

Assignment 4

Problem Statement: To develop any distributed algorithm for leader election.

Tools: Java, Ubuntu

Theory:

Distributed Algorithm: Distributed Algorithm is an algorithm that runs on a distributed system. Each processor has its own memory and they communicate via communication networks.

Many algorithms used in distributed systems require a coordinator that performs functions needed by other processes in the system.

Election algorithms are designed to choose a coordinator

Why Election Algorithm?

- Many distributed algorithms require a process to act as a coordinator.
- The coordinator can be any process that organizes actions of other processes.
- A coordinator may fail.

Assumptions:

Each process has a unique number to distinguish them. Processes know each other's process number.

There are two types of Election Algorithms:

- Bully Algorithm
- Ring Algorithm

Bully Algorithm: This algorithm applies to systems where every process can send a message to every other process in the system.

Algorithm: Suppose process P sends a message to the coordinator.

- If coordinator does not respond to it within a time interval T, then it is assumed that coordinator has failed.
- Now process P sends an election message to every process with a high priority number.
- It waits for responses, if no one responds for time interval T then process elects itself as a coordinator.
- Then it sends a message to all lower priority number processes that it is elected as their new coordinator.
- However, if an answer is received within time T from any other process Q,
 - Process P again waits for time interval T to receive another message from Q that it has been elected as coordinator.
 - If Q doesn't respond within the time interval T' then it is assumed to have failed and the algorithm is restarted.

Ring Algorithm: This algorithm applies to systems organized as a ring (logically or physically). In this algorithm we assume that the link between the process are unidirectional and every process can message to the process on its right only. Data structure that this algorithm uses is active list, a list that has priority number of all active processes in the system.

Algorithm:

- If process P1 detects a coordinator failure, it creates a new active list which is empty initially. It sends election messages to its neighbour on the right and adds number 1 to its active list.
- If process P2 receives message elect from processes on left, it responds in 3 ways:
 - If the message received does not contain 1 in active list then P1 adds 2 to its active list and forwards the message.

- If this is the first election message it has received or sent, P1 creates a new active list with numbers 1 and 2. It then sends election message 1 followed by 2
- If Process P1 receives its own election message 1 then the active list for P1 now contains numbers of all the active processes in the system. Now Process P1 detects the highest priority number from the list and elects it as the new coordinator.

Conclusion: In this assignment we have studied about election algorithms and have implemented the Bully and Ring algorithm.