Distributed Algorithms

Homework 2

Group G10

Piotr Mrowczynski 387521
Gabriel Vilén 387555
Stefan Stojkovski 387529
Tong Li 387568

Exercise 1.1: Bully Algorithm

i. Describe the Bully Algorithm

The bully algorithm is a way to elect a leader in a distributed system. It is a form of election algorithm, a technique to selects a coordinator in a distributed algorithm. All process are assumed to known the ID of all other processes and a process initiates an election if it has just recovered from failure or if it notices that the coordinator has died.

In detail:
Any process P can initiate an election

- P sends *Election* messages to all process with higher IDs and awaits *OK* messages
  - – If no *OK* messages, P becomes coordinator and sends *Coordinator* messages to all processes with lower IDs
  - – If it receives an *OK*, it drops out and waits for an *Coordinator* message
- If a process receives an *Election* message
  - – Immediately sends *Coordinator* message if it is the process with highest ID
  - – Otherwise, returns an *OK* and starts an election
- If a process receives a *Coordinator* message, it treats sender as the coordinator

ii. How many messages are passed for a leader election with the bully algorithm

Each of the node in the topology will start a complete ring circulation. For one node, there will have n messages passed all around the circle. As we have n node, there will have n complete circulations. That's, in total we will have n^2 messages passed for a leader election with the bully algorithm.

iii. The Bully Algorithm is an example for a leader election. Give examples of applications that need a unique leader.

Determining the root node of a spanning tree

Exercise 1.2: Election

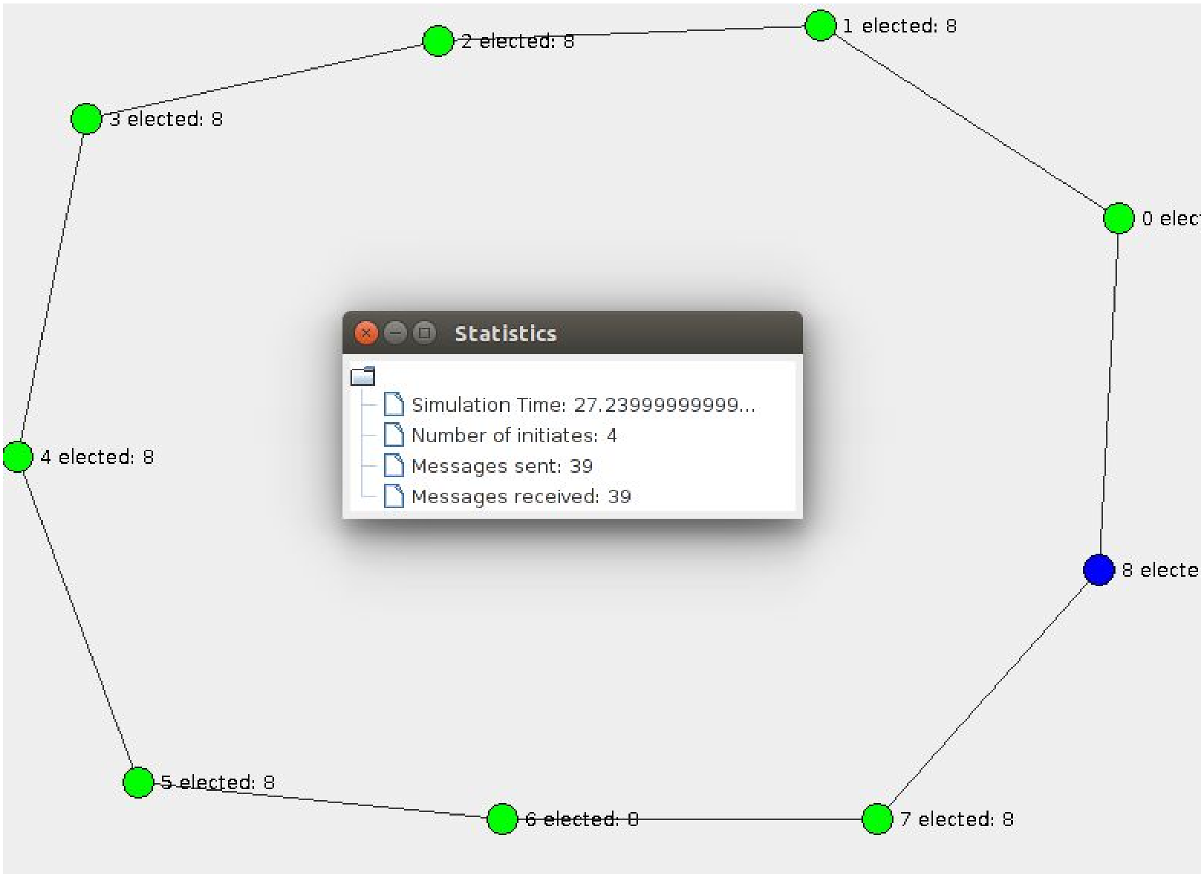
i. Implement the algorithm of Chang and Roberts that have been introduced in the lecture. Afterwards evaluate the message complexity of your implementation compared to the formulas provided in the lecture given the following scenarios:

1. WorstCase: Configure your topology that it reflects the worstcase scenario.

Worst case is given when the initiators are arranged in descending order and initiate the election in ascending order. This is because noone of the messages gets evicted and they have to travel the longest path before electing the leader.

Worst case #messages = $\frac{k(k-1)}{2} + n * k$

Equation for *k = 4, n = 9:*
Worst case #messages = 39 for k=4 and n=9

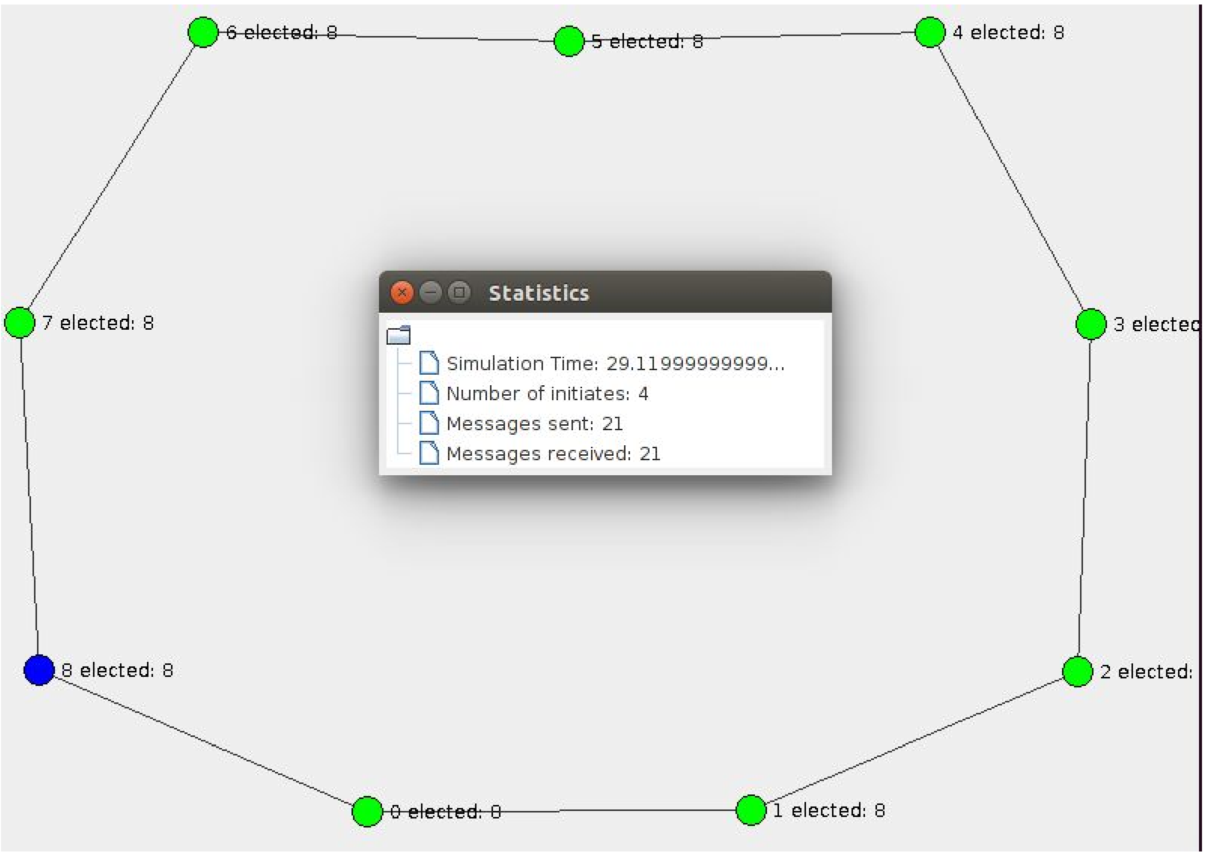


2. BestCase: Configure your topology that it reflects the bestcase scenario.

Best case occurs when the initiator are arranged in ascending order. This is because the next neighbours id is larger than the id of the initiator and the message can then be evicted.

Best case #message = 2n + k - 1

Equation for *k = 4, n = 9*
2*9 + 4 - 1 = 21,



3. AverageCase: Configure your topology that it randomly assigns node IDs and examine the average message complexity calculated over several runs.

average messages is calculating by running the topology x times, adding them and dividing by x.

We used k = 4, n = 9

Run1 : 30msg
Run2: 28msg
Run3: 35msg
Run4: 33msg
Run5: 30msg
Run6: 30msg

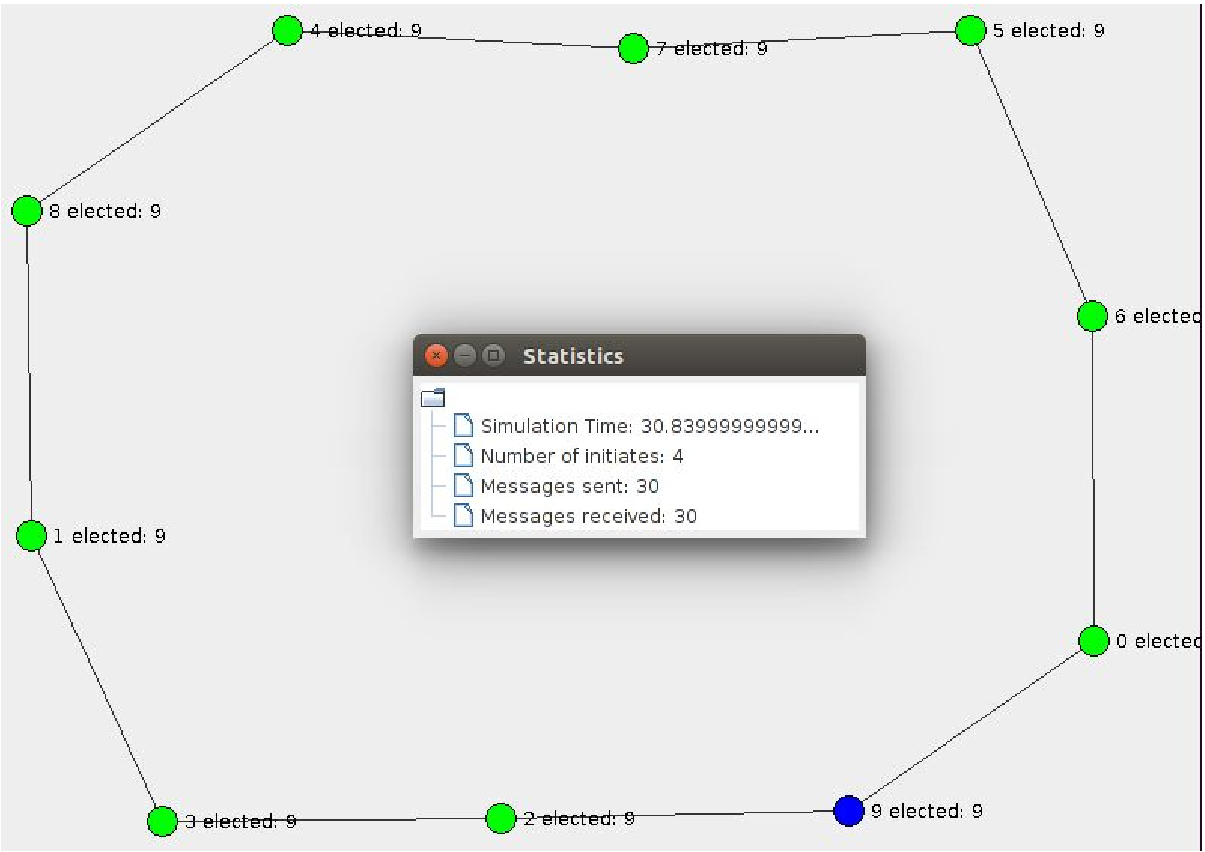
Run7: 31msg

average msgs = $\frac{\sum msg}{\#runs}$ = 31

The average messages is in the middle value of the worst and best-case, as can be calculated by this,

$$(worst\ case\ +\ best\ case)\ /\ 2\ =\ (39\ +\ 21)\ /\ 2\ =\ 30$$

Our average messages (31) is close to what we get from the average mean (30).



ii. There is a precondition for the algorithm of Chang and Roberts that all node IDs have to be unique (no duplicate IDs). Assume, we drop this precondition and allow multiple nodes to have the same ID.

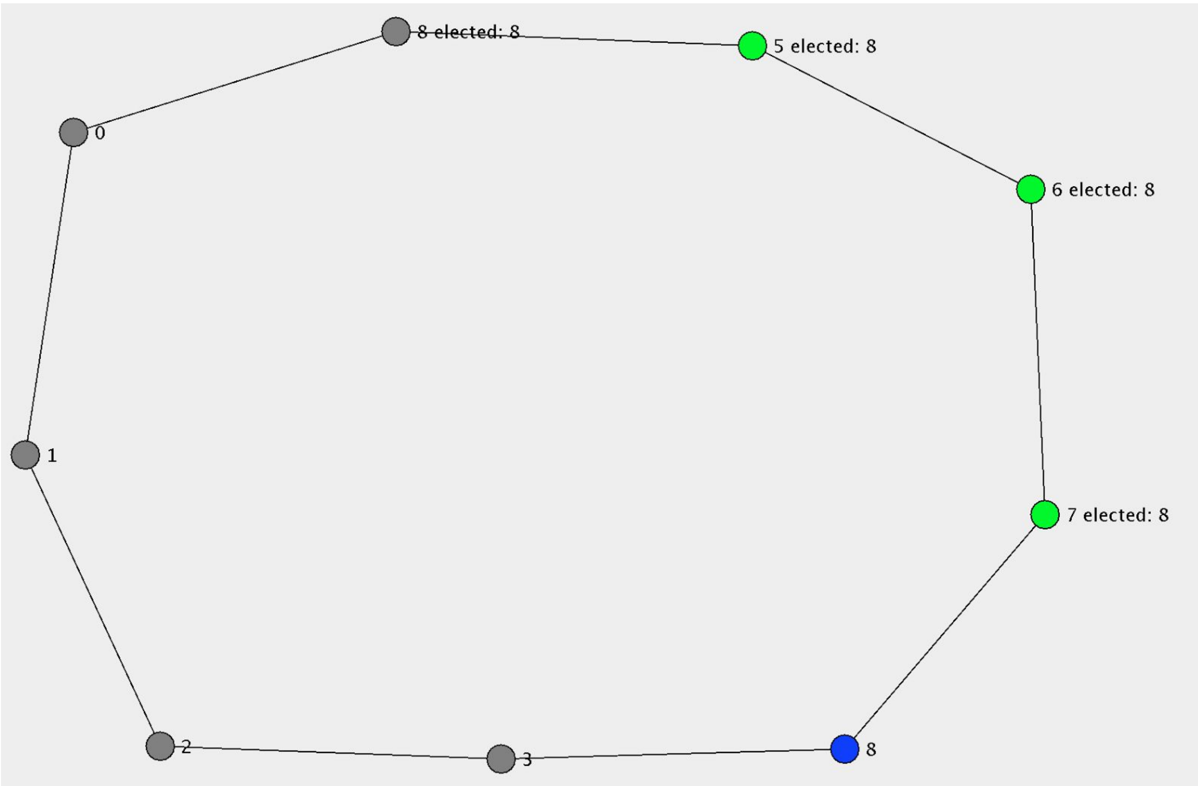
1. Does the algorithm still work properly? Please provide a reasonable answer.

Depends on the situations, the algorithm sometimes work exactly the same as the previous condition, but during most of the cases, the algorithm is not working properly if there is no the precondition of unique ID.

Generally, what we see is 8 different cases, and 5 different results.

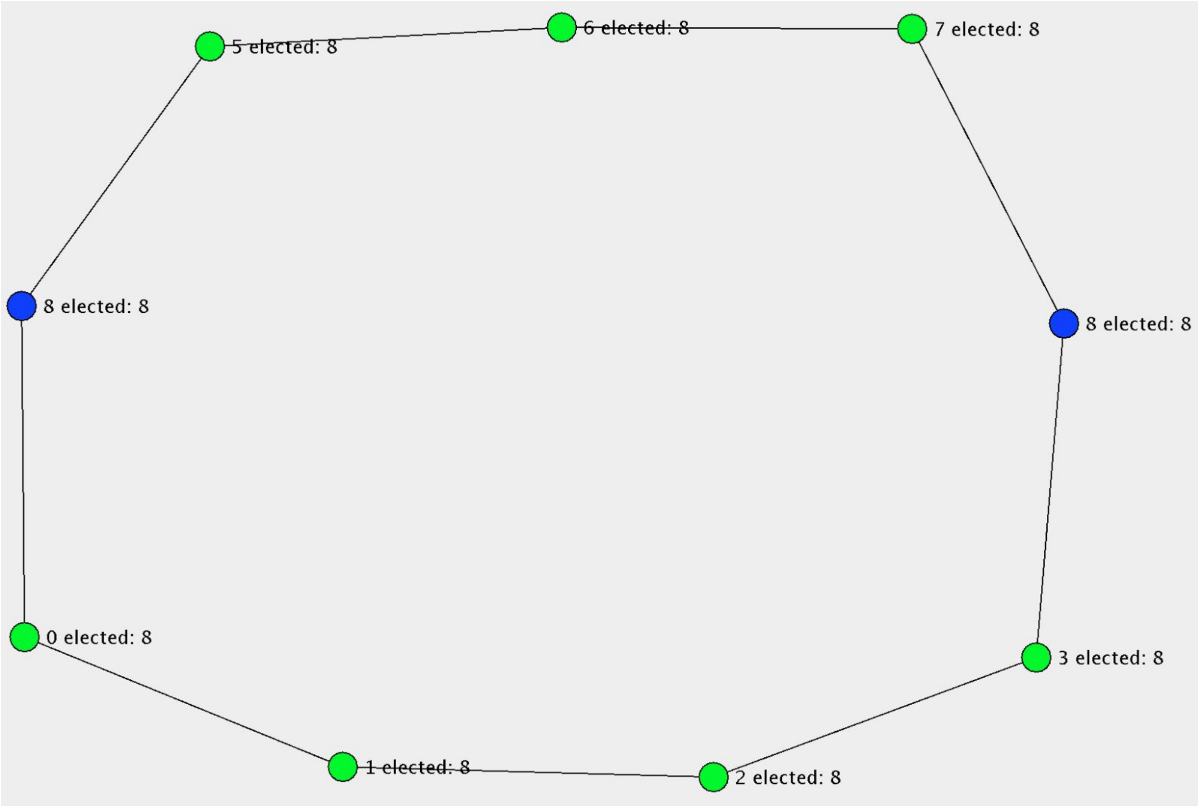
1) If the duplicate ID is the highest ID among the ring, all the initiators are none of these highest ID and all these initiators are located between only one side of the duplicated highest IDes. Then the result will be 1 leader elected. But the problem is that not all of the nodes on the ring are notified of the winner.

As we can see in the figure below, 8 is the duplicate highest ID among the nodes. Node 7 and node 5 are the two initiators and are located between the two Node 8. First Node 8 will forwarding the election message, the second election message received by first node 8 will be extinguished. When the second Node 8 receives message of 8. Second Node 8 is the winner, and the notification of the winner will terminate at the first Node 8. Resulting the rest of the nodes not notified.



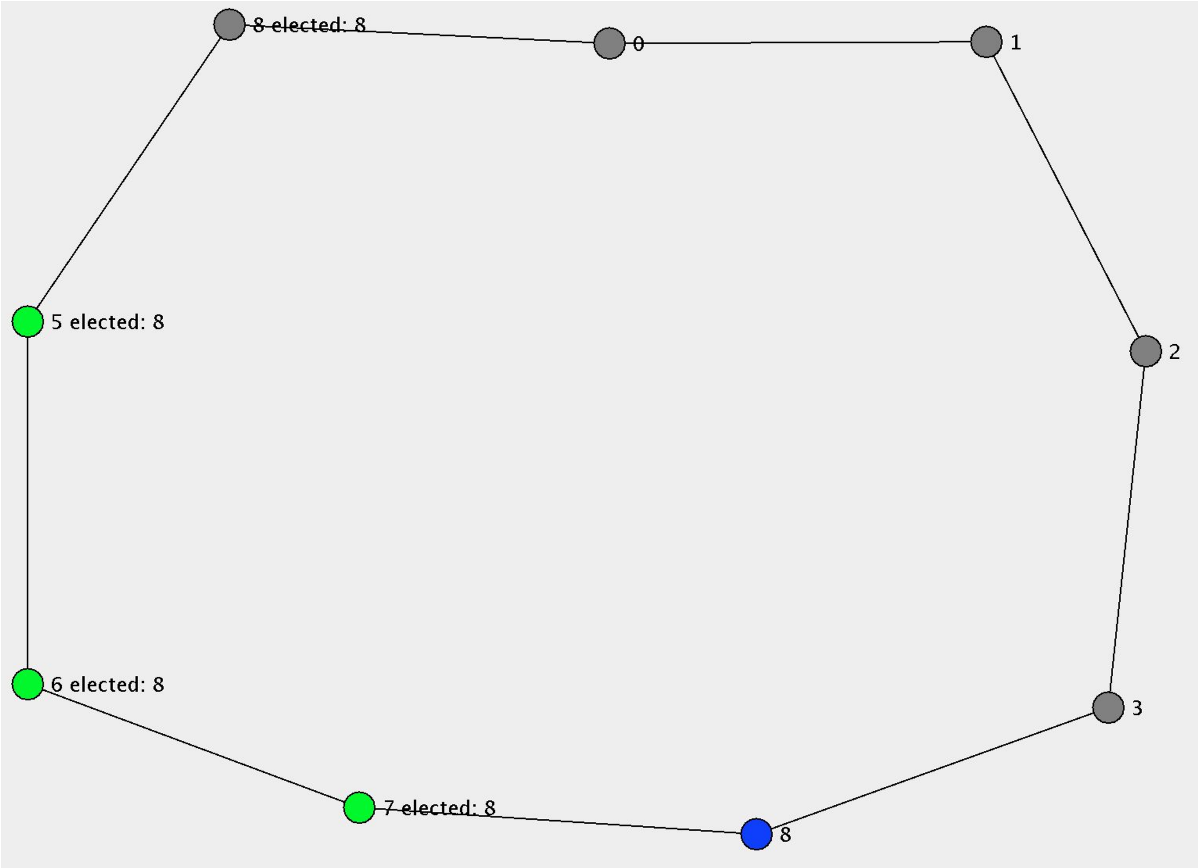
2) If the duplicate ID is the highest ID among the ring, all the initiators are none of these highest ID. And, the initiators are located between more than one side of the duplicated highest IDes. Then the result will be duplicated leaders elected.

As we can see in the figure below, 8 is the duplicate highest ID among the nodes. Node 7 and node 2 are the two initiators and are located at the two sides of the two Node 8. First Node 8 will forwarding the election message to the second Node 8, and the second 8 will forwarding the election message to the first Node 8. After the second Node 8 receive the election message from first Node 8, second Node 8 will become the winner, the same as the first Node 8. Each of the two Node 8 will win and will send out the notification message to the other nodes around. Resulting 2 Leaders in the network.



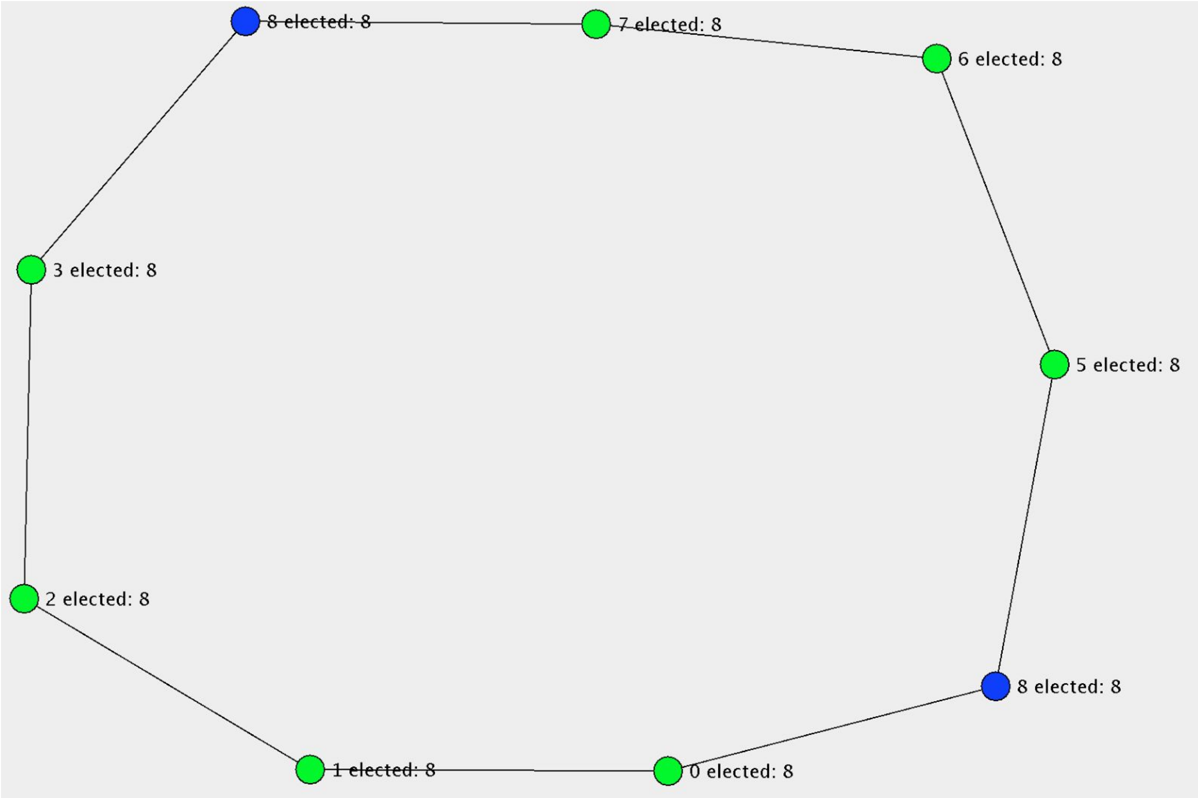
3) If the duplicate ID is the highest ID among the ring. Initiators are all the highest ID nodes, but not all the highest ID nodes are the initiators. Then the result will be 1 leader elected. But the problem is that not all of the nodes on the ring are notified of the winner.

   As we can see in the figure below, 8 is the duplicate highest ID among the nodes. One of the Node 8 is the initiator, let's say the first Node 8. The second Node 8 will be the winner and the winner notification message will terminate at the first Node 8. Resulting the rest of the nodes not notified.



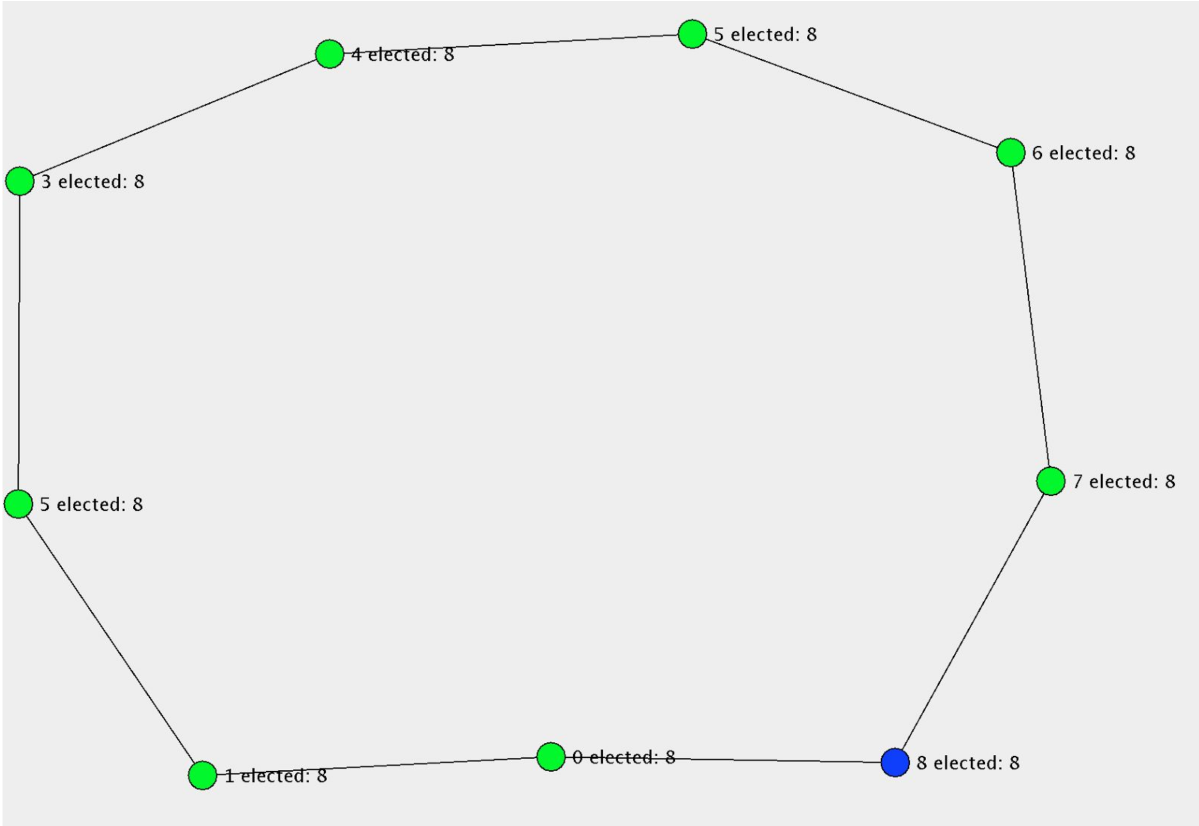
4) If the duplicate ID is the highest ID among the ring. Initiators are all the highest ID nodes, and all the highest ID nodes are the initiators. Then the result will be duplicated leader elected.

As we can see in the figure below, 8 is the duplicate highest ID among the nodes. Both of the two Node 8 are the two initiators. First Node 8 will forwarding the election message to the second Node 8, and the second 8 will forwarding the election message to the first Node 8. After the second Node 8 receive the election message from first Node 8, second Node 8 will become the winner, the same as the first Node 8. Each of the two Node 8 will win and will send out the notification message to the other nodes around. Resulting 2 Leaders in the network.



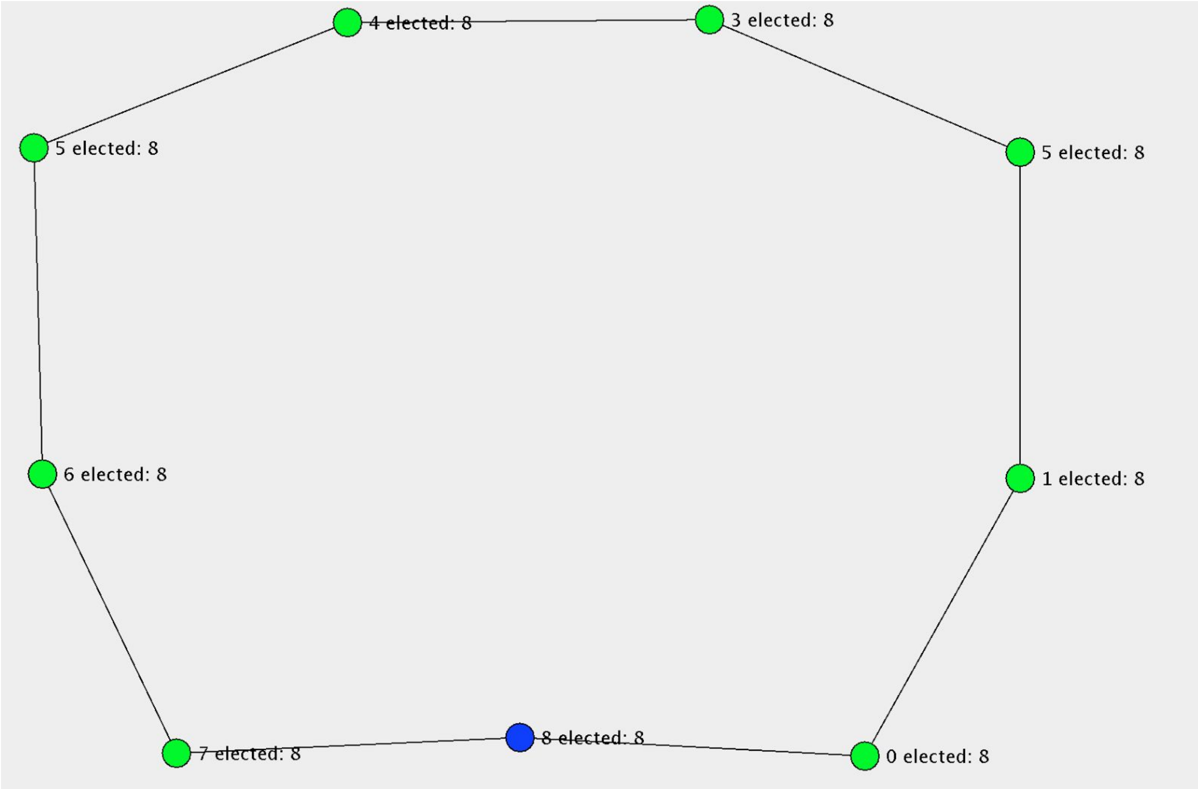
5) If the duplicate ID is not the highest ID among the ring. Initiator is the node with the highest ID. Then the result will be totally the same as the Chang and Roberts algorithm with precondition.

   As we can see in the figure below, 5 is the duplicate ID among the nodes. The Node 8 is the initiator. Then Node is going to be the winner and it will notify all other nodes. Just the same as the situation with precondition.



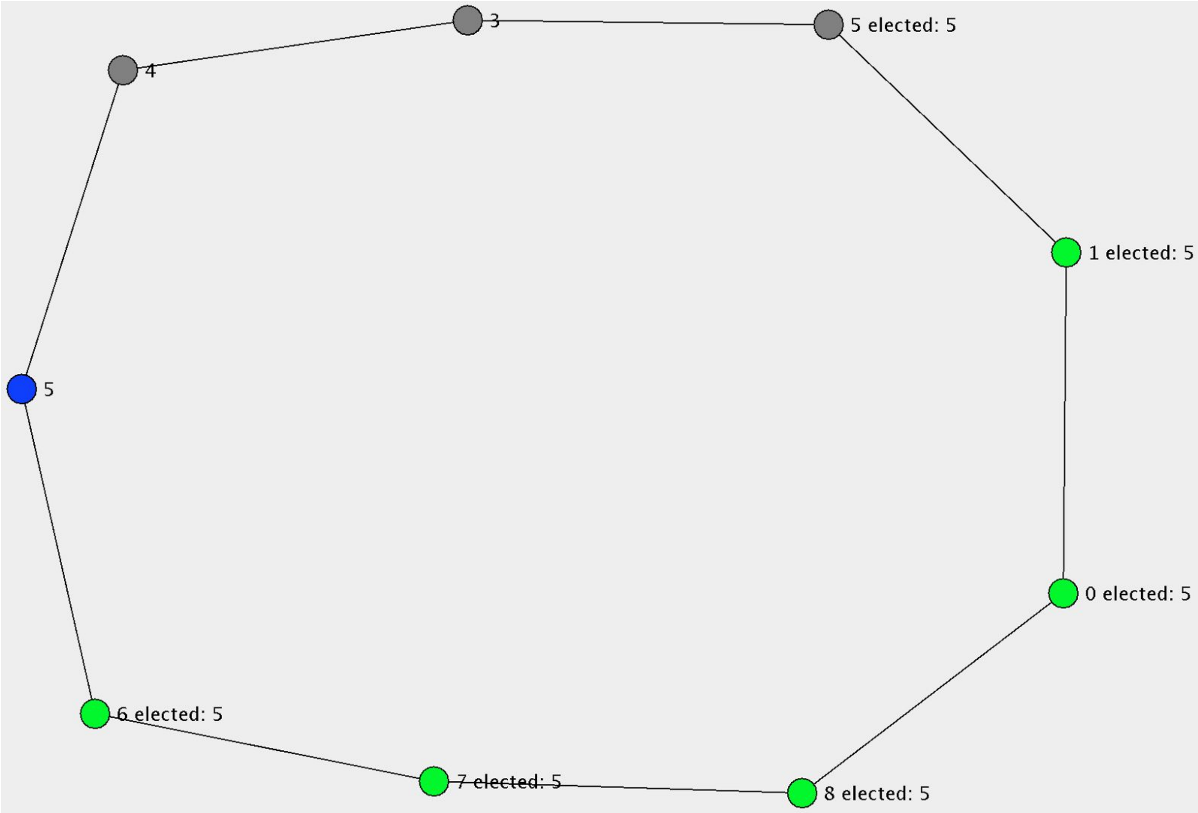
6) If the duplicate ID is not the highest ID among the ring. Initiator is not the node with the highest ID. And, If the initiate elective message is going to pass the node with highest ID first, other than the duplicate ID nodes. The result will be totally the same as the Chang and Roberts algorithm with precondition.

   As we can see in the figure below, 5 is the duplicate ID among the nodes. The Node 3 is the initiator. The election massage will pass to first Node 5, then Node 8, then to the second Node 5. The result is Node 8 is going to be the winner and it will notify all other nodes. Just the same as the situation with precondition.



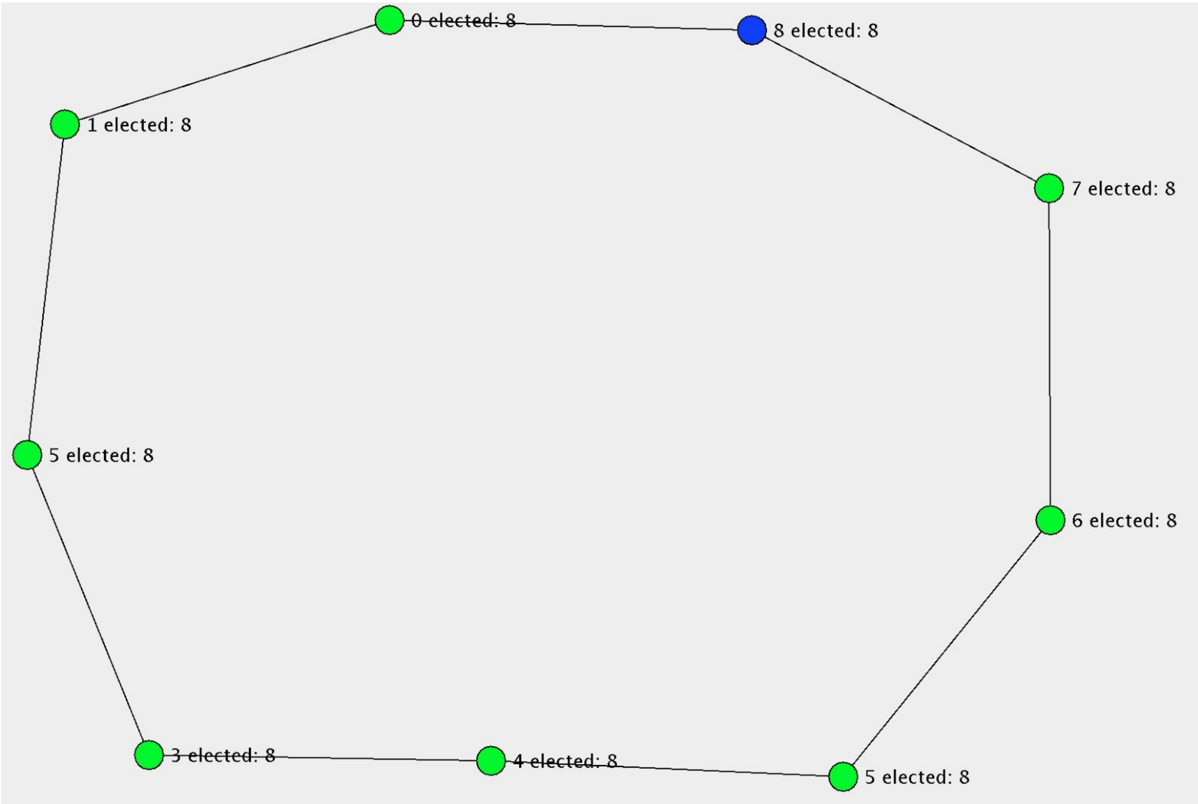
7) If the duplicate ID is not the highest ID among the ring. Initiator is not the node with the highest ID. And, If the initiate elective message is going to pass the duplicate ID nodes pairs first, other than the node with highest ID. The result is one of the duplicated nodes going to be elected as leader. Instead of the real highest ID node as the leader. The winner notification will not send to all of the nodes.

As we can see in the figure below, 5 is the duplicate ID among the nodes. The node 0 is the initiator. The election message will pass to the first and second Node 5(Namely, the duplicated node pairs). Then the second Node 5 is elected as the leader, the notification winner message will send to Node 8 and will terminate at the first Node 5. The result is Node 5 is the winner instead of Node 8. Part of the nodes are noticed of the winning results.



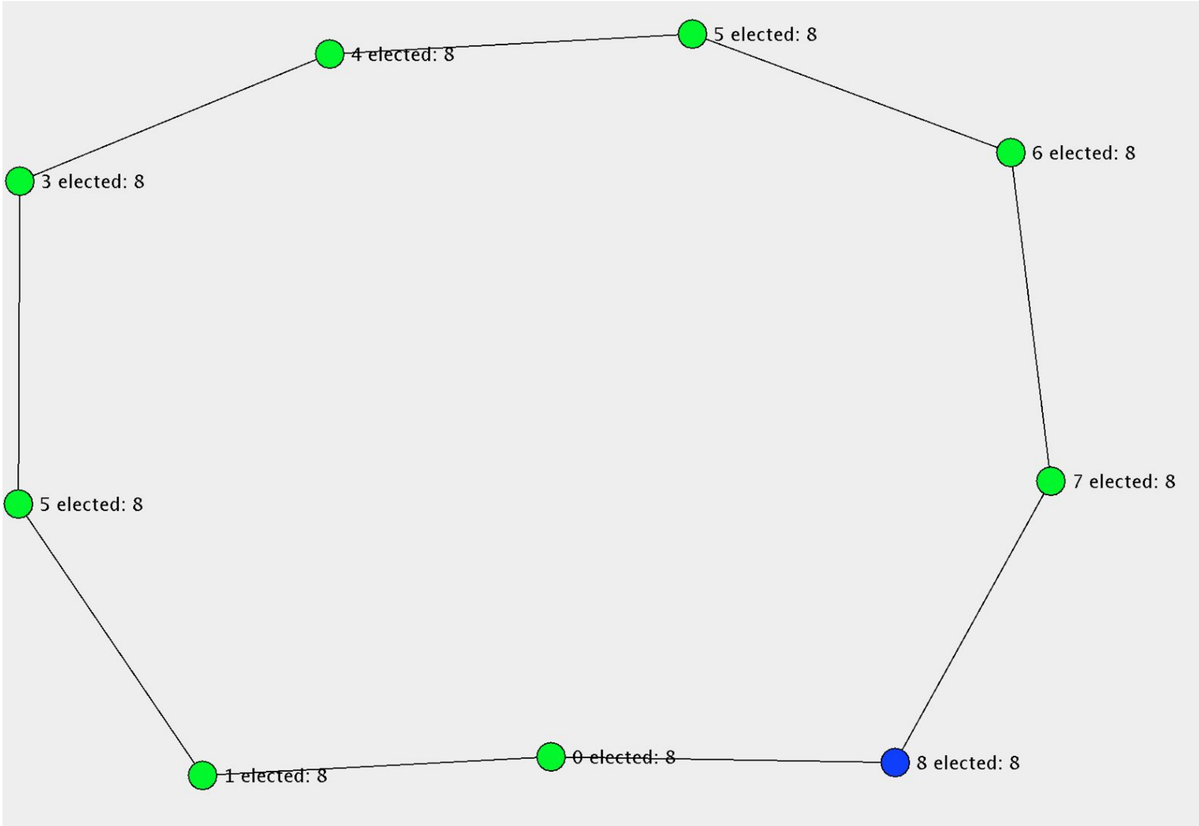
8) If the duplicate ID is not the highest ID among all the nodes. And, we have several initiators among all the nodes. Like the combination of case 6) and case 7). The result will be totally the same as the Chang and Roberts algorithm with precondition. The only problem is that, in this case, there will cause more messages sent on the topology.

As we can see in the figure below, 5 is the duplicate ID among the nodes. The node 0 and node 3 are the initiators. The two election messages will elect two leaders, Node 5 and Node 8 respectively. Then after receiving the winner notification message from Node 8, node 5 will also recognise Node 8 as the elected leader. The result is totally the same as the Chang and Roberts algorithm with precondition. But, as the Node 5 will send out the Node 5 leader notification message to other nodes, it will generally cause more message on the topology.

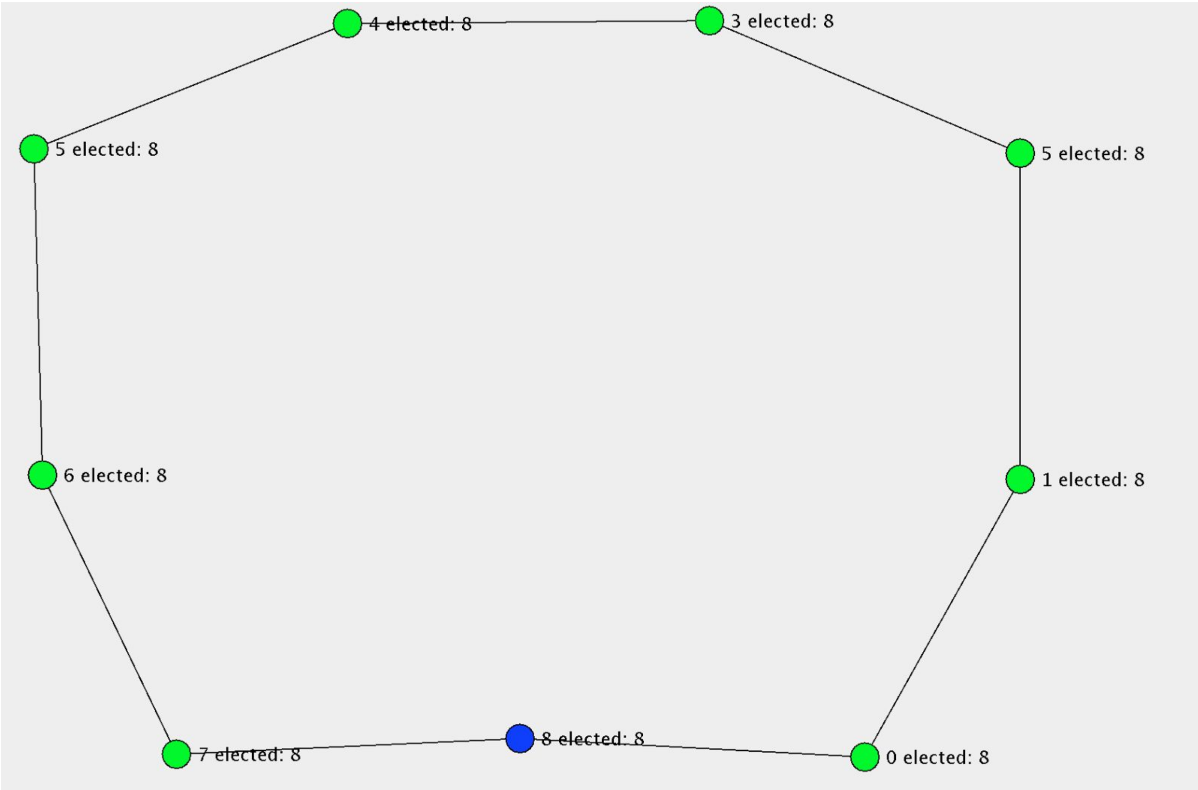


2. In which cases does the algorithm still deliver a proper result? Explain at least two cases.

As we can see in the answer from 1.2 ii 1, the case 5) and case 6) will result in a result which is the same as the Chang and Roberts algorithm with precondition.
For case 5)
If the duplicate ID is not the highest ID among the ring. Initiator is the node with the highest ID. Then the result will be totally the same as the Chang and Roberts algorithm with precondition. As the figure shown below.



For case 6)
If the duplicate ID is not the highest ID among the ring. Initiator is not the node with the highest ID. And, If the initiate elective message is going to pass the node with highest ID first, other than the duplicate ID nodes. The result will be totally the same as the Chang and Roberts algorithm with precondition.



Exercise 1.3: Process MeshAlgorithm (Maekawa)

i. What is the advantage of the Maekawa algorithm against the Broadcast Algorithm

The advantage of the Maekawa algorithm against the Broadcast algorithm is that the number of messages per access is lower. Maekawa has 3[(2√n) – 2] messages per access altogether (request, confirmation, release), while Broadcast has 3 (n – 1) messages per access altogether.
Each process does not ask for permission from all the processes, but from a subset of processes.

ii. The Meakawa Algorithm is not deadlock free. Give an example of a deadlock situation.

Assume three sites Si , Sj , and Sk simultaneously invoke mutual exclusion
Suppose Ri ∩ Rj= {Sij}, Rj ∩ Rk= {Sjk }, and Rk ∩ Ri= {Ski}.
Consider the following scenario:
Sij has been locked by Si (forcing Sj to wait at Sij)
Sjk has been locked by Sj(forcing Sk to wait at Sjk)
Ski has been locked by Sk (forcing Si to wait at Ski).
This state represents a deadlock involving sites Si , Sj , and Sk .