Distributed Systems

Tim Bachman, Nour Shokry

September 17th, 2021

Task 1

a)

- Synchronous vs. Asynchronous
  - Synchronous execution: All messages in out-buffer delivered at once. There is one exec event in all processors. Nodes operate in synchronous rounds. In each round, each node sends messages to its neighbors, receive messages that were sent by neighbors of the same round, and do some local computation.
  - Asynchronous execution: Messages are sent between nodes one by one, unlike synchronous execution which can send multiple messages at the same time. There can be arbitrary message delay and the processor speeds for each node may differ.
  - Synchronous model is simpler to implement than asynchronous model.
- Anonymous vs. Non-anonymous
  - Anonymous: no possibility to distinguish or compare the nodes, no unique identifier and all. Leader election is not possible in an anonymous ring.
  - Non-Anonymous: Nodes have an identifier. Leader election is possible.
- Uniform vs. Non-uniform
  - Uniform: all nodes behave the same way for the same input.
  - Non-uniform: The nodes can have different algorithms and behave differently even for the same input.

Uniform: nodes don't know how many nodes are in the system

b)

Yes, it is possible to elect a leader, which is going to be the only node the has ID = 1.

Algorithm:

```
if ID = 0 then do nothing
if ID = 1 then elect self
```

c)

- If n = 3, it's not possible to elect a leader since the three nodes will have the same ID = 1, hence they will be anonymous.
- If n = 4, it is possible to elect a leader, in which the leader has ID = 0 and the other three nodes have ID = 1.

  Algorithm:

  ```
  if ID = 1 then do nothing
  if ID = 0 then elect self
  ```

- If n > 4, it's not possible to elect a leader. Since we have two sets of nodes, where the first set has nodes of ID = 0, and the second set has nodes of ID = 1. Within both sets, the nodes are anonymous, so we cannot determine the leader.

d)

The node having ID = 1 will think it's a leader, since it doesn't know about the other nodes that also have the same IDs. Therefore multiple nodes of ID = 1 will think they are leaders, and there will be more than 1 leader. Therefore, leader election is not possible in this case.

Task 2

a

We elect the most centered node as the leader. We check each path from the current node to every leaf node in the graph and add the distances of all the paths. The node which has the minimum total distance to all leaf nodes is the leader.

Finding the total distance to all leaf nodes

The election goes as follows: The inner node send packets to each leaf node having degree = 1. The packets are returned from every leaf node with their path distances. The inner node then calculates the total path distances it received from all the leaf nodes, and the inner node with least total distance is the leader.

To send a packet to the leaf nodes, a special type of packet is sent to each neighbour of the current node. This packet is extended on every step to contain the full path it took to reach a leaf node, which allows it to ”find its way back” and simultaneously keep track of the distance to the leaf node. Every node that receives such a package will append the source and destination neighbour to the package contend and forward it to the respective neighbour. If a node has multiple neighbours to forward it to, it will send a different copy of the packet to each neighbour. If a node is a leaf node and does not have any neighbours to forward it to, it marks the package as ”returning” and sends it back. A returning packet follows the same path it came from.

If such a returning packet is received there are two possibilities: The packet is destined for the current node, or it is still on the way. If it is still on the way, the packed is updated by ticking ”ticking off” the current step by setting the corresponding Returning value to true and forwarding it to the neighbour as described in the packet.

If all Returning values are set to true, this means the Packet is destined for the current node. The node counts the number of steps encoded in the packet.

The packet consists of a list where the entries are structured in the following way:

- Step:
  The path's distance count, which increments from an inner node to the leaf node
- Source:
  Every node can distinguish its neighbours by a number. The source field is a number which denotes which neighbour the packet came from.
- Destination:
  The destination is a number which denotes which neighbour the packet will be sent to.
- Returning:
  The Returning status is set to False along the path from the inner node to the leaf node.

  When a leaf node returns the distance back to the inner node it uses the return route via the source and destination node that were stored and the packets returned from the leaf node to the inner node has Returning status set to True.

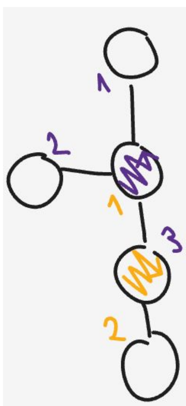
Finding the minimal total distance for all nodes

Every node knows its total distance to all leave nodes. To elect a leader, it must be clear what node has the minimal total distance. To do this every node sends a packet containing only its own total distance to all neighbours.

Every node keeps track of a value *min_distance* which is initialized to its own total distance. Every received total distance is compared to the current *min_distance*, which is updated accordingly. Every time such an update happens, the current node forwards this new *min_distance* to all its neighbours.

A node is considered the leader if its total distance is equal to the *min_distance* value.

b



Purple Node

- Send Packet (P1) from the purple node to the top node (neighbour 1)

Step	Source	Destination	Returning
0	null	1	False

- Send Packet (P2) from the purple node to the left node (neighbour 2)

Step	Source	Destination	Returning
0	null	2	False

- Send Packet (P3) from the purple node to the orange node (neighbour 3)

Step	Source	Destination	Returning
0	null	3	False

- Forward packet (P3) from the orange node to the bottom node (neighbour 2)

Step	Source	Destination	Returning
0	null	3	False
1	1	2	False

- Returning packet (P1) from the top node to the orange node

Step	Source	Destination	Returning
0	null	1	True

- Returning packet (P2) from the left node to the orange node

Step	Source	Destination	Returning
0	null	2	True

- Forward returning packet (P3) from the bottom node to the orange node

Step	Source	Destination	Returning
0	null	3	False
1	1	2	True

- Forward returning packet (P3) from the orange node to the purple node

Step	Source	Destination	Returning
0	null	3	True
1	1	2	True

- $\Rightarrow$ The total distance to the leaf nodes is $1 + 1 + 2 = 4$

Orange Node Sending

- Send Packet (O1) from the orange node to the purple node (neighbour 1)

Step	Source	Destination	Returning
0	null	1	False

- Send Packet (O2) from the orange node to the bottom node (neighbour 2)

Step	Source	Destination	Returning
0	null	2	False

- Forward Packet (O1.a) from the purple node to the top node (neighbour 1)

Step	Source	Destination	Returning
0	null	1	False
1	3	1	False

- Forward Packet (O1.b) from the purple node to the left node (neighbour 2)

Step	Source	Destination	Returning
0	null	1	False
1	3	2	False

- Return Packet (O2) from the bottom node to the orange node

Step	Source	Destination	Returning
0	null	2	True

- Return Packet (O1.a) from the top node to the purple node

Step	Source	Destination	Returning
0	null	1	False
1	3	1	True

- Return Packet (O1.b) from the left node to the purple node

Step	Source	Destination	Returning
0	null	1	False
1	3	2	True

- Return Packet (O2.a) from the purple node to the orange node

Step	Source	Destination	Returning
0	null	1	True
1	3	1	True

- Return Packet (O1.b) from the purple node to the orange node

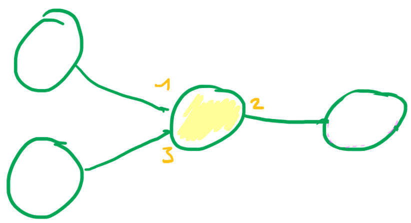
Step	Source	Destination	Returning
0	null	1	True
1	3	1	True

- $\Rightarrow$ The total distance to the leaf nodes is $2 + 2 + 1 = 5$

Finding minimal total distance

- All leaf nodes do not send any packets
- The purple node sends a packet with content `"4"` to the following nodes:
  - The top node
  - The left node
  - The orange node
- The orange node sends a packet with content `"5"` to the following nodes:
  - The purple node
  - The bottom node
- The orange node receives the lower value `"4"` and forwards it to:
  - The purple node
  - The bottom node
- All nodes agree that the number 4 is the lowest number and the purple node is the leader.

c.1



- Send Packet (M1) from the yellow node to the left top node

Step	Source	Destination	Returning
0	null	1	False

- Send Packet (M2) from the yellow node to the left bottom node

Step	Source	Destination	Returning
0	null	3	False

- Send Packet (M3) from the yellow node to the right node

Step	Source	Destination	Returning
0	null	2	False

- Return Packet (M1) from the left top node to the yellow node

Step	Source	Destination	Returning
0	null	1	True

- Return Packet (M2) from the left bottom node to the yellow node

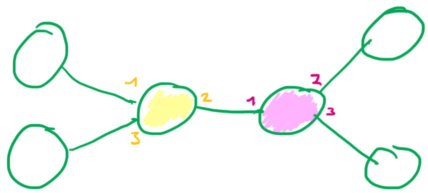
Step	Source	Destination	Returning
0	null	3	True

- Return Packet (M3) from the right node to the yellow node

Step	Source	Destination	Returning
0	null	2	True

- $\Rightarrow$ The total distance to the leaf nodes is $1 + 1 + 1 = 3$
- The yellow node sends a packet with content `"3"` to the the leaf nodes
- All leaf nodes do not send any packets
- All nodes agree that the number 3 is the lowest number and the yellow node is the leader.

c.2



Packets from the yellow node

- Send packet (Y1) from the yellow node to the top left node

Step	Source	Destination	Returning
0	null	1	False

- Return packet (Y1) from the top left node to the yellow node

Step	Source	Destination	Returning
0	null	1	True

- Send packet (Y2) from the yellow node to the pink node

Step	Source	Destination	Returning
0	null	2	False

- Send packet (Y2.a) from the pink node to the top right node

Step	Source	Destination	Returning
0	null	2	False
0	1	2	False

- Send packet (Y2.b) from the pink node to the bottom right node

Step	Source	Destination	Returning
0	null	2	False
0	1	3	False

- Return packet (Y2.a) from the the top right node to the pink node

Step	Source	Destination	Returning
0	null	2	False
0	1	2	True

- Return packet (Y2.a) from the the bottom right node to the pink node

Step	Source	Destination	Returning
0	null	2	False
0	1	3	True

- Return packet (Y2.a) from the pink node to the yellow node

Step	Source	Destination	Returning
0	null	2	True
0	1	2	True

- Return packet (Y2.b) from the pink node to the yellow node

Step	Source	Destination	Returning
0	null	2	True
0	1	3	True

- Send packet (Y3) from the yellow node to the bottom left node

Step	Source	Destination	Returning
0	null	3	False

- Return packet (Y3) from the bottom left node to the yellow node

Step	Source	Destination	Returning
0	null	3	True

- $\Rightarrow$ The total distance to the leaf nodes is $1 + 1 + 2 + 2 = 6$
- $\Rightarrow$ The yellow node sends the total distance 6 to all neighbours

Packets from the pink node

- Send packet (P1) from the pink node to the top right node

Step	Source	Destination	Returning
0	null	2	False

- Return packet (P1) from the top right node to the pink node

Step	Source	Destination	Returning
0	null	2	True

- Send packet (P2) from the pink node to the yellow node

Step	Source	Destination	Returning
0	null	1	False

- Send packet (P2.a) from the yellow node to the top left node

Step	Source	Destination	Returning
0	null	1	False
0	2	1	False

- Send packet (P2.b) from the yellow node to the bottom left node

Step	Source	Destination	Returning
0	null	1	False
0	2	3	False

- Return packet (P2.a) from the the top left node to the yellow node

Step	Source	Destination	Returning
0	null	1	False
0	2	3	True

- Return packet (P2.a) from the the bottom left node to the yellow node

Step	Source	Destination	Returning
0	null	1	False
0	2	3	True

- Return packet (P2.a) from the yellow node to the pink node

Step	Source	Destination	Returning
0	null	1	True
0	2	1	True

- Return packet (P2.b) from the yellow node to the pink node

Step	Source	Destination	Returning
0	null	1	True
0	2	3	True

- Send packet (P3) from the pink node to the bottom right node

Step	Source	Destination	Returning
0	null	3	False

- Return packet (P3) from the bottom right node to the pink node

Step	Source	Destination	Returning
0	null	3	True

- $\Rightarrow$ The total distance to the leaf nodes is $1 + 1 + 2 + 2 = 6$
- $\Rightarrow$ The pink node sends the total distance 6 to all neighbours

The pink and yellow nodes both consider them-self the leader.

Task 3

Part a

1)

The event e_1^2 happens before the event e_2^3 in the given linearization example, however in Figure 2 the arrow points from e_2^3 to e_1^2. Therefore, the given order of events is not a linearization, since it is not consistent with the figure.

2)

e_1^1, e_1^2, e_1^3, e_2^1, e_2^2, e_2^3, e_3^1

3)

e_1^4, e_3^2, e_3^3, e_3^4

Part b

1)

The cut is not consistent because the event e_1^3 which is located after the cut happens before the event e_2^4 which is located before the cut.

2)

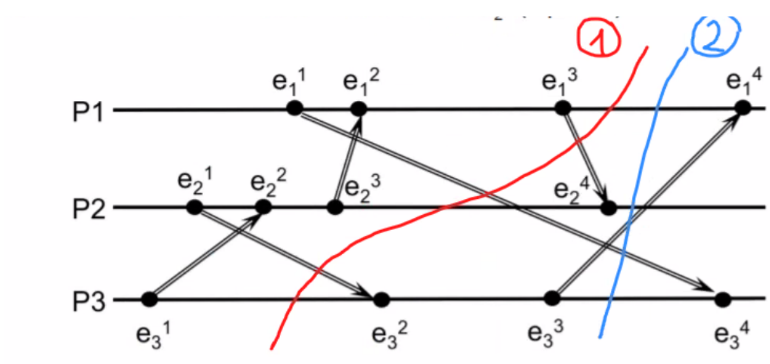


Figure 2