Πετράκης Κωνσταντίνος
Γιώργος Ηλιόπουλος

A. Shanti Bruyn

Blockchain: An Introduction

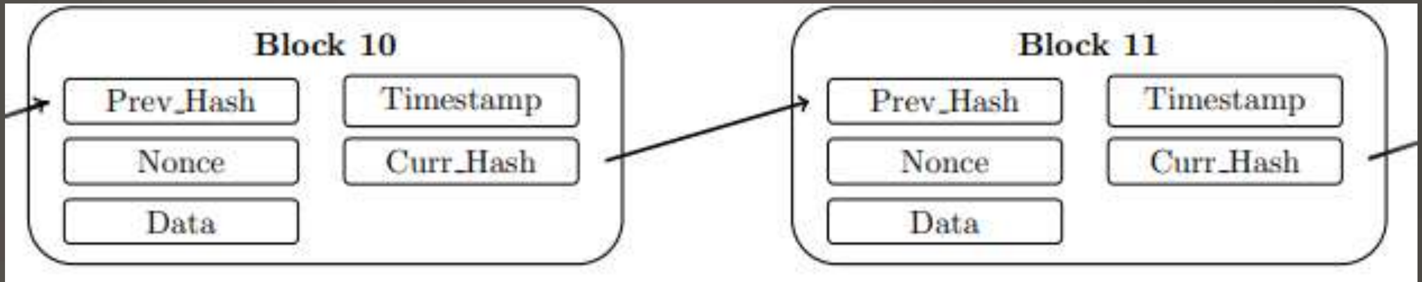
Overview

- What is blockchain?
- How does it work?
- How do the different variables within blockchain work together?

Blockchain

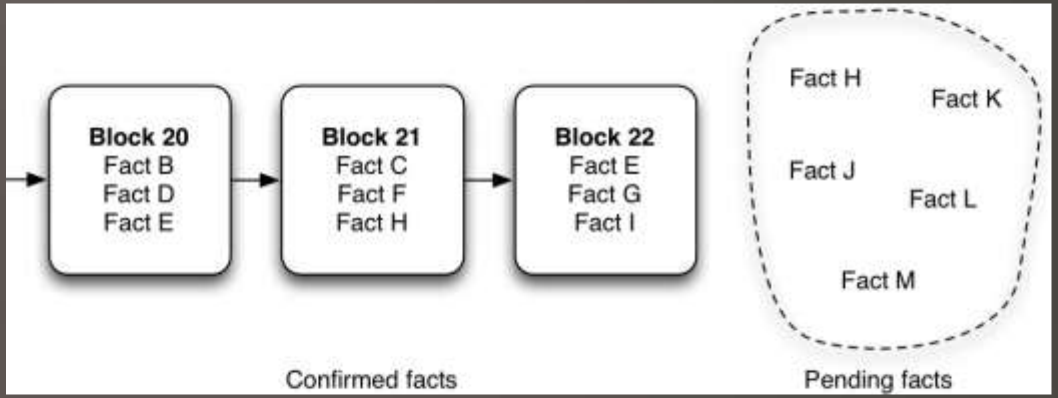
- Blockchain is a new type of a decentralized, distributed database
  - Solves the previously unsovable double spending problem without a middleman.

Building Blocks of Blockchain

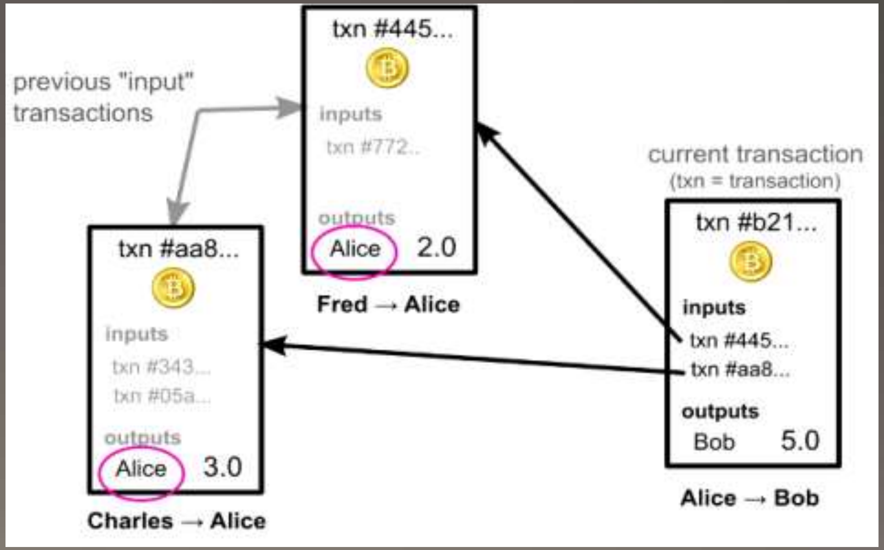


- The Database
- A Block
  - A block number
  - Hash of the previous block
  - Nonce
  - Data: the transactions
  - Timestamp with the time the block is created
  - Hash of the current block
- The Hash (the ‘proof-of-work’)
  - A random number is guessed (‘Nonce’)
  - The Nonce is added at the end of all the data in the block
  - This is all hashed to SHA256 method
  - If the hash starts with a predetermined number of zero’s a new block is found. Else the miner starts again guessing another Nonce.

Building Blocks of Blockchain

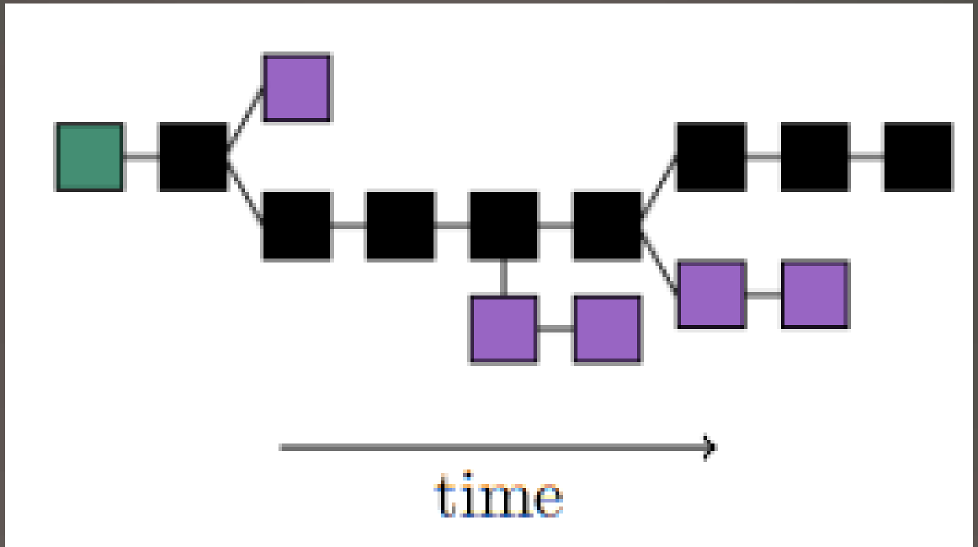


- A miner/node
  - New transactions are broadcast to all nodes
  - Each node collects new transactions into a block
  - Each node works on finding a difficult proof-of-work for its block.
  - When a node finds a proof-of-work, it broadcasts the block to all nodes
  - Nodes accept the block only if all transactions in it are valid and not already spent.
  - Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash
- A transaction (valuta or complete programmable e.g. Ethereum, voting rights)



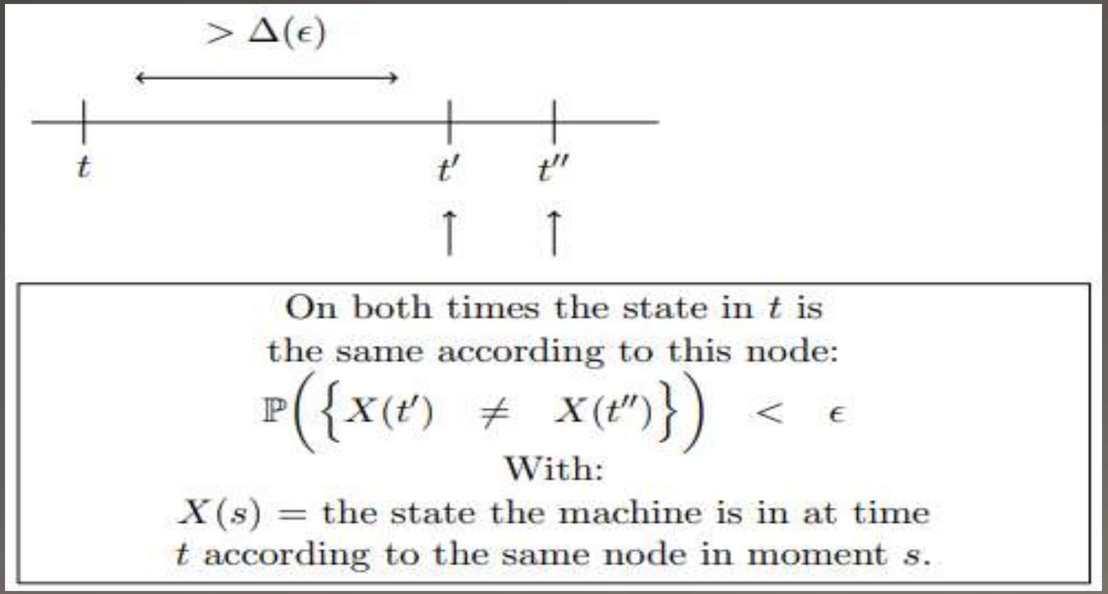
Building Block of Blockchain

- A fork
- Blockchain Safety
- Incentive



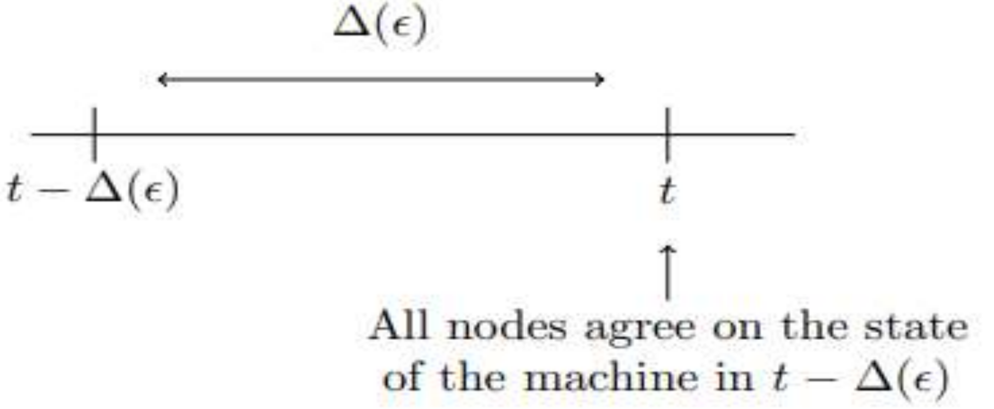
Nakamoto consensus Mathematically

- Termination: There exists a time difference function $\Delta(.)$ such that, given a time t and a value $0 < \epsilon < 1$, the probability is smaller than ϵ that at times $t', t'' > t + \Delta(\epsilon)$ a node returns two different states for the machine at time t.



Nakamoto consensus Mathematically

- Agreement: There exists a time difference function Δ(.) such that, given a $0 < \epsilon < 1$, the probability that at time t two nodes return different states for $t - \Delta(\epsilon)$ is smaller than ϵ.



$Y(k)$ = state machine at time $t - \Delta(\epsilon)$ according to node k, at current time t.

$$\mathbb{P}\left(\left\{Y(k) \neq Y(j) \mid k \neq j\right\}\right) < \epsilon$$

Nakamoto consensus Mathematically

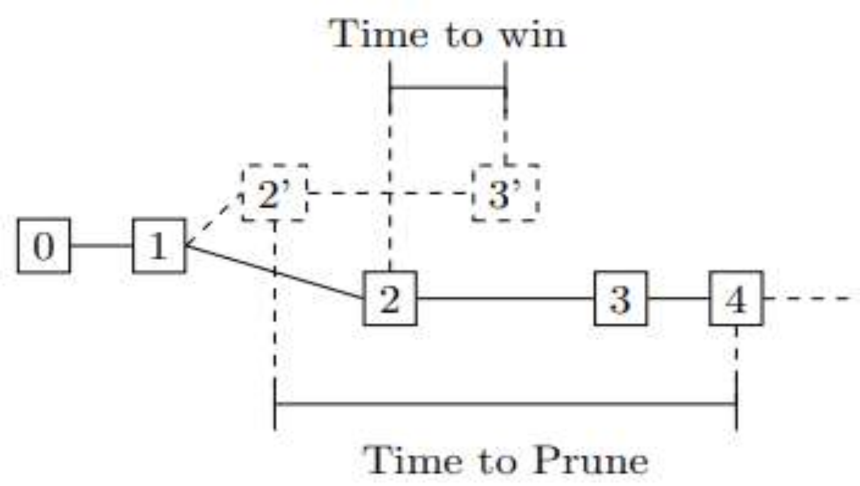
- Validity: If the fraction of mining power by Byzantine nodes is bounded by f, i.e. $\forall t\colon \frac{\sum_{b \in B(t)} m(b)}{\sum_{n \in N} m(n)} < f$, then the average fraction of state machine transitions that are not inputs of honest nodes is smaller than f. (m(i)=mining power of node i)
- At any time t, a subset of nodes $\mathrm{B(t)} \subset N$ are Byzantine.

Indicators/Metrics

- Consensus Delay: the time it takes a system to reach agreement
  - (ϵ, δ) consesnsus delay = δ% of the time ϵ % of the nodes agree on the state (ϵ, δ) seconds ago.
  - e.g. (95%, 90%) consensus delay = 10 seconds means: 90% of the time, 95% of the nodes agree on the state of the machine 10 seconds ago.
- Fairness: optimally the largest miner and the non-largest miners' representation in the transitions set should be the same as their respective mining powers.
  - 1. $\frac{transactions\ not\ coming\ from\ largest\ miner}{all\ transactions}$
  - 2. $\frac{mining\ power\ not\ owned\ by\ the\ largest\ miner}{all\ mining\ power}$
  - Fairness = $\frac{1.}{2.}$ optimally fairness = 1.0

Indicators/Metrics

- Mining power utilization = $\frac{mining\ power\ that\ secures\ the\ system}{total\ mining\ power}$
- Time to Prune: This implies what time a user has to wait to be confident a transaction has occured.
  - δ time to prune =$\delta\ percentile\ \left(\begin{matrix} time\ node\ learns\ this\ transaction \\ has\ never\ taken\ place \end{matrix} - time\ a\ node\ learns\ about\ a\ transaction\right)$
- Time to Win: Average time wasted due to forks
  - $\delta\ time\ to\ win =$ $\delta\ percentile\left(\begin{matrix} last\ time\ a\ (different)\ node \\ dissagrees \end{matrix} - \begin{matrix} the\ first\ time\ a\ node\ believes\ a\ never-to-be-pruned \\ transition\ has\ occured \end{matrix}\right)$



How everything is connected

formula

$$\underset{(\text{MB})}{\text{blocksize}} = \underset{(\text{bytes})}{\text{headersize}} + \underset{(\text{bytes})}{\text{transaction size}} \times \#\ \text{transactions in a block}$$

$$\underset{(\#\ \text{hashes / second})}{\text{total mining power}} = \underset{(\#\ \text{hashes / second})}{\sum_{\text{nodes} \in \text{system}} m(i)} \quad \text{with } m(i) = \text{mining power of node } i$$

$$\underset{(\text{seconds})}{\text{inter nodes times}} = \frac{\text{blocksize}}{b(i,j)} \qquad \frac{(MB)}{(MB/\text{second})} \quad \text{with } b(i,j) = \text{bandwidth between nodes } i \text{ and } j,\ i \neq j$$

$$\underset{(\text{seconds})}{\text{system width}} = \max \left\{ \underset{(\text{seconds})}{\text{inter nodes times}} \right\}$$

$$\underset{(\text{blocks/minute})}{\text{blockfrequency}} = \frac{\text{total mining power}}{\text{difficulty cryptopuzzel}} \qquad \frac{(\#\ \text{hashes / second})}{(\text{expected}\ \#\ \text{hashes needed})}$$

$$\#\ \text{transactions per second} = \underset{(\#\ \text{blocks / minute})}{\text{blockfrequency}} \times \#\ \text{transactions per block}$$

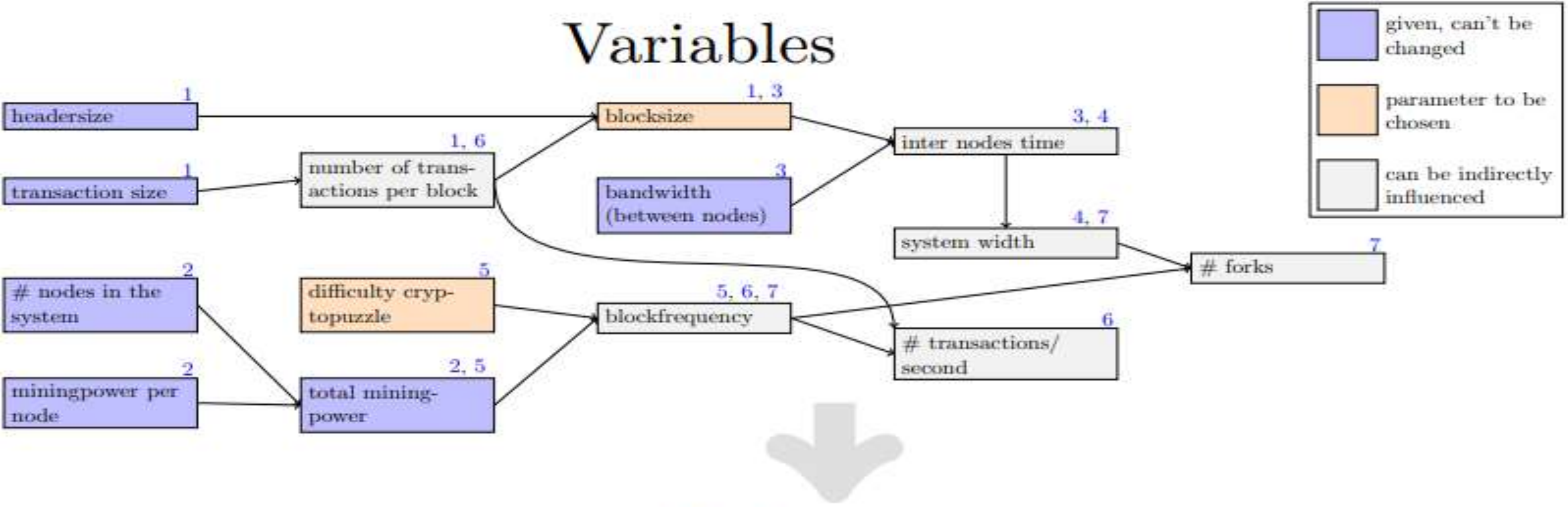
$$\underset{(\in \{0,1\})}{\mathbb{P}(\text{fork})} = 1 - (1+\text{blockfrequency} \times \text{system width}) \times e^{-\text{blockfrequency} \times \text{system width}}$$

P(fork) calculation

$$\begin{aligned}
\mathbb{P}(\text{fork}) &= \mathbb{P}\Big(N(X + \underbrace{\Delta t}_{\substack{\geq \text{ system} \\ \text{width}}}) - N(X) > 1\Big) \\
&= 1 - e^{-\text{blockfrequency} \times \text{system width}} \\
&\qquad - (\text{blockfrequency} \times \text{system width}) \\
&\qquad\quad \times e^{-\text{blockfrequency} \times \text{system width}} \\
&= 1 - (1 + \text{blockfrequency} \times \text{system width}) \\
&\qquad \times e^{-\text{blockfrequency} \times \text{system width}}
\end{aligned}$$

with $N(X) = \#$ blocks found in time $(0, X)$

$$N(X) \sim \text{Poisson}\left(\frac{1}{\text{blockfrequency}}\right)$$



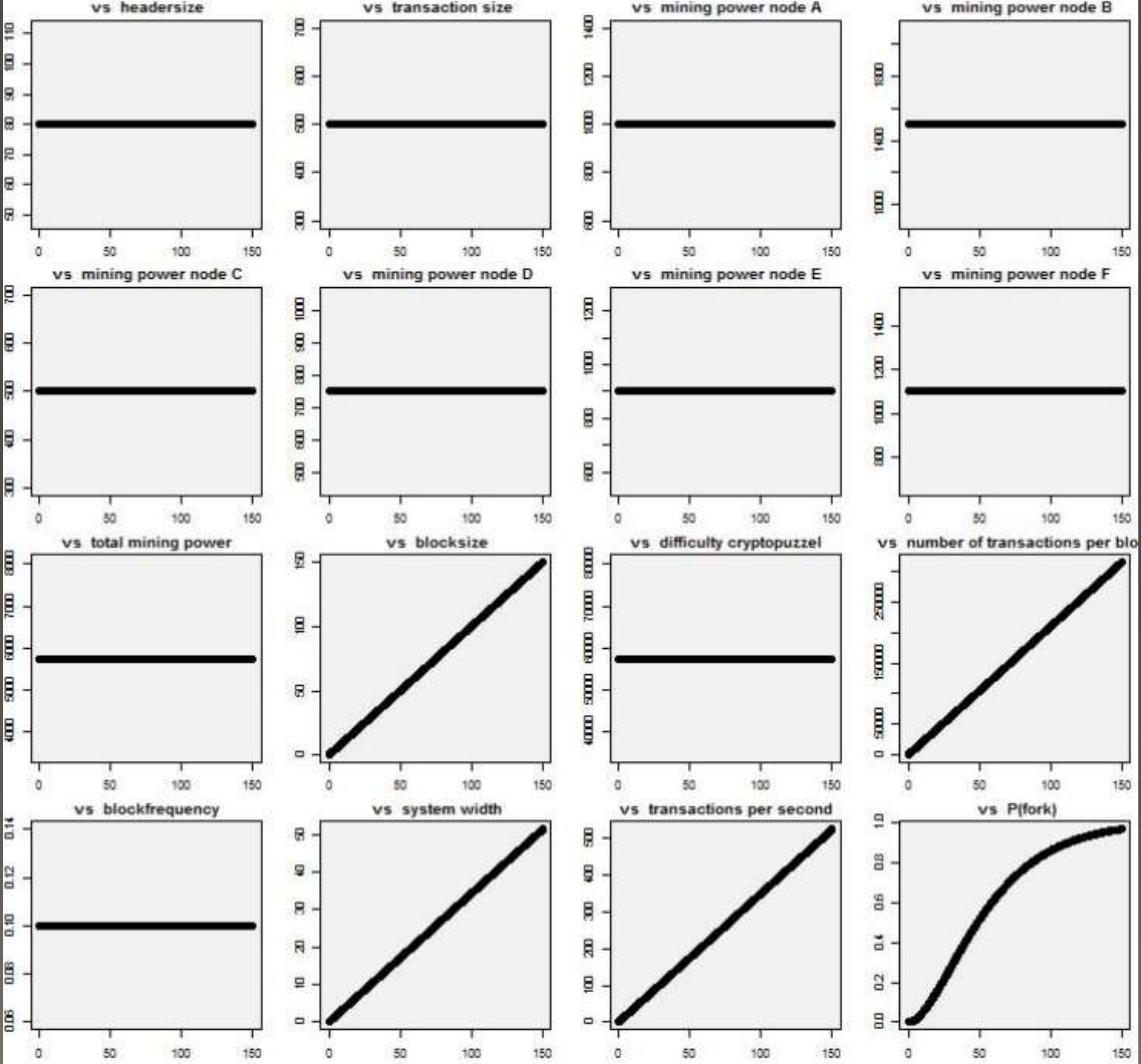
Indicators

	link with security		link with latency		
	Fairness	Mining Power Utilization	Consensus Delay	Time to Prune	Time to Win
The idea behind the indicator	A fair system should not allow the largest miner to get disproportionate power	The higher the mining power utilization, the more secure the system	A measurement for the time it takes the system to agree	Average time it takes till a fork is solved	Average time wasted due to forks
Some (but possibly not all) of the above variables associated with this indicator	• # nodes in the system • miningpower per node	• miningpower per node • total mining power	• system width • # forks	• blockfrequency • # forks	• mininpower per node • # forks

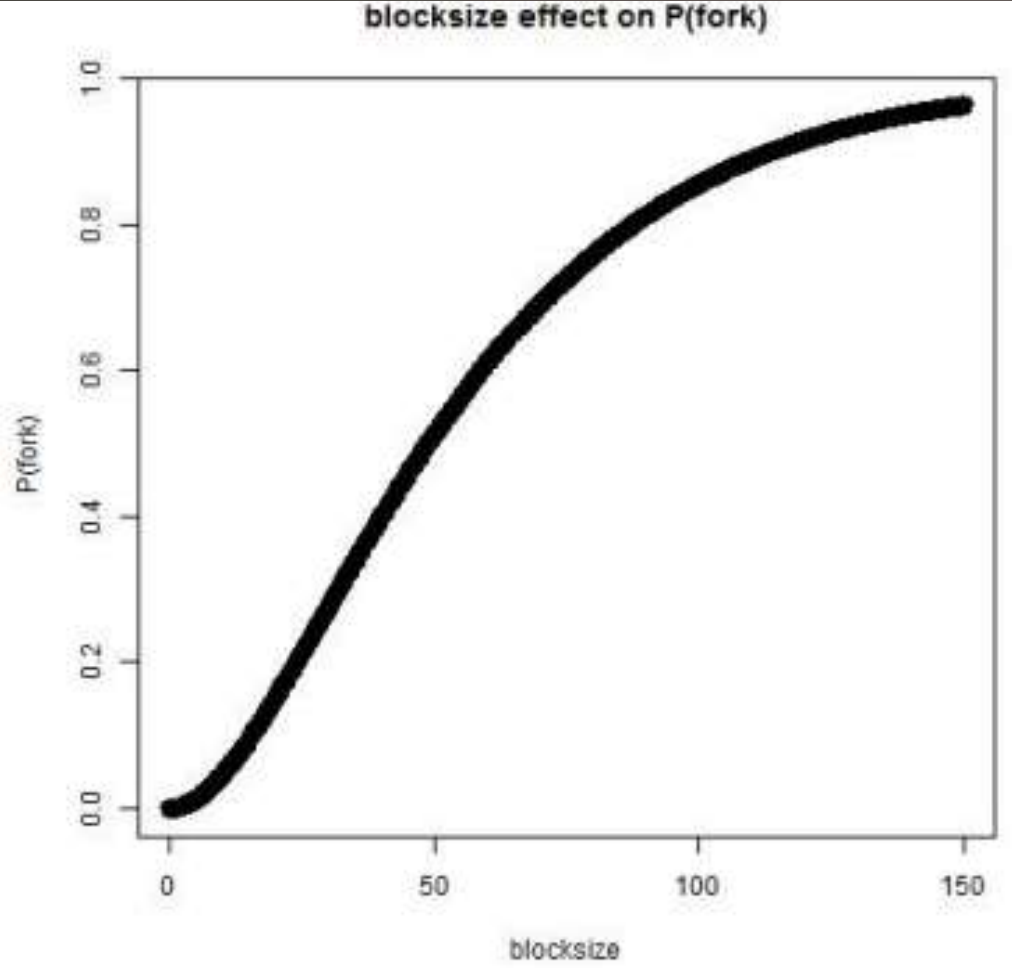
Relation between variables

- Blocksize
- Headersize
- Transaction size
- Mining power node A
- Difficulty Cryptopuzzle

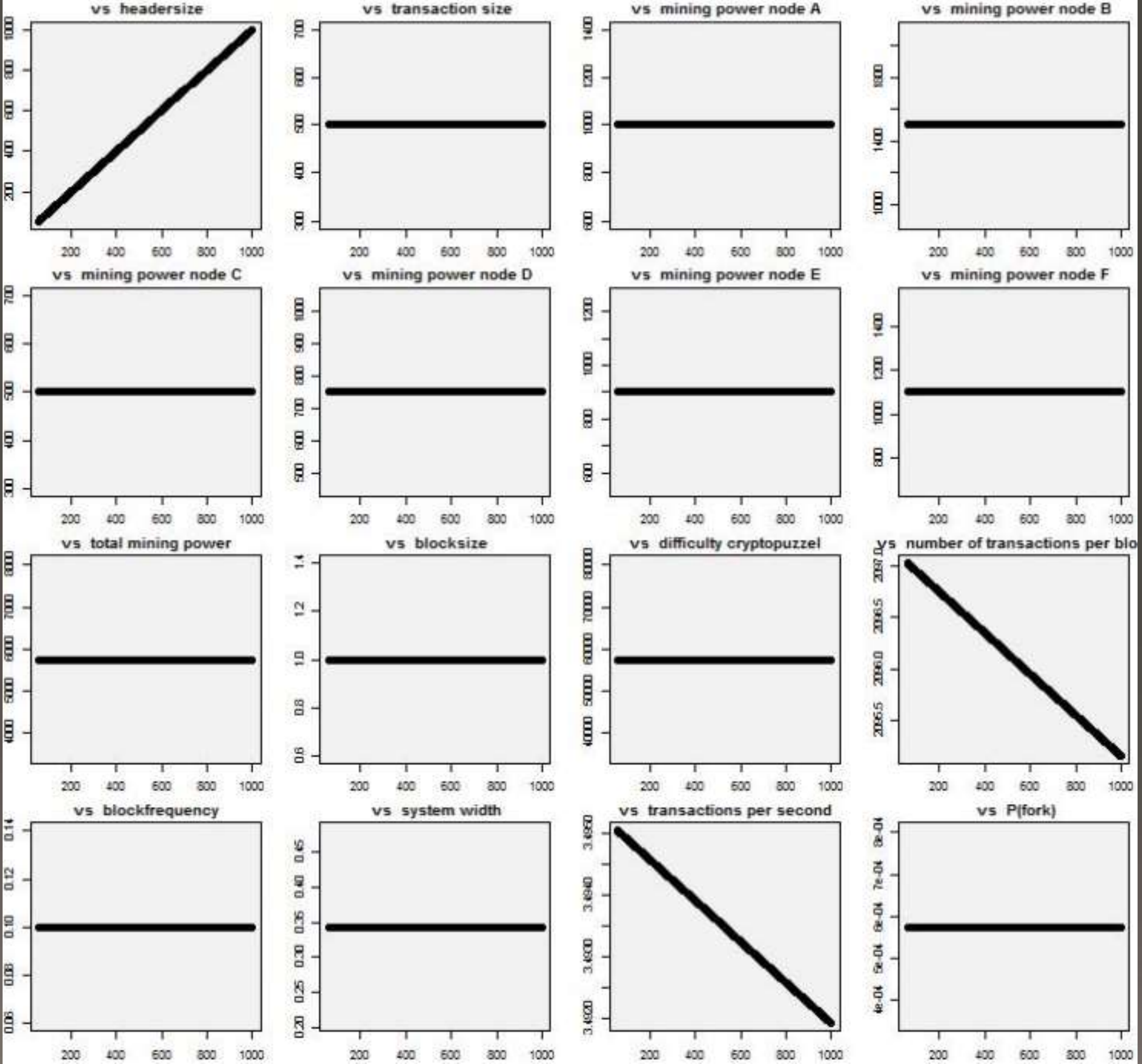
Changing Blocksize



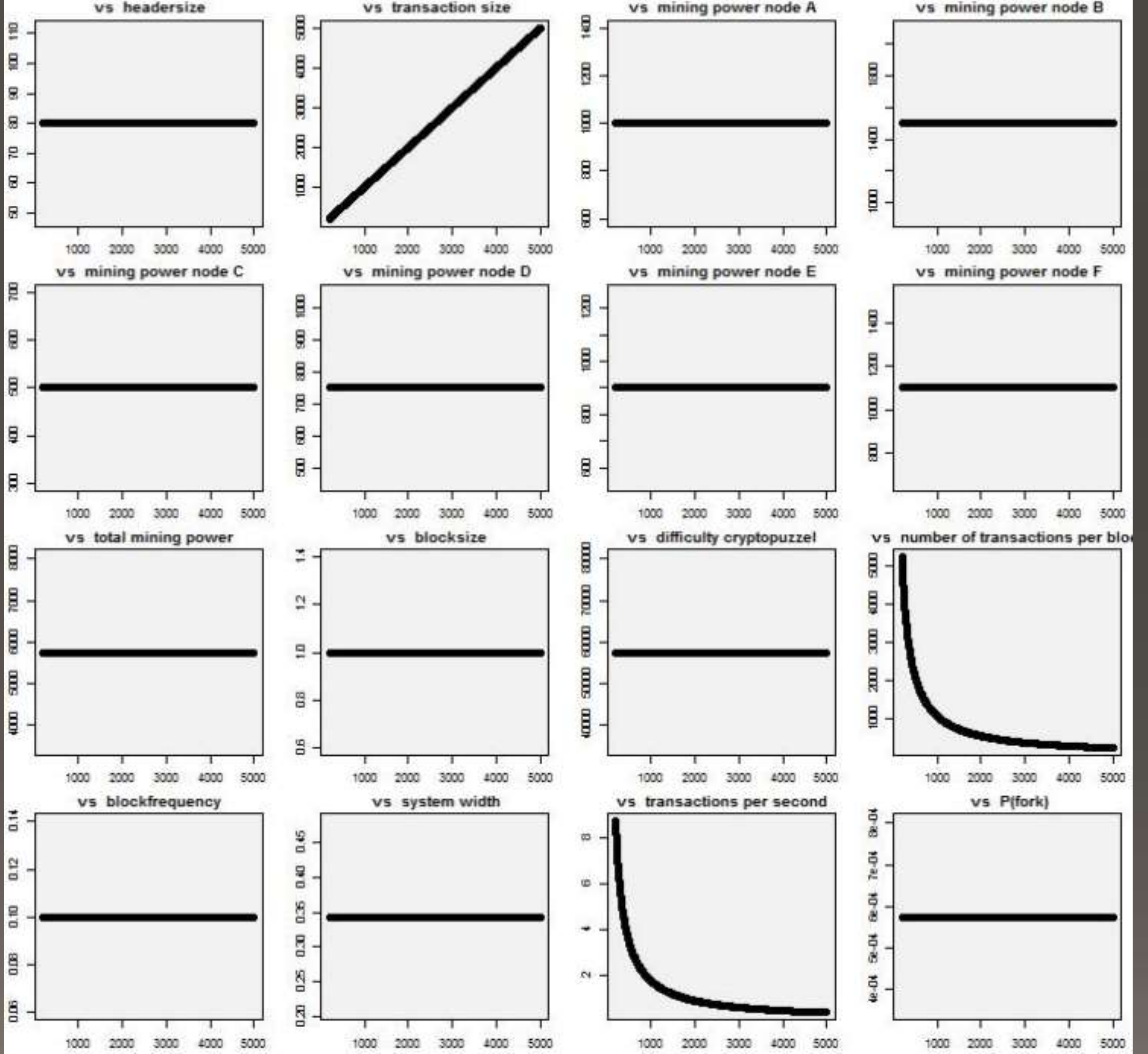
Blocksize on P(fork)



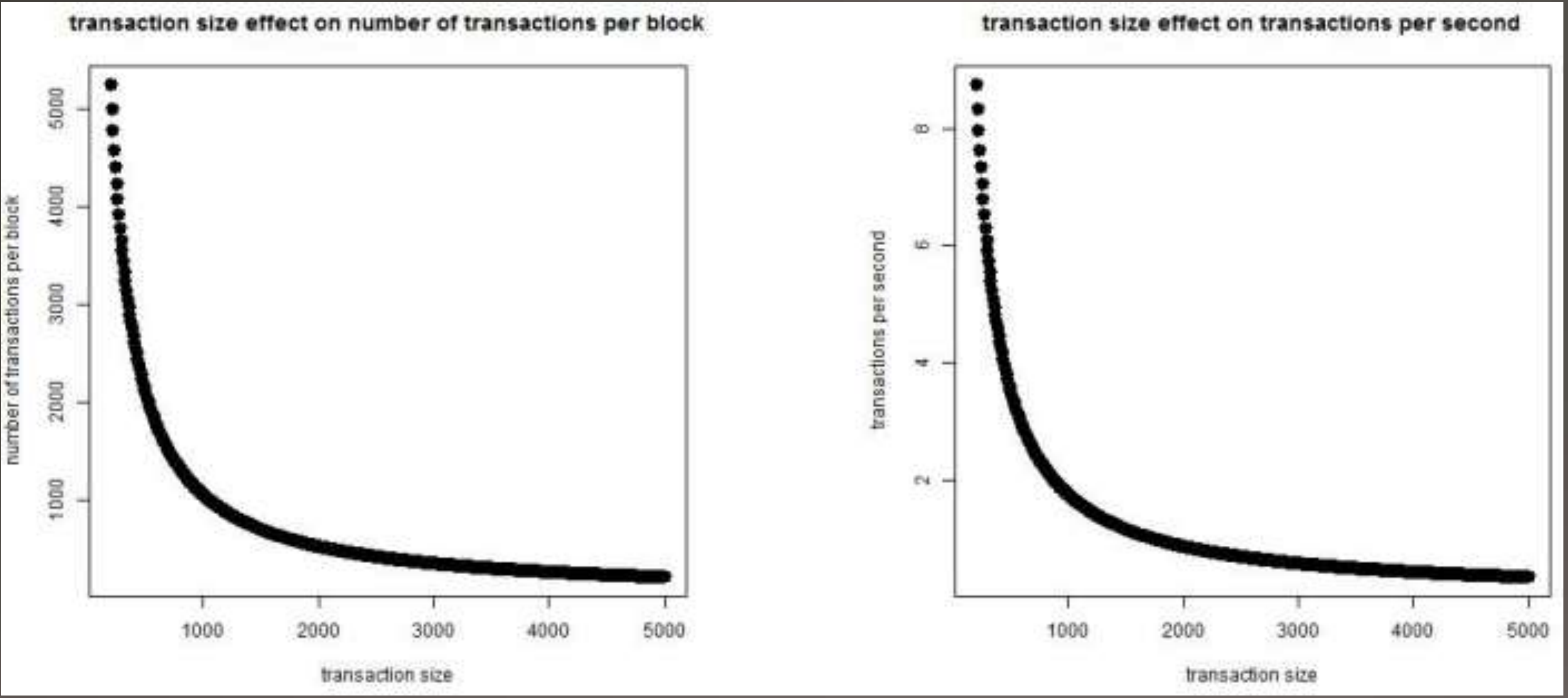
Changing Header size



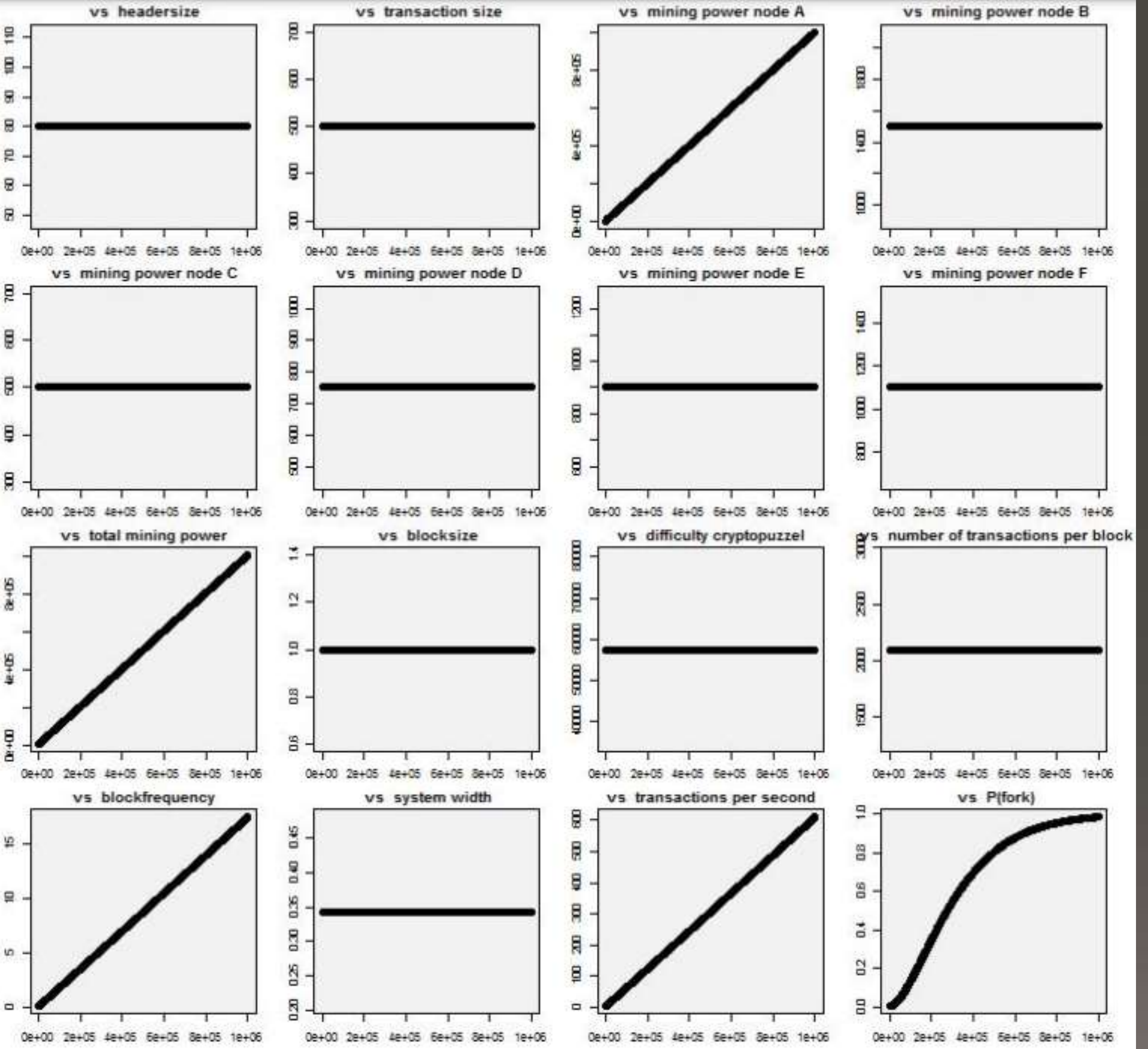
Changing Transaction size



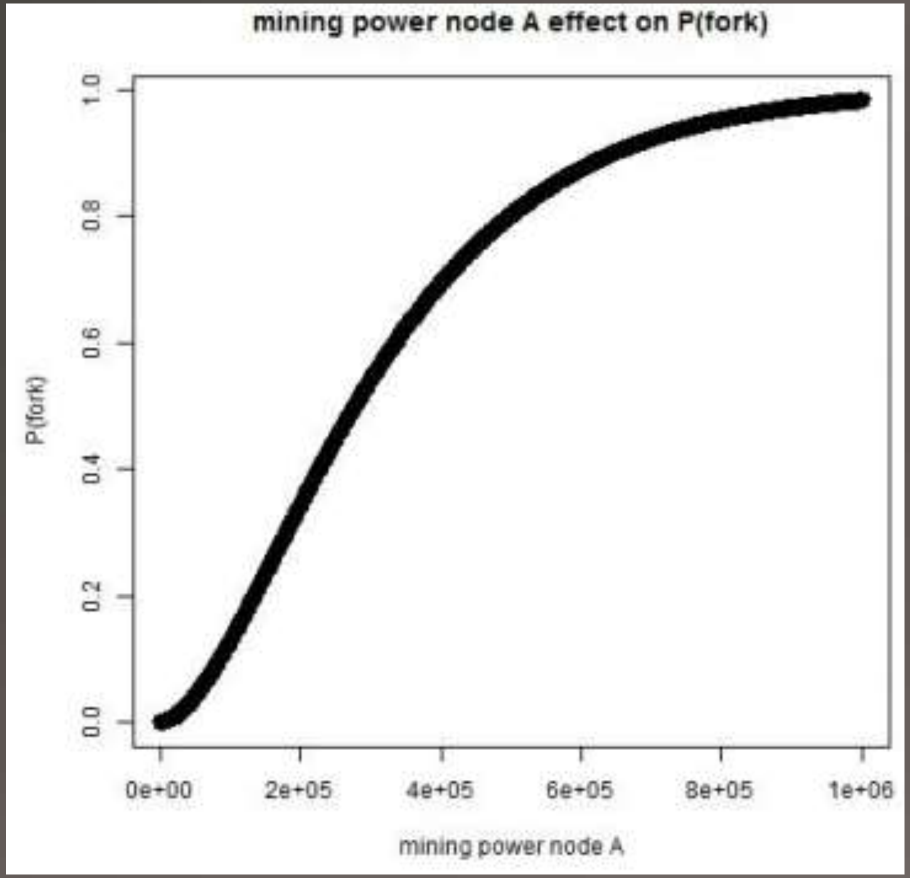
Transaction size effects



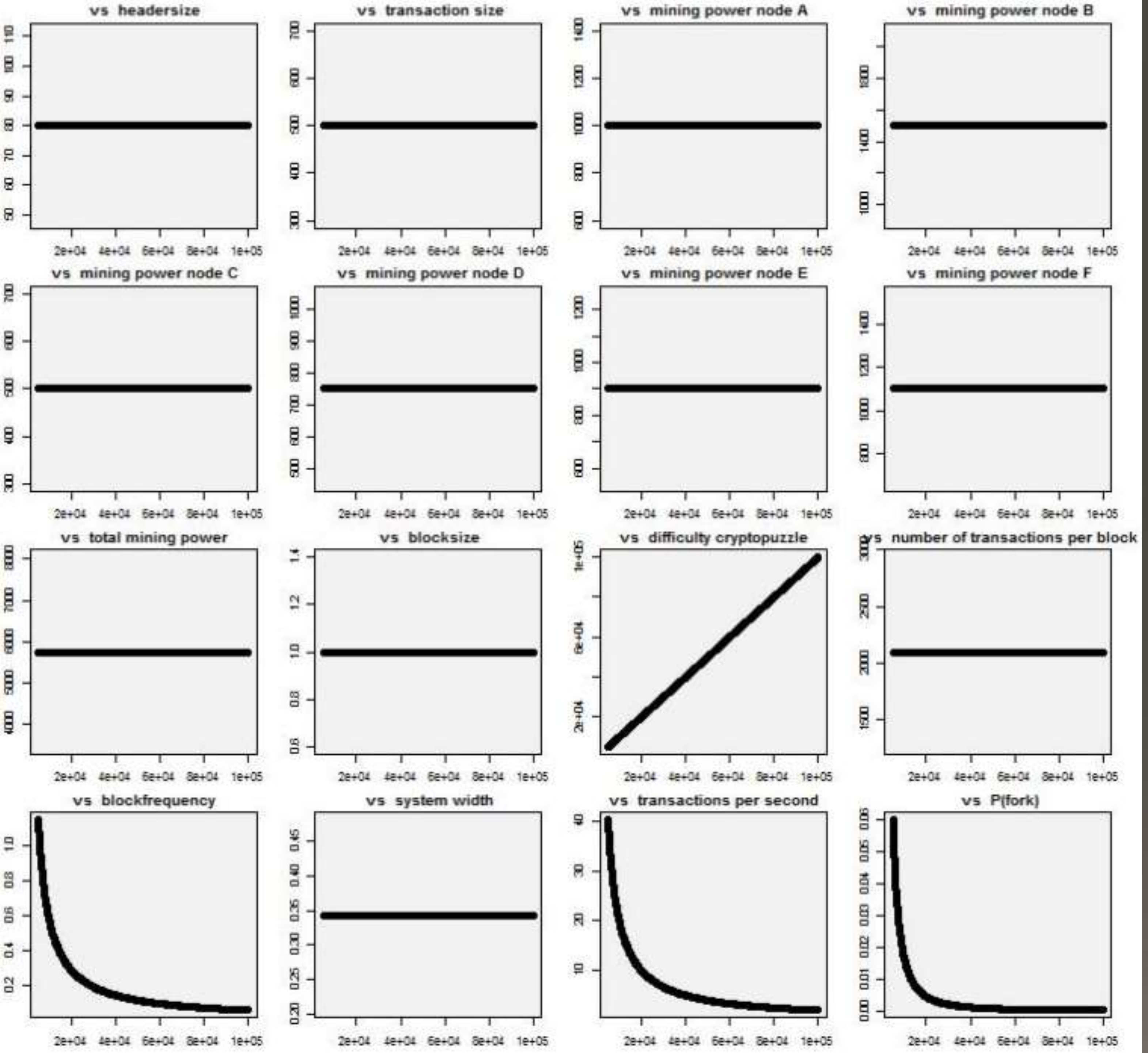
Changing Mining Power node A



Mining power effect



Changing Difficulty of Cryptopuzzle



Situation Today

- Block size still an issue.
- Proof of Stake cryptocurrencies (e.g. Ethereum 2.0, Cardano)
- Blockchain for Social Networks and lots of other applications (e.g. Internet Computer)

Thank You!