Leader Election

Leader Election

- Elect one node as leader
- Requirements
  - Terminates
  - Leader process knows it is the leader on termination
  - All other processes know they are not the leader on termination, and (optionally) knows who the leader is
- Have been studied in different topology
  - Rings
  - Arbitrary topology

Leader Election in Rings

- Models
  - Synchronous or Asynchronous
  - Unidirectional or bidirectional ring
  - Anonymous (no unique id) or Non-anonymous (unique ids)
  - Uniform (no knowledge of '*n*', the number of processes) or non-uniform (knows '*n*')
- Known Impossibility Result
  - There is no deterministic, synchronous, non-uniform leader election protocol for anonymous rings

Lelann-Chang-Robert's Algorithm

- Model
  - Asynchronous, reliable, unidirectional ring, unique ids
- Algorithm
  - Send own id to node on left
  - If an id received from right, forward id to left node only if received id greater than own id, else ignore
  - If own id received, declares itself "leader"
- Worst case message complexity $= O(n^2)$
  - Can you draw a ring with ids and a sequence of message exchanges that achieves this worst case?

Hirschberg-Sinclair Algorithm

- Model
  - Same as LCR, but requires bidirectional ring
- Operates in phases
- In k-th phase ($0 \leq k \leq \lg n$), send probe with own id to 2^k processes on both sides of yourself
  - Directly send only to neighbors with id and TTL = 2^k in it
- If id received
  - If own id > received id, drop the probe (and start algo with own id as above if not already done so)
  - Else
    - If TTL > 0, forward probe with TTL = TTL-1
    - Else send reply to originator

- Replies are always forwarded irrespective of id
- A process goes to (k+1)-th phase if and only if it receives a reply from both sides in k-th phase
- Process receiving its own id in a probe – declare itself "leader" (id must have traversed and passed by all other nodes)
- Leader can then circulate a leader message around the ring to inform all other nodes who the leader is

- Message Complexity: O($n$lg$n$)
  - After (k-1)-th phase, at most $n/(2^{k-1} + 1)$ nodes can be alive
  - Each of these nodes can send at most 4.2^k messages
    - 2^{k+1} probes and 2^{k+1} replies on each side
  - $k = O(\lg n)$

Lower Bound for Rings

- Can we do it in O(nlgn) time in unidirectional rings?
  - Yes (Peterson's algorithm)
- Lots of other algorithms exist for rings
- Lower Bound result:
  - Any comparison-based leader election algorithm in a ring requires $\Omega(n \lg n)$ messages
- What if not comparison-based?
  - Can you break the message lower bound with increased time?

Variable Time Algorithm

- Synchronous, round based
- Round = maximum message transmission delay
- Phase = n rounds
- Node k does the following
  - If no message received when k-th phase starts, declare itself the leader and send a leader message with its id around the ring
  - If message received before k-th phase starts, record id in message as leader and forward the message
- Message complexity O(n)
- Time complexity O(K), where K is the lowest id in the ring

Leader Election in Arbitrary Networks

- FloodMax
  - Synchronous, round-based
  - At each round, each process sends the max. id seen so far (not necessarily its own) to all its neighbors
  - After diameter no. of rounds, if max. id seen = own id, declares itself leader
  - Complexity = O(d.m), where d = diameter of the network, m = no. of edges
  - Does not extend to asynchronous model trivially
- Variations of building different types of spanning trees with no pre-specified roots. Chosen root at the end is the leader
  - Will study DFS spanning tree later