

Leader Election

COMP750

Leader Election

- The purpose is to select one process from many.
- Once a leader is selected, all nodes need to be informed of the new leader.
- useful to provide failure recovery in many centralized algorithms. If the original leader fails, a new leader must be elected.
- It can also be used in a token based system when the token is lost.

Failure Assumptions

- Failing processes halt and do nothing.
There are no malicious processes.
- Failures are usually detected by timeouts, so functioning processes must produce a result within a maximum time period.
- Recovering processes know they have halted.

Bully Algorithm

- All processes have a priority.
- All processes know of all other processes and know their priority.
- Upon completion, all processes agree on the same leader.
- There are several bully algorithms in the literature. This one is slightly different from the text.

Bully Algorithm

- There are three types of messages:
 - election — announce an election
 - answer — response to an election message
 - coordinator — announces the winner

Bully Algorithm

When a node feels the need to elect a leader:

- Send an election message to all higher priority nodes.
- If no answer message is received, the node declares itself the leader and sends a coordinator message to all other nodes.
- If an answer message arrives, it waits for a coordinator message.
- If a node receives an election message, it sends back an answer message and begins another election unless one is underway.

Bully Analysis

- In the best case, the second highest node realizes the coordinator is dead. It can immediately elect itself and send $n-2$ coordinator messages.
- This bully algorithm requires N^2 messages in the worst case when the lowest priority process calls the election.

Ring based election algorithm

- All processes are connected in a logical ring.
- A process can only know of the one process ahead of it in the ring.
- Each process has an identifier/priority.

Ring based election algorithm

When a node feels the need to elect a leader:

- Send an election message to neighbor with its ID and mark itself as a participant..
- When a node receives an election message it compares the ID in the message with its own.
- If the msg ID is greater, it forwards the message.

Ring based election algorithm

- if the msg ID is smaller and it is not a participant, it sends an election message with its own ID and marks itself as a participant.
- If the msg ID is the same as the node, it is the leader. It sends a coordinator message.
- When a node receives a coordinator message, it says the leader ID and forwards the message until it returns to the leader.