Computing in Distributed Systems
Chapter 2: Leader Election

Akka Zemmari

LaBRI - Université de Bordeaux

Definitions

Leader Election

Each node eventually decides whether it is a leader or not, subject to the constraint that there is *exactly* one leader.

Why electing

- Once elected, the leader may centralise some informations
- the leader can initiate some reset procedure
- the election protocol can be used to solve mutual exclusion problems,
- ...

Impossibility result

Theorem

Electing a leader in anonymous ring is impossible.

Proof: see the whiteboard.

Electing in non-anonymous rings

We assume the nodes of the ring have unique IDs.

Algorithm 1.

each node v does the following:

- v sends a message with its $ID(v)$ to clockwise neighbour
- if v receives message $ID(w)$ with $ID(w) > ID(v)$ then
  - v forwards $ID(w)$ to clockwise neighbour
  - v decides not to be the leader
- else if v receives its own $ID(v)$ then
  - v decides to be the leader

Electing in non-anonymous rings

Analysis of the algorithm

Theorem. Algorithm 1 is correct, its time complexity is $O(n)$, and its message complexity is $O(n^2)$.

Proof: see the whiteboard.

Electing in anonymous rings

We assume the nodes of the ring are anonymous but they know n the size of the ring.

Algorithm 2.

each node v does the following:

- $id_v = \begin{cases} 1 & \text{with probability } 1/\text{n} \\ 0 & \text{with probability 1-}1/\text{n} \end{cases}$
- v sends a message with $< id_v >$ to clockwise neighbour
- v receives message M from its neighbour w
- If $\mid M \mid = n$ then
  - If id_v is the unique maximum in M then
    - v decides to be the leader
  - else
    - v decides not to be the leader
- else v sends $< M.id_v >$ to clockwise neighbour

Electing in anonymous rings

Analysis of the algorithm

Try to answer the following questions:

1. When the algorithm terminates with a leader node ?
2. What is the probability for the algorithm to end with an elected node ?
3. What is the messages complexity of the algorithm ? Is it a random variable ?
4. What is the category of this algorithm ?
5. Can this algorithm be transformed into a Las Vegas algorithm ? and if so, analyse its complexity.

Responses: see the whiteboard.