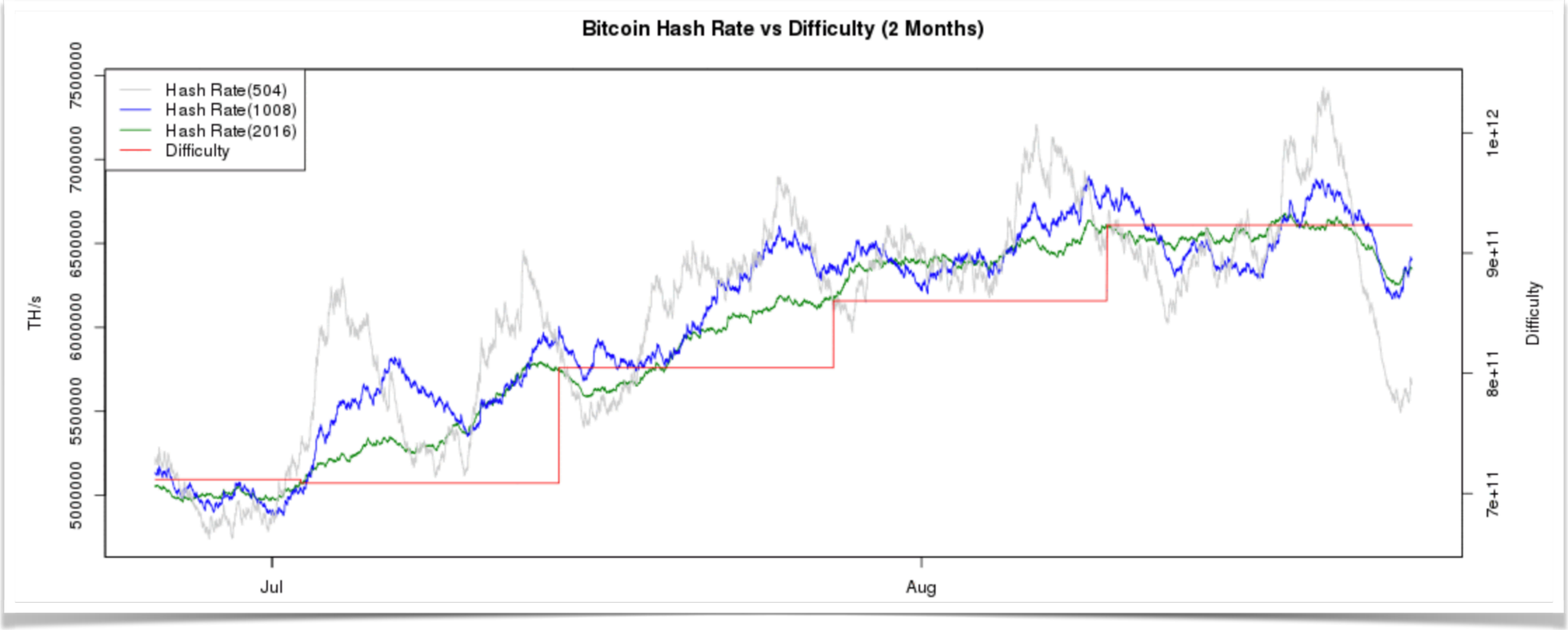
REVISITING DIFFICULTY CONTROL FOR BLOCKCHAIN SYSTEMS

DMITRY MESHKOV
Ergo Platform, IOHK Research

ALEXANDER CHEPURNOY
Ergo Platform, IOHK Research

MARC JANSEN
University of Applied Sciences Ruhr West

BITCOIN DIFFICULTY RECALCULATION



- Recalculates every 2016 blocks (epoch).
- Assumes that hash rate is constant.

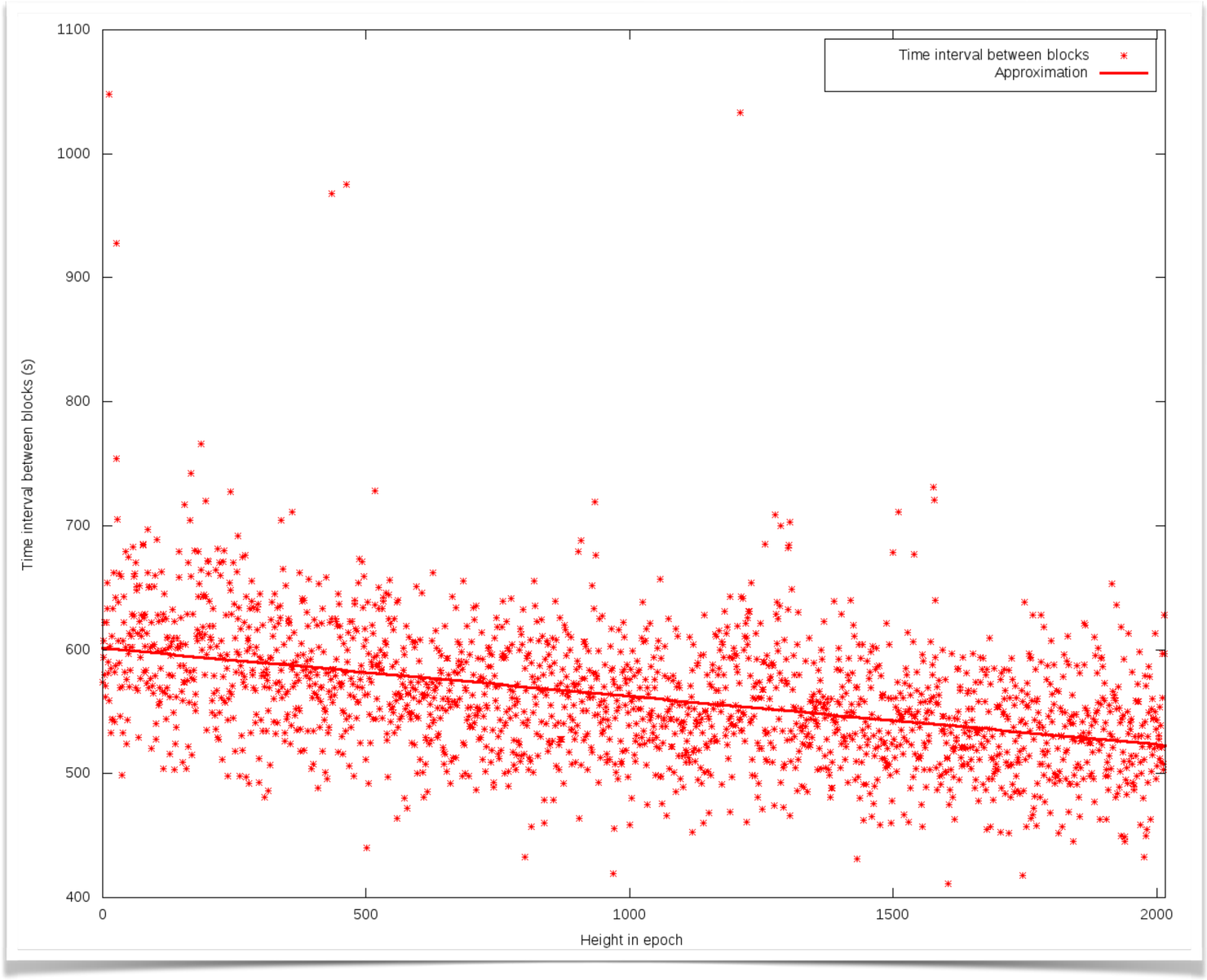
Graph from https://bitcoinwisdom.com/bitcoin/difficulty

BITCOIN

DIFFICULTY RECALCULATION

- Mean time
  9 min 20 seconds < 10 min.

- Time interval at the end of the epoch is less than 9 minutes.



PRIOR WORK

01

Research paper on comprehensive analysis of difficulty control. [1]

02

Better difficulty recalculation algorithm.

03

Assumes exponential hash rate growth in a deterministic way.

04

Difficult to implement in fixed-point or integer arithmetic.

05

The adversary might manipulate difficulty.

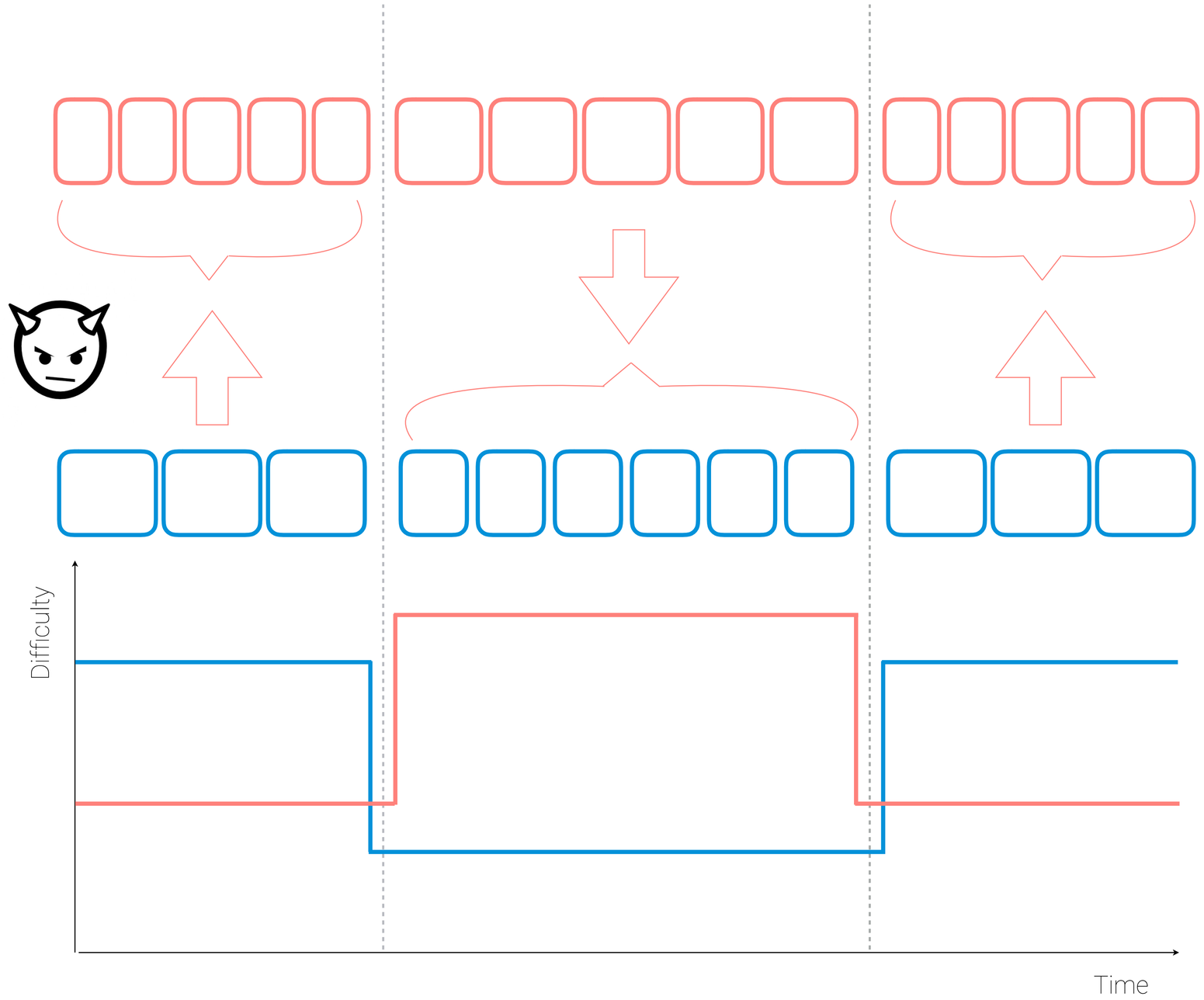
06

Attack on Bitcoin based on difficulty enables attacker to discard n-depth block, for any n and hash rate with probability 1 if he is willing to wait long enough. [2]

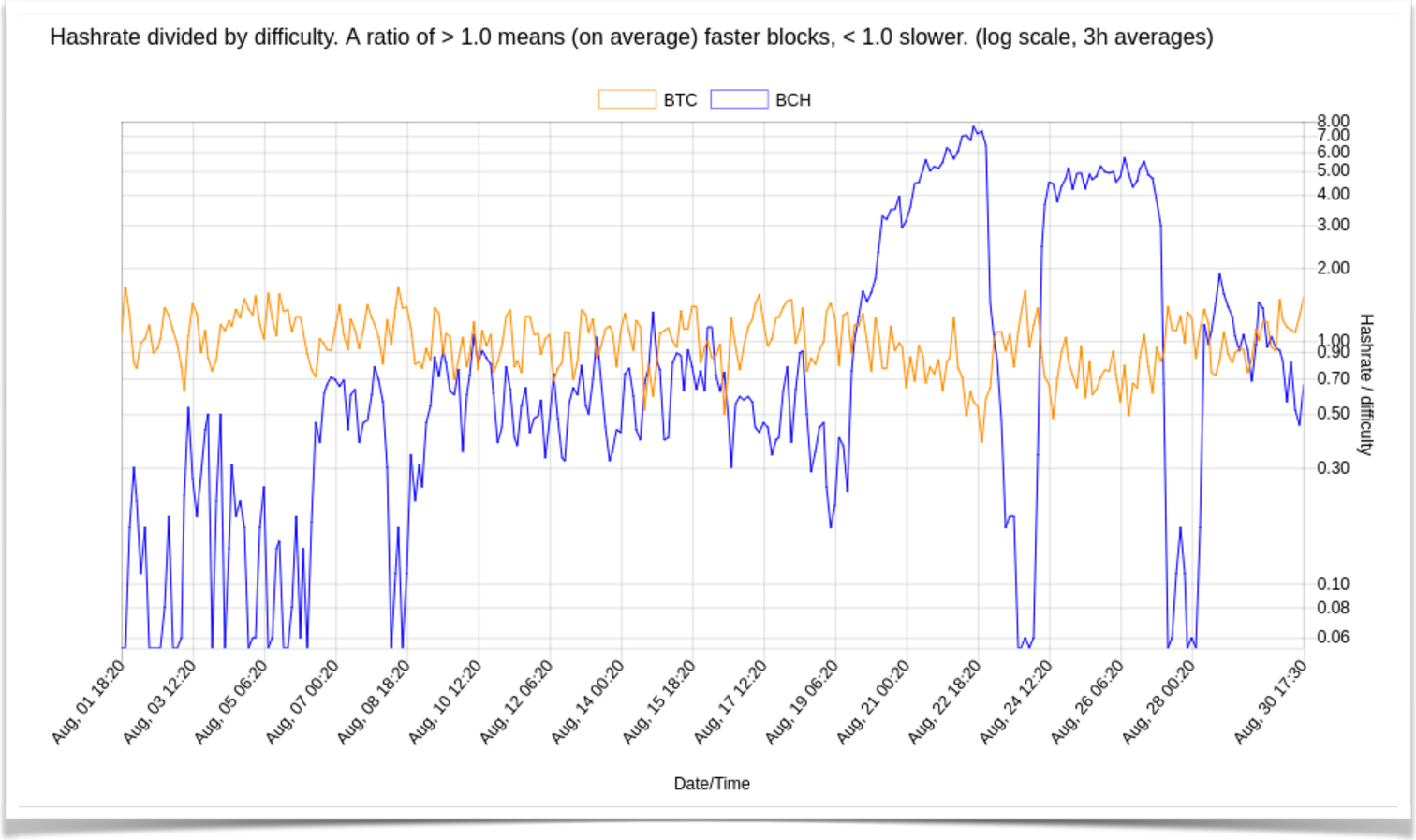
1. Kraft D. Difficulty control for blockchain-based consensus systems. 2016
2. Bahack L. Theoretical Bitcoin Attacks with less than Half of the Computational Power. 2013

COIN-HOPPING ATTACK

COIN-HOPPING ATTACK



EXAMPLE:
BTC/BCH FORK



Graph from http://fork.lol/pow/speed

COIN-HOPPING

Attacker profit:

$$P^2/(1+P)$$

Time interval between blocks:

$$T(1 + P^2/2(1 + P))$$

where

p — ratio of attacker hashrate to honest hashrate
T — desired time interval between blocks

IMPROVED DIFFICULTY ADJUSTMENT

DIFFICULTY UPDATE ALGORITHM

Properties of an ideal difficulty update algorithm:

01 IT SHOULD BE RESISTANT TO KNOWN TYPES OF ATTACKS BASED ON DIFFICULTY MANIPULATION.

02 IT SHOULD LEAD TO DESIRED BLOCK RATE FOR RANDOM FLUCTUATIONS IN THE NETWORK HASHRATE.

03 (OPTIONAL) SHOULD BE CALCULATED IN INTEGER ARITHMETIC.

LINEAR REGRESSION

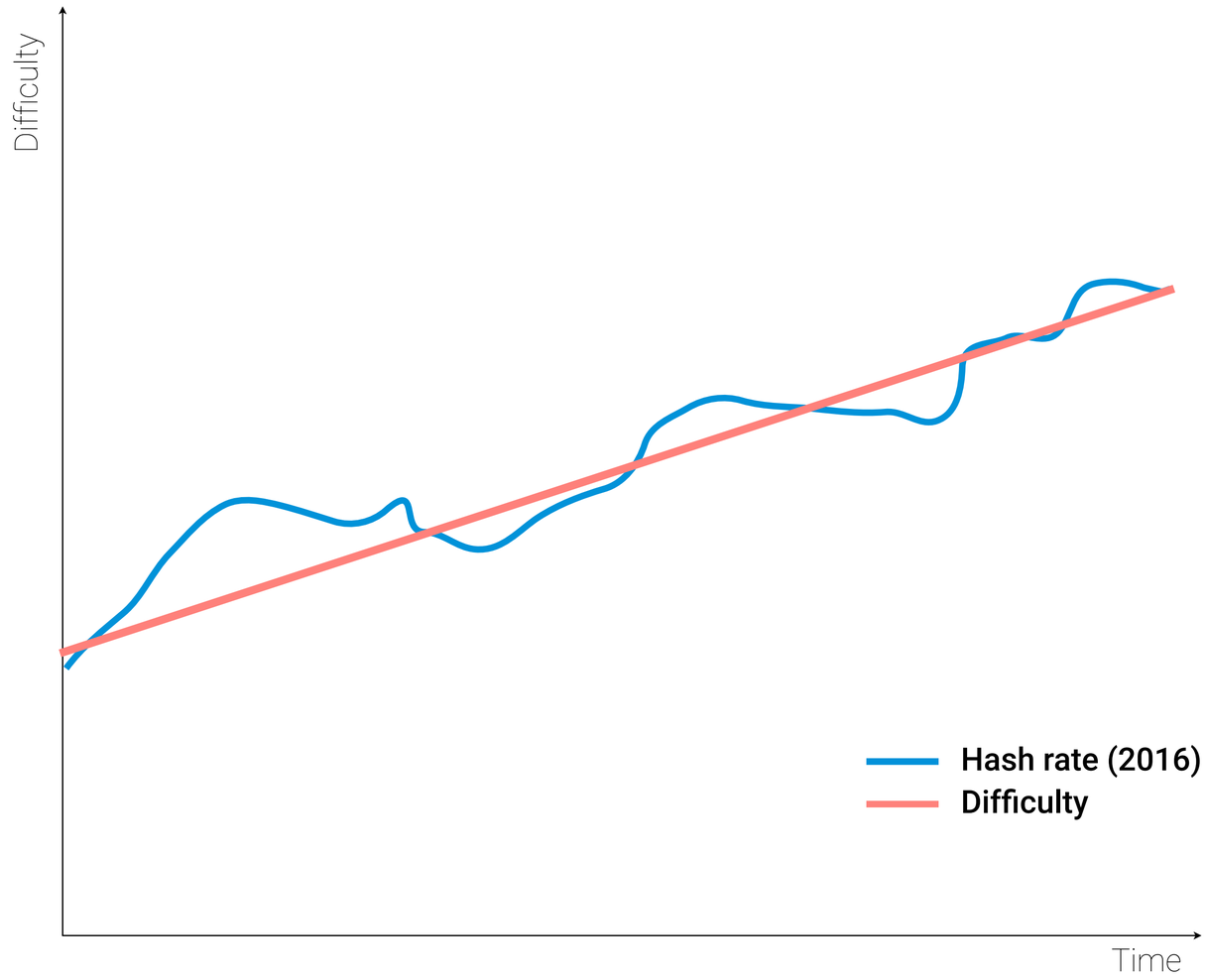
- Linear least squares method:

$$B = \overline{Y} - K\overline{X}$$

$$K = \frac{\overline{XY} - \overline{X}\,\overline{Y}}{\overline{X^2} - \overline{X}^2}$$

- Use multiple epochs.

- Take into consideration hashrate growth rate.



LINEAR REGRESSION

Ideal for linear hashrate changes (including constant)

Exponential 10% hashrate growth:
9,1 % error for Bitcoin algorithm
1,9 % error for our algorithm

Coin-hopping attack (p=20%):
1.7% error for Bitcoin algorithm
0,8 % error for our algorithm

Real Bitcoin data:
12.3% error for Bitcoin algorithm
8,4 % error for our algorithm

DMITRY MESHKOV

Founder of ERGO Platform, researcher at IOHK.

dmiry.meshkov@iohk.io
https://twitter.com/DmitryMeshkov