Programming Assignment in IN5420/IN9420

We have developed a simulator to simulate the use of selfish mining strategies in Bitcoin. The simulator implements the selfish mining strategy described in the paper[1] along with the basic building blocks of Bitcoin such as consensus, and communication.

The assignment:

1. Understand the code:
    a. consensus protocol implementation in consensus/customized/consensus.go
    b. pow implementation in consensus/customized/pow.go
    c. network dissemination in communication/customized/network.go
2. Experiment with different parameters in the configuration file and observe the behavior of selfish mining.
    a. Parameters of interest are number of nodes (2 to 100), mining power of the selfish node, mining powers of the honest nodes, block interarrival times, and propagation delay.
    b. While this is a less influential parameter, you can also vary the number of blocks (500 to 10000).
    c. For a large number of nodes, it is advisable to keep the logging level at WARN or ERROR. However, for 2-3 nodes and a smallish number of blocks, one can set the logging level to INFO or even DEBUG, in order to understand what is going on.
    d. Output metrics - number of blocks produced vs expected.
3. Present the results in graphical plots
4. Identify when the selfish mining strategy is useful for a miner based on the analysis
5. Understand the differences between the implementation of selfish mining in consensus/customized/consensus.go and the paper
6. Optional: change the code so as to make the implementation closer to that in the paper, run the simulator and analyze the results

The deliverables:

A document (pdf) including the following:

1. An overview of the differences between the code and the paper
2. Visual plots presenting the experimental results
3. Analysis of the results and the resulting conclusions regarding selfish mining

Submission Deadline: March 21, 2023 before 24:00
How to submit: email to the TA (email address: praveema@ifi.uio.no)

How to get the simulator?

The TA will send it to your email address. Please do not distribute it publicly.

How to run the simulator?

- run the main() method

Usage:

```
go run main.go [configFile]
```

Args:

configFile: The path to the configuration file which contains the configuration for the simulation. If not present, a default configuration file “config.toml” is used.

The configuration parameters are documented in config.toml.

References:

1. Eyal, I., & Sirer, E. G. (2018). Majority is not enough: Bitcoin mining is vulnerable. *Communications of the ACM*, *61*(7), 95-102.