethereum uses a Kademlia like method to discover other nodes, in the end a node more or less still chooses uniform random nodes to establish TCP connections.

My research interest is mostly in [random graphs](#). So I am wondering if the following model could reasonably capture the topology of Ethereum P2P networks

- A node in the network can have at most 25 connections (capacity)
- At the beginning there is only 1 node in the network
- New nodes arrive one by one in the network
- When a new node joins the network, it chooses 25 nodes whose capacity has not been reached to connect

Of course this is a extremely simplified model, but we can ask how much it actually resembles the true network topology. And if the difference is not big, maybe Ethereum network protocols can be simplified. If the difference is significant, we can ask what new parameters or rules can we add to the model?