The Theory Behind Blockchains (Spring 19)
Problem Set 3

Submission: By 15:00 on May 8th, 2019. To be graded by Nathan Geier.

Instructions: You can collaborate with each other and consult external resources. However, you should write the solution on your own, and for each question, list all collaborators and external resources. Otherwise, write explicitly "None". Do not discuss solutions in the course forums; you are welcome to ask for clarifications if needed.

Total number of points in this problem set: 110.

1 Lightweight Clients (20 points)

Suppose Bob runs an ultra lightweight client which receives the current head of the blockchain from a trusted source. This client has limited memory and so it only permanently stores the most recent blockchain header (deleting any previous headers).

a. **(10 points)** If Alice wants to send a payment to Bob, what information should she include to prove that her payment to Bob has been included in the blockchain?

b. **(10 points)** Assume Alice's payment was included in a block k blocks before the current head and there are n transactions per block. Estimate how many bytes this proof will require in terms of n and k, assuming that the size of each transaction is 1 KB. Compute the proof size for $k = 8$ and $n = 256$.

2 Feather Forking (20 points)

In class we learned about feather forking: a coalition of miners controlling a fraction α of the total mining power attempts to censor transactions by announcing: "if we see a block containing a transaction from our blacklist B, we will attempt to fork until we are n blocks behind the main chain". Recall that in recitation 6 we computed the probability that the attack succeed for $n = 3$.

On expectation, for how long (expressed as a number of blocks found) will the system be in a forked state after the attack is launched before it either succeeds or fails (assuming $n = 3$)? Hint: You can solve this problem using a similar system of equations as done in the recitation, noting that every time a transition is taken, the fork lasts one additional block. As a sanity check, make sure that the attack is expected to resolve in 2 blocks for both $\alpha = 0$ and $\alpha = 1$.

3 Mining Pool Sabotage (20 points)

Recall that in recitation 6 we saw a sabotaging attack where a pool P_1 with mining power α_1 dedicates $\beta < \alpha_1$ of its power for sabotaging a pool P_2 with mining power α_2, and we saw that for some values of $\alpha_1, \alpha_2, \beta$ the attack is advantageous for P_1.

Suppose two pools, each with power α, sabotage each other with power $\beta < \alpha$. For what range of β will the two pools lose revenue by attacking each other? How much will they lose?

4 Shamir's Secret Sharing Scheme (15 points)

Let $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$ be shares of $x \in \mathbb{Z}_p$ and $y \in \mathbb{Z}_p$, respectively, under Shamir's t-out-of-n secret sharing scheme.

a. **(10 points)** Let $L : \mathbb{Z}_p^2 \to \mathbb{Z}_p$ be a linear function. Show that every party $i \in [n]$ can compute from (x_i, y_i) a new share z_i, so that $(z_1, \ldots, z_n)$ are shares of $L(x, y)$.

b. **(5 points)** Let $P(u, v) = u \cdot v$. Show that every party $i \in [n]$ can compute from (x_i, y_i) a new share z_i, so that $2t$ shares z_i are sufficient to reconstruct $P(x, y)$ assuming that $t < n/2$.

5 Analysis of Nakamoto's Consensus (35 points)

Recall Nakamoto's protocol in the ideal tree model from lecture 6. For a given execution of the protocol, we say that a chain ch is an honest chain in step r if it was the chain of some honest node in step r of the execution. We say that ch is an honest chain if it was an honest chain in some step of the execution. We say that two honest chains ch and ch' are consistent if they agree on all but their last n blocks.

a. **(10 points)** Assume that with overwhelming probability over an execution of the protocol, any two honest chains are consistent. Describe a policy for merchants who use the blockchain for transactions that would guarantee that with overwhelming probability merchants will not fall victim to a double spending attack. Explain why it works.

b. **(10 points)** Fix some specific execution and assume that any two honest chains ch_r and $chain_{r+t}$ in steps r and $r+t$, respectively, are consistent provided that $t \leq \Delta$, where Δ is the network's latency. Prove that any two honest chains in the execution are consistent.

c. **(15 points)** Consider an attacker that controls ρ of the nodes in the network. Further assume that due to controlling key points on the network, messages between the attacker and honest nodes (in either direction) are delivered right away with no delay, whereas messages between honest nodes always take three steps to reach one another. Show that the attacker can mine at least a $\frac{\rho}{1-\rho}$ fraction of any honest chain.