Peer-to-Peer (P2P) Network

- A distributed network of computers; no distinction between a server and a client.
- A dynamic network: nodes (peers) and edges (currently established connections) appear and disappear over time.
- Nodes communicate using only local information.
- Advantages: decentralized computing (e.g., search), sharing data and resources.
- Real-life Systems: Gnutella, Freenet.

Gnutella

Joining:

Nodes contact a (central) host server to get entry-points to the network.

Search:

- A node sends query to its neighbors.
- They in turn forward it to their neighbors: decrement Time to Live (TTL) for query; query dies when TTL = 0.
- Search answers sent back along requested path.

Building P2P Networks

- Connectivity and low-diameter are two critical properties.
- Maintaining (even) connectivity under a dynamic setting is a non-trivial issue.
- Current real-life systems (e.g., Gnutella):
  - take an ad hoc approach
  - results in partitioning of the network into disconnected pieces
- Challenge is to design distributed protocols which operate with only local knowledge.

A P2P Protocol

A distributed protocol to build P2P networks with provable guarantees under a reasonable model:

- connectivity.
- logarithmic diameter.
- constant degree.
- low overhead.
- operates with no global knowledge.
- can be easily implemented with local message passing.

Protocol Basics

A set of rules applicable to various situations a node may find itself in:

- How to join the network ?
- What happens if a neighbor drops out ?
- How to maintain bounded number of connections ?

A central host server:

- a gateway mechanism to enter the network.
- maintains a **cache** - a list of K $(=$ constant$)$ nodes (i.e., their IP addresses) at all times.
- is reachable by all nodes at all times.
- need not know the network topology nor the identities of all the nodes in the network.

- When a node is in the cache we refer to it as a *cache node*; accepts connections from other nodes.

- A node is *new* when it joins the network, otherwise it is *old*.

- Protocol will ensure that the degree (number of neighbors) of all nodes will be in the interval $[D, C+1]$, for two constants D and C.

- **c-node:** node that was a cache node at some time.

- **d-node:** all other nodes.

Protocol for Node v

1. **On joining the network:** Connect to D cache nodes, chosen uniformly at random from the current cache.

2. **Reconnect rule:** If a neighbor of v leaves the network, and that connection was not a preferred connection, connect to a random node in cache with probability $D/d(v)$, where $d(v)$ is the degree of v before losing the neighbor.

3. **Cache Replacement rule:** When a cache node v reaches degree C while in the cache (or if v drops out of the network), it is replaced in the cache by a d-node from the network. Let $r_0(v) = v$, and let $r_k(v)$ be the node replaced by $r_{k-1}(v)$ in the cache. The replacement d-node is found by the following rule:

   $k = 0$;
   while (a d-node is not found) **do**
   search neighbors of $r_k(v)$ for a d-node;
   $k = k + 1$;
   endwhile

4. **Preferred Node rule:** When v leaves the cache as a c-node it maintains a *preferred connection* to the d-node that replaced it in the cache. (If v is not already connected to that node this adds another connection to v.)

5. **Preferred Reconnect rule:** If v is a c-node and its preferred connection is lost, then v reconnects to a random node in the cache and this becomes its new preferred connection.

Analysis

Model:

- The arrival of new nodes – Poisson distribution with rate λ.

- The duration of time a node stays connected to the network is independently and exponentially distributed with parameter μ.

- Let G_t be the network at time t. Interested in the evolution in time of the stochastic process $\mathcal{G} = (G_t)_{t \geq 0}$.

- The evolution of the size of $\mathcal{G}$ depends only on the ratio $\lambda/\mu = N$, the steady state size of the network.

Network Size

W.l.o.g let $\lambda = 1$; thus $\mu = 1/N$.

Let $G_t = (V_t, E_t)$ be the network at time t.

Theorem 1. *1. For any $t = \Omega(N)$, w.h.p. $|V_t| = \Theta(N)$.*

2. If $\frac{t}{N} \to \infty$ then w.h.p. $|V_t| = N \pm o(N)$.