Blockchain Foundations **NTUA, Spring 2026**

Theory Exercise 4 Solutions

Not due

Problem 1

In this problem, we will consider the Streamlet protocol run by n permissioned parties, with a tunable quorum size q for notarization.

1. Suppose we want to maximize the resilience of the protocol, i.e., maximize the number of adversarial parties f that can be tolerated such that the protocol is both safe and live. Derive the optimal quorum size q and show that the resulting optimal resilience is $f = n/3$.

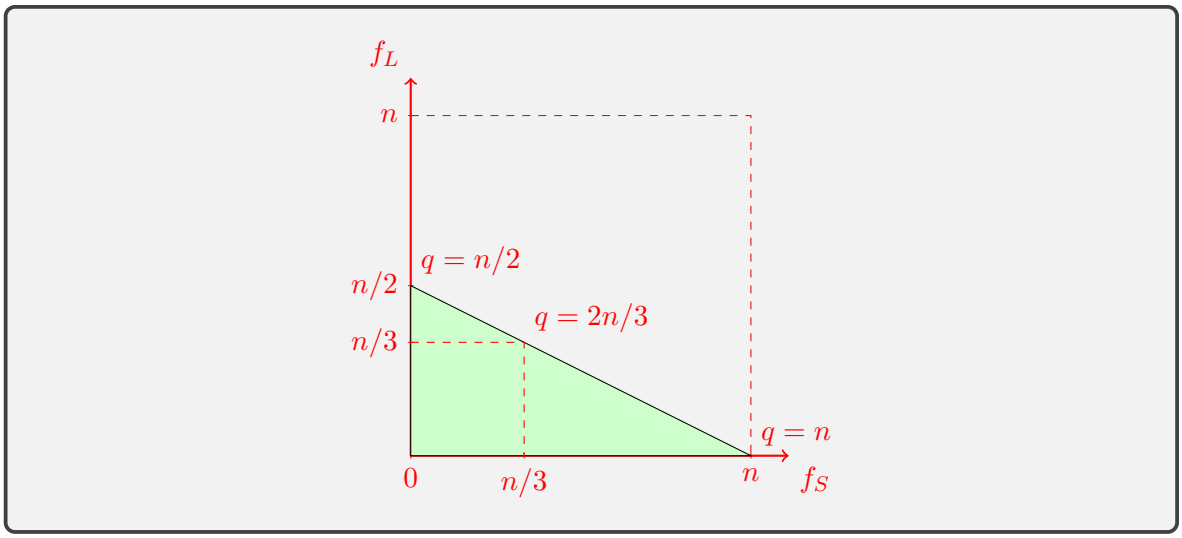
   For safety, the quorum size should be large enough so that the adversary cannot notarize two conflicting blocks. For this, we use a quorum intersection argument. If two conflicting blocks are notarized, they have received q votes each. At most f adversary parties may vote for both blocks. With n parties in total, for both blocks to be notarized, we require $2q - f \leq n$. To prevent double notarization, we require $2q > n + f$.
   For liveness, the quorum should be small enough so that the adversary cannot prevent notarization by not voting. So we require $q \leq n - f$.
   With both these requirements together,

   $$\frac{n+f}{2} < q \leq n - f.$$

   This implies that $\frac{n+f}{2} < n - f$, i.e. $f < n/3$. Thus the optimal resilience is $n/3$, and the resulting optimal quorum size is $q = 2n/3$.

2. Suppose you believe that an attack against safety is more likely than an attack against liveness (since a double-spend can provide significant rewards to the attacker). Hence, you want to tune Streamlet to increase the resilience against safety attacks, even at the expense of decreasing the resilience against a liveness attack. Can this be done? If so, exhibit and plot the tradeoff between the two resiliences. If not, explain why not.

   By increasing the quorum size further, we can improve the resilience against safety attacks. From the analysis in part (a), Streamlet is resilient against $f_S < 2q - n$ adversary parties for safety attacks. But then, the liveness resilience is $f_L \leq n - q$. Thus, we can tune the quorum size to increase safety resilience and decrease liveness resilience, subject to the constraint $f_S + 2f_L < n$. This tradeoff is shown in the plot below.



Problem 2

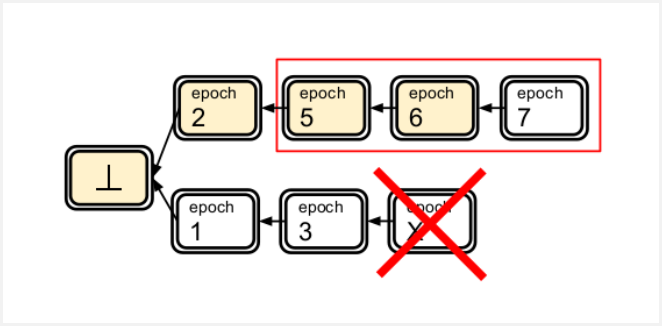
A consensus protocol is said to be t-safe if whenever the number of adversary parties is less than t, the protocol is safe. A consensus protocol is said to be t-accountable-safe if *whenever* safety is violated, at least t parties can be irrefutably proven to have violated the protocol.

1. Does t-safety imply t-accountable-safety? Does t-accountable-safety imply t-safety? Or neither property implies each other? Explain.

   Yes. t-accountable-safety implies that at least t parties have to violate the protocol in order that a safety violation to happen. This means that if there are less than t adversary parties, no safety violation will happen. Hence the protocol is t-safe.

2. In Lecture 18, we claimed that Streamlet (with quorum size $2n/3$) is $n/3$-accountable, where n is the total number of parties. But we have only gone through one possible safety violation scenario, not all. Complete the analysis by considering all safety violation scenarios.

   As in the security analysis for Streamlet, consider three notarized blocks from epochs $5, 6, 7$, at consecutive heights. This is when the block from epoch 6 will be confirmed. A safety violation on block 6 would mean that a conflicting block is notarized at the same height as block 6. See the figure below from the Streamlet paper for an example.



If the conflicting block is from epoch 6, then at least $n/3$ parties voted for both blocks in epoch 6, so they are accountable. (The parties who sent two different votes can be identified easily.) Similarly, the conflicting block could not have been notarized in epoch 5 or 7, because blocks 5 and 7 are already notarized.
Now suppose the conflicting block is from epoch $X < 5$. Then at least $2n/3$ parties voted for block X. This means that the honest voters observed a notarized chain (block 3 in this example) with the same length as block 5. These parties, if honest, should not vote for block 5 in epoch 5, since it does not extend their longest notarized chain. But block 5 has also received $2n/3$ votes. Thus by the quorum intersection argument, at least $n/3$ parties voted for both block X and block 5, so they are accountable.
Finally, suppose the conflicting block is from epoch $X > 7$. Since block 7 is notarized, at least $2n/3$ parties voted for block 7. This means that the honest voters observed block 6 as notarized. These parties, if honest, should not vote for block X since it does not extend their longest notarized chain (block 6 or longer). But block X has also received $2n/3$ votes. Thus by the quorum intersection argument, at least $n/3$ parties voted for both block 7 and block X, so they are accountable.
This analysis covers all possible safety violations, just as in the security analysis. This shows that Streamlet is $n/3$-accountable.

Problem 3

Does the safety of the Streamlet protocol depend on the Δ-synchrony assumption? Explain.

No. The safety analysis never assumes that a message broadcast has to be received at all parties within delay Δ. It only uses the fact that if a block is notarized at a particular epoch in the view of a party, then it will remain notarized in all future epochs.

Problem 4

Suppose all n parties running the Streamlet protocols are honest. Compute the average confirmation latency in terms of protocol parameters. Contrast this with the confirmation latency of the PoS longest chain protocol.

If all parties are honest, then every block proposed will be immediately notarized in the block's epoch. Then a transaction proposed in the beginning of epoch m will be confirmed by the end of epoch $m+1$, i.e. a latency of 4Δ.
For PoSLC, the latency is $O(k\Delta/f)$. Note that this latency does not improve even though all parties are honest. It is essentially a worst-case latency, depending only on the assumption of what is the worst-case fraction of adversary parties there can be in the system (which affects k and f.)