P2P Systems - Final Project

Master Degree in Computer Science,
Computer Science and Networking
Business Informatics
Academic Year 2016/2017

TinyCoin:
Simulating fraudolent mining strategies in
a simplified Bitcoin Network

1. Goal of the Project

The goal of the project is the evaluation of the Bitcoin mining process, through its simulation in *TinyCoin*, a simplified version of Bitcoin. In particular, it is required to consider one or more fraudolent versions of the mining process and to compare them with the standard mining process/between themselves.

In *TinyCoin*, each user has a single address and the transactions are simplified with respect to Bitcoin as follows:

- *TinyCoin* defines an account-based system to record the unspent amount of each node, i.e. the amount of a node is paired with the node itself and is not spread in the blockchain, like in *Bitcoin*.
- a transaction has a single input, the reference to the user's account, and a single output address, which refers the target account.
- we assume that only the rightfull owner can create transactions spending his/her own funds. This hypothesis enables to omit the implementation of the digital signature of the transaction.
- transactions do not include neither *scripts* nor *fees*

Network nodes may be *miners* or *normal nodes*. There are four different tyes of miners, characterized by different computational powers: CPU, GPU, FPGA or ASIC nodes. The distribution of the different types of miners and their computing power must be properly chosen.

TinyCoin includes a centralized oracle that chooses one of the miners of the network as the one which has mined the new block to be added to the blockchain. The probability that a certain miner is chosen by the oracle at each step depends solely on its computational power. Since the miners do not execute the Proof of Work, they do not insert the nonce in the block and the blockchain is a simple chain of blocks, not tamper free. The Bitcoin header of each block is replaced by an identifier which uniquely identifies that block.

When a miner has been chosen by the oracle, it cannot modify the mined block any more.

The oracle performs its choice of the next miner at regular intervals of time whose frequency is defined by a probability distribution. Mined block are propagated on the P2P network like in *Bitcoin*, i.e. when a node receives a block it propagates it to all its neighbours.

When a miner is chosen by the oracle, it makes some strategic decisions about the mined block, among others:

- *choosing between blocks at the same height.* If two different blocks are mined and announced at almost the same time, it results in a one block fork, with either blocks admissible under the longest valid chain policy. Miners then have to decide which block to extend. The default behavior is to build on top of the block that they heard about first.

- *when to announce new blocks.* When a new block is mined, miners have to decide when to announce this block to the Bitcoin network. The default behavior is to announce it immediately, but they can choose to wait some time before announcing it. This behaviour is referred as *selfish mining* in [2]

The strategy chosen by the miner has an impact on the overall behaviour of the network.

The goal of this project is to evaluate at least the selfish mining strategy defined in [2] in *TinyCoin*. This strategy has to be evaluated by considering different parameters:

- the number of selfish miners: only one/a percentage/all selfish miners
- the computational power of selfish miners with respect to the computational power of honest miners
- the interval of time between the generation of two consecutive blocks
- the latencies between the nodes of the TinyCoin network.

Evaluate how these choices affect the behaviour of the overall network, in particular, the number of forks occurring in the blockchain and the time required to solve them. Consider and evaluate other fraudolent mining strategies (optional).

2. Implementation

The simulation can be implemented through the Peersim simulator [1]. The Peersim nodes correspond to the clients of the Bitcoin network and it is possible

to assume that a Bitcoin address equals the Peersim identifier of a node. It is more advisable to exploit the event driven version of Peersim which enables to simulate different network latencies between the nodes.

All the nodes (clients) of *TinyCoin* are present when the simulation starts, i.e. dynamic joins of new nodes must not be considered. Furthermore, the topology of the overlay which contains the interconnections between the nodes is generated at random, at the start of the simulation, provided that the resulting network results connected (it is possible to exploit Peersim overlay generation facilities). Furthermore, each node has initially a given amount of bitcoins generated at random.

To simulate the generation of the transactions each node, at each round of the protocol, decides at random whether to generate a transaction and, in this case, the target of the transaction and the amount to be spent are generated at random as well. As far as concerns the amount to be spent, it must be less than the total amount owned by a node.

The flooding of a transactions and of blocks generated by the miners must be simulated by the Peersim mechanisms.

The expected value and the variance of the different simulation parameters are fixed at the start of the simulation.

3. Project Submission Rules

The project must be developed individually. The material to be submitted for the evaluation is the following one:

- a report (pdf document) describing the main features of the project. The report should include:
  - a brief report of the project choices (mining strategies chosen, configuration of the parameters,...) and of the implementation.
  - a set of plots reporting the metrics evaluated in the experiments
- a pdf document reporting the code of all the JAVA classes defined to set up the simulation.

The report and the code must be submitted both electronically, through the Moodle, and at the reception desk of the Department of Computer Science.
The project must be submitted a week before the date of the oral examination (if required). The discussion of the project consists in the presentation of a short demo, which can be run on the personal laptop, a general discussion of the choices made in the implementation of the system, and a general discussion of the Bitcoin system. The oral examination will regard a review of the topics presented in the course. I recall that the oral examination is waived for the the students who have passed the Mid and Final Term.

Do not hesitate to contact us by e-mail
(laura.ricci@unipi.it, emanuele.carlini@gmail.com, d.difrancesco@for.unipi.it)
or during the question time, Thursday 15.00 PM-18.00 PM.

References

[1] *The Peersim Simulator* http://peersim.sourceforge.net/.

[2] Ittay Eyal and Emin Gun Sirer *Majority is not Enough: Bitcoin Mining is Vulnerable* Financial Cryptography and Data Security, 18th International Conference, Lecture Notes in Computer Science 8437, Springer.