Trading Network Performance for Cash in the Bitcoin Blockchain - University of Tromsø

Longitudinal study on the Bitcoin blockchain from 2013 to 2017. In our thesis we mainly focus on three major problems on the Bitcoin blockchain: 1-Scalability. 2-Performance. 3-Fees and Tolls. We make our assumptions and test the results by analyzing the real blockchain with a data analytics system created for that purpose, BAS (Blockchain Analytics System), then we propose approaches that might be followed in order to get more performance in the Bitcoin network by optimizing the amount of fee paid from users. We also consider miner's revenue and give advide about the right creation time in order to optimize miner's profit, according to the mining hardware he is usign. We finally discuss whether is good to increase or not the block size to increase system performance.

Getting Started

```
\thesis: contains the thesis in .pdf format
\BAS: Blockchain Analytics System folder
\BAS\dataframe: data frame D generated for the analysis
\BAS\info: contains the info.txt file with information about data retrieved
\BAS\plot: plots generated with data retrieved
\BAS\src: source code containing main.py, data_manipulation.py, plotting.py and retrieval.py
```

Prerequisites

Some libraries used for the computations might have to be installed, such as numpy, pandas or matplotlib.

```
pip install numpy
pip install matplotlib
pip install pandas
```

Usage

Usage of the blockchain alaytics system:

```
observ.py -d number
    -h | --help : usage
    -i          : gives info of the blockchain stored in /info
    -p          : plot data
    -t number   : get the amount of unconfirmed transactions for <number> minutes
    -d j        : retrieve information about transactions and save them in a Panda DataSet, having a
                jump of j blocks with a default number of blocks retrieved b = 10
```

Example of use: use -d command for retrieval and set an initial jump J = 10

```
python main.py -d 10
```

Note: this jump will remain of 10 even if in the later analysis the variable is changed

once D is created data can be plotted

```
python main.py -p
```

Note: to have a nice plotting is suggested to have downloaded at least few months of activity in the blockchain

```
reward_fee.png                  : (date, BTC) plot the revenue from the block
reward R compared to the fee from users M
profit_multiple_miners.png      : (creation time, profit) plot the profit using
AntMinerS9 having 1, 50, 100, 500 miners in the mining pool.
profit_creation_time.png        : (creation time, profit) plot the revenue, costs
and profit for miners according the creation time
total_btc.png                   : (date, BTC) total bitcoin in circulation
fee_input_miners.png            : (miners, fee%) comparison between the percentage
of fee paid by the 20 biggest mining pools
fee_latency.png                 : (fee, latency) plot the transaction fee in
relation with the fee latency
txs_fee_distribution.png        : (date, %) plot the transaction fee distribution,
divided in category
txs_feedensity_distribution.png : (date, %) plot the transaction fee density
distribution, divided in category
fee_latency_years.png           : (fee, latency) plot the relation between the
transaction fee and the latency, distributed during years
blocksize_latency.png           : (block size, latency) plot the block size Q in
relation with the transaction latency
throughput.png                  : (date, throughput) plot throughput during time
creation_time_miners.png        : (creation time, blocks mined) bar plot of
occasional miners and mining pools about the creation time
block_size.png                  : (date, block size) plot the block size during
time
top_miners_monthly.png          : (date, blocks) plot the occasional miners and the
mining pools every months according to how many blocks they mine
trendy_miners.png               : (date, transactions) plot the transactions
approved by the 15 major miners during the years
number_of_miners.png            : (date, miners) plot the number of active miners
in the network
```

Built With

- [Python] :v2.7.12
- [PyCharm] :v2017.1.4