COMS 6998-006 (Foundations of Blockchains)
Homework Set 1 (Sol)

Matin M.Babaei M. Parsa Dini

September 9, 2023

Problem 1

Without cryptographic assumptions, Byzantine broadcast and agreement can be solved if $f < \frac{n}{3}$. With assumptions on digital signatures, Byzantine broadcast can be solved if $f < n$ and Byzantine agreement can be solved if $f < \frac{n}{2}$.

Simple But Enough. As usual, it is the conjunction of Safety(Agreement) and Liveness(Validity) that makes the problem hard. But if all we want is agreement(and termination), every node can always output a canonical value or if all we care is validity(and termination) each node can just output its private input. We found such simple protocols so obvoius and axiomatic that we don't discuss them. Instead, There are other more interesting protocols that might be considered more practical.

A. Here are two deterministic Byzantine Agreement protocols, one that satisfies Agreement and another that satisfies Validity. Note these protocols are just some suggestions and many other protocols can satisfy these properties as well. Others are left to the reader.

1. Protocol for Agreement: Majority Voting Algorithm (+ Echo Algorithm)

Description:

1) Each node broadcasts its private input(bit) to all other nodes.

2) Each node echos the bits it has received. Considering synchrony, note that under digital signature and PKI assumption, nobody can forge a message(i.e signature). Therefore, if byzantine nodes try to forge a message, honest nodes will notice that and neglect their messages.

3) Upon receiving messages from other nodes, each node sorts the received inputs and selects the median value (middle value) as the agreed-upon output. If median value happens to be 0.5, the node can choose either the lower or the higher middle value, as the protocol guarantees agreement on either choice. Thus, all nodes output the majority vote as their agreed-upon value.

In the f = 1 case, no echo is needed. In the f = 2 case, one echo is necessary and so on.

Proof of Agreement:

In the synchronous model, all nodes receive messages simultaneously, ensuring that honest nodes receive the same set of inputs from other nodes. By sorting the received inputs and selecting the median value, the protocol ensures that no matter the behavior of Byzantine nodes, they cannot force the honest nodes to output a value that is not among the original inputs. Byzantine nodes can attempt to manipulate a subset of inputs, but they cannot affect the median value without changing the inputs from multiple honest nodes. Honest nodes will output their own input if it is the median, guaranteeing that all honest nodes will agree on the output.

2. Protocol for Validity: King Algorithm

Description:

1: x = my input value

2: **for** phase = 1 to f + 1 **do**

Vote

3: Broadcast value(x)

Propose

4: **if** some value(y) received at least n - f times **then**

5: Broadcast propose(y)

6: **end if**

7: **if** some propose(z) received more than f times **then**

8: x = z

9: **end if**

King

10: Let node v_i be the predefined king of this phase i

11: The king v_i broadcasts its current value w

12: **if** received strictly less than n - f propose(y) **then**

13: x = w

14: **end if**

15: **end for**

Claim. *King Algorithm fulfills the all-same validity.* (Validity described in this problem)

Proof. If all correct nodes start with the same value, all correct nodes propose it in Line 5. All correct nodes will receive at least n - f proposals, i.e., all correct nodes will stick with this value, and never change it to the king's value. This holds for all phases.

B. To prove that there exists a deterministic Byzantine Broadcast (BB) protocol satisfying validity and agreement if and only if there exists a deterministic Byzantine Agreement (BA) protocol satisfying validity and agreement, we'll need to demonstrate the equivalence of these two scenarios. This involves proving both directions of the statement.

Direction 1: BB Protocol implies BA Protocol

Assume that there exists a deterministic Byzantine Broadcast (BB) protocol satisfying validity and agreement. We want to show that this implies the existence of a deterministic Byzantine Agreement (BA) protocol satisfying validity and agreement.

1. Validity and Agreement in BB Protocol: Let's denote the sender as "S" and the honest receivers as "R_1","R_2",...,"R_n". In the BB protocol:

Agreement: All honest receivers (R_1 to R_n) receive the same message from the sender S (agreement).
Validity: If the sender S is honest and all honest receivers (R_1 to R_n) have the same input, they will receive the same message that is consistent with their input (validity).

2. Construction of BA Protocol: Now, let's construct a BA protocol based on the BB protocol:

Each node "N_i" will act as a sender in the BB protocol to broadcast its input to all other nodes. Each node "N_i" will also act as a receiver to determine the agreed-upon output based on the BB protocol's messages.

In other words, Each node "N_i" acts as both a sender and a receiver. As a sender, "N_i" follows the BB protocol to broadcast its input to all other nodes. As a receiver, "N_i" follows the procedure defined by the BB protocol to determine the agreed-upon output. If all honest nodes have the same input, they will all broadcast the same message according to the BB protocol. Honest nodes acting as receivers will receive the same message from other honest nodes and determine the agreed-upon output based on the BB protocol's messages.

3. Validity and Agreement in BA Protocol: Consider the following cases:

If all honest nodes have the same input, then in the BB protocol, they will all send the same message to all other nodes (consistent with their input), and this message will be received by all other honest nodes. If the sender in the BB protocol (S) is honest, then the honest nodes acting as receivers will receive the same message from S (consistent with S's input). Thus, in this constructed BA protocol, each node acts as both sender and receiver. If all honest nodes have the same input, they will broadcast the same message (consistent with their input) and receive the same message from other honest nodes. This satisfies both validity and agreement in the BA protocol.

Direction 2: BA Protocol implies BB Protocol

Assume that there exists a deterministic Byzantine Agreement (BA) protocol satisfying validity and agreement. We want to show that this implies the existence of a deterministic Byzantine Broadcast (BB) protocol satisfying validity and agreement.

1. Validity and Agreement in BA Protocol: Given the existence of a deterministic BA protocol satisfying validity and agreement:

Agreement: All honest nodes in the BA protocol agree on an output (agreement).

Validity: If all honest nodes have the same input, they will all output the same value that is consistent with their input (validity).

2. Construction of BB Protocol: Now, let's construct a BB protocol based on the BA protocol:

Each node "N_i" will act as a sender in the BA protocol to broadcast its input to all other nodes. Each node "N_i" will also act as a receiver to determine the agreed-upon output based on the BA protocol's output.

In other words, Each node "N_i" acts as both a sender and a receiver. As a sender, "N_i" follows the BA protocol to broadcast its input to all other nodes. As a receiver, "N_i" follows the procedure defined by the BA protocol to determine the agreed-upon output. If all honest nodes have the same input, they will all output the same value according to the BA protocol. Honest nodes acting as senders will broadcast the same output value to all other nodes based on the BA protocol's output. Honest nodes acting as receivers will receive the same output from other honest nodes and determine the agreed-upon output based on the BA protocol's output.

3. Validity and Agreement in BB Protocol: Consider the following cases:

If all honest nodes have the same input, then in the BA protocol, they will all output the same value that is consistent with their input. If the output of the BA protocol (the agreed-upon value) is honest, then the honest nodes acting as receivers in the BB protocol will all receive the same output (consistent with the agreed-upon value). Thus, in this constructed BB protocol, each node acts as both sender and receiver. If all honest nodes have the same input, they will output the same value (consistent with their input) and receive the same value from other honest nodes. This satisfies both validity and agreement in the BB protocol.

Thus, By proving both directions, we have established that there exists a deterministic Byzantine Broadcast (BB) protocol satisfying validity and agreement if and only if there exists a deterministic Byzantine Agreement (BA) protocol satisfying validity and agreement. This demonstrates the equivalence of the two scenarios. The key idea is to leverage the validity and agreement properties of the source protocol (BB or BA) to ensure the same properties in the constructed protocol.

Problem 3

This proof can be done by numerous point of views. Some are explained below and others are left to the reader.

1st View. The extension of the impossibility result from the specific case of n = 3 and f = 1 to the general case of $f \geq \frac{n}{3}$ requires showing that the same core principles and arguments used in the original proof for n = 3 and f = 1 still hold true when considering larger values of n and f while maintaining the essence of the original proof. To demonstrate how the proof reduces to the n = 3 and f = 1 case, let's summarize the

key elements of the original proof for n = 3 and f = 1, and then indicate how these elements extend to the general case of $f \geq \frac{n}{3}$.

Original Proof for n = 3 and f = 1:

Assumption: In the n = 3 and f = 1 case, there are three nodes, one of which is Byzantine (f = 1).

Core Argument: The Byzantine node can manipulate the Agreement process by sending conflicting messages to the two honest nodes, causing them to disagree and violate Agreement.

Essence of the Proof: The proof shows that in the absence of a PKI, Byzantine nodes can exploit the uncertainty of their own input to create disagreements among honest nodes, making it impossible to achieve Agreement.

Extension to General Case of $f \geq \frac{n}{3}$:

Assumption: In the general case, $f \geq \frac{n}{3}$, where f represents the number of Byzantine nodes.

Core Argument: The key principle remains the same: Byzantine nodes can manipulate the Agreement process by sending conflicting messages to honest nodes to create disagreements.

Essence of the Proof: Similar to the original proof, the argument highlights that Byzantine nodes have the power to influence Agreement by exploiting the uncertainty of their inputs to create disagreements. With more Byzantine nodes ($f \geq \frac{n}{3}$), their ability to manipulate Agreement becomes more pronounced, making consensus impossible.

In essence, the extension of the impossibility result from the n = 3 and f = 1 case to the general case of $f \geq \frac{n}{3}$ involves applying the same core principles of manipulation and disagreement by Byzantine nodes. The main difference is that as the number of Byzantine nodes increases ($f \geq \frac{n}{3}$), their ability to disrupt Agreement becomes stronger, rendering consensus unattainable without a PKI assumption.

2nd View. In the synchronous model, the impossibility result for Byzantine Broadcast (BB) with f n/3 and no Public Key Infrastructure (PKI) can be reduced to the n = 3 and f = 1 case using the following steps:

1) Assumption and Notations: Assume we have a synchronous network with n nodes and f Byzantine nodes, where $f \geq \frac{n}{3}$. As before, we assume that honest nodes behave correctly and Byzantine nodes can behave arbitrarily.

2) Impossibility for n = 3 and f = 1: Recall the original impossibility proof for the case of n = 3 and f = 1. In this case, it was demonstrated that Byzantine Broadcast cannot be achieved without a PKI.

3) Generalization to Arbitrary n and f with $f \geq \frac{n}{3}$: We will use a reduction argument to generalize the impossibility result to all n and f with $f \geq \frac{n}{3}$.

4) Divide and Conquer: Assume we have a network with n nodes and f Byzantine nodes, where $f \geq \frac{n}{3}$. Divide this network into smaller sub-networks, each consisting of three nodes: two honest nodes and one Byzantine node.

5) Reduce to n = 3 and f = 1 Case: For each sub-network, we know that the impossibility result holds for n = 3 and f = 1, as previously proven. We can think of the Byzantine nodes in these sub-networks as "super Byzantine nodes" that behave according to the original Byzantine node's strategy.

6) Agreement in Sub-Networks: Within each sub-network, the two honest nodes must agree on the message they receive, as per the impossibility result for n = 3 and f = 1.

7) No Agreement Across Sub-Networks: However, across different sub-networks, the "super Byzantine nodes" can create disagreements among honest nodes, preventing them from reaching consensus.

8) Conclusion: By reducing the larger network to a collection of sub-networks, we have shown that even in the case of $n > 3$ and $f > 1$, the impossibility result for n = 3 and f = 1 holds true. The impossibility of achieving Byzantine Broadcast without a PKI is maintained in the synchronous model for all n and f with $f \geq \frac{n}{3}$. Please note that while this outline provides a general approach to reducing the impossibility result to the n = 3 and f = 1 case, a rigorous formal proof would require detailed mathematical analysis and technical arguments to fully establish the result in the synchronous model.

3rd View. There are 2 possible problems that can arise when a node undergoes byzantine failure. The first problem is not sending a message at all. More specifically, within a distributed system, in order for n nodes to function properly while having f nodes suffering from byzantine failure, a consensus has to be reached with n - f messages.

The second problem is when a node in byzantine failure maliciously sends different messages. In an extreme scenario, let's say that among the n - f nodes that achieved quorum, f were sent by byzantine

failure. Even in this case, the system has to operate normally, and thus (n - f) - f messages must be greater than f messages(sent by nodes suffering from byzantine failure).

In order to resolve the two problems above, (n - f) - f > f. n > 3f, which means when there are f nodes that has a byzantine failure, there has to be more than 3f nodes in order for the system to be byzantine fault tolerant. The smallest n value here is 3f + 1. Thus, here in the impossibility proof we need to consider n $\leq$ 3f and the smallest n value here is 3f. Accordingly, it would be sufficient to do the proof for the f = 1 and n = 3 case and it automatically extends to other $f \geq \frac{n}{3}$ cases.

Problem 4

Problem 5

A. We will consider the scenario where protocols are implemented in polynomial time $p(n)$ and the adversary is also polynomially bounded by $p(n)$. We will show that the impossibility result for Byzantine Broadcast (BB) may not hold under these conditions. Remember that the time complexity is $O(p(n))$ and $O(q(n))$ for the honest nodes and the adversary, respectively where p and q are polynomials. Now we can assume that