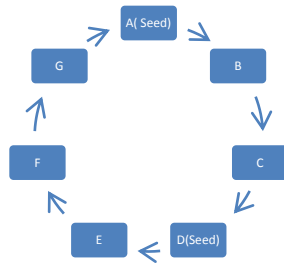
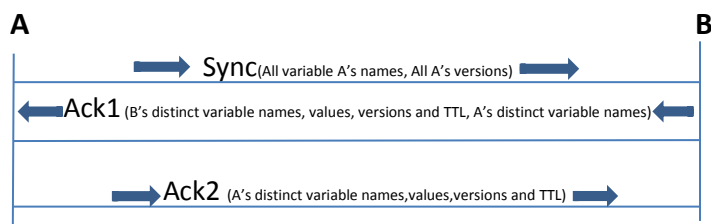


A model of the problem

- 1- We can consider each node as some vertex in a graph of n vertices.



- 2- Each node keeps a list of variables it has receives from other nodes.
3- Each node initially knows of seeds only.
4- Seeds are nodes that are known to all other nodes by default.
5- Each node starts learning about non-seed nodes through the seeds.
6- The system can have many seeds.
7- After learning about all other nodes each nodes can reach them directly.
8- All nodes communicate through a handshake which depending on the nodes could consist of 2 or 3 messages.
9- Nodes can reach each other simultaneously.
10- Each variable on a node has a TTL (Time to Live) value.
11- $TTL=0$ implies that the variable and as a result the node will be forgotten.
12- Seeds' variables have $TTL=\infty$.
13- Nodes update the version of their variables after a while in order to avoid being forgotten.
14- The time to process variables on a receiving node is almost Zero.
15- Each node picks only one other node at a time with equal probability.
16- Each node waits for 500 milliseconds before initiating a new handshake but is accessible to all other nodes in the meantime.
17- The handshake protocol is modeled as follows:



- 18- If a node's information about another node is not updated the latter node is forgotten.
- 19- No node selects itself for handshake.
- 20- Each node initiate a handshake only once in each 500 millisecond interval.
- 21- There's an upper bound of 1000 mess/sec for the total number and 10^8 bytes for the size of messages a node is capable of sending and receiving within each connection interval.
- 22- The average message size on the system is 10^3 bytes.
- 23- The maximum message size on the system is 10^4 bytes.
- 24- The number of messages takes precedence over their size in calculating the efficiency of the DRL, meaning that as long as the number of messages stays constant it doesn't matter how one divides some bytes of information between them.
- 25- Each node keeps itself remembered on other nodes by updating the version of its variable once every while.