The Blockchain folk theorem

Bruno Biais (HEC & TSE), Christophe Bisière (TSE), Matthieu Bouvard (Mc Gill & TSE), Catherine Casamatta (TSE)

April 2018

What's a blockchain?

Distributed ledger, records transactions and ownership, operated within a peer to peer network

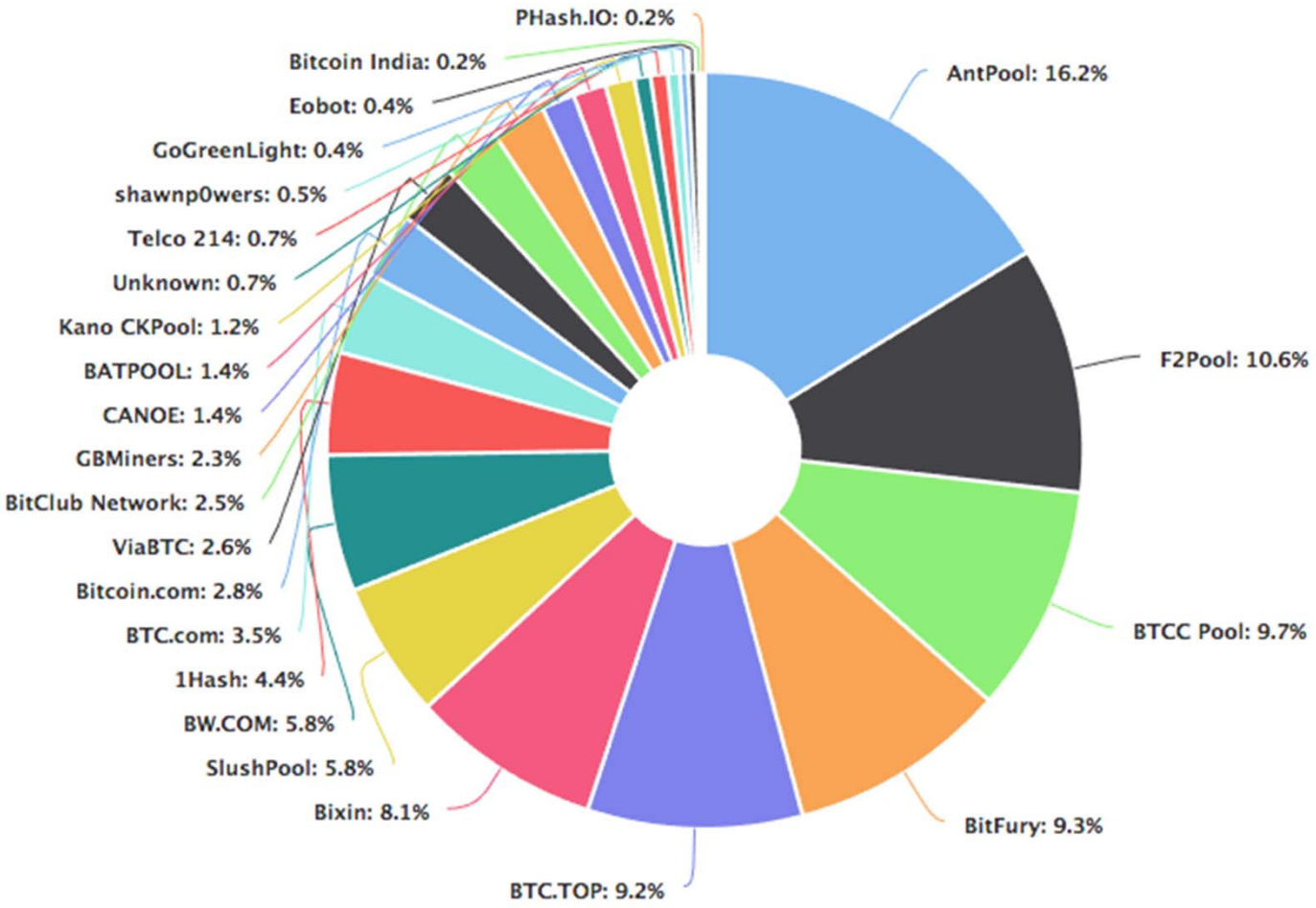
Important nodes in that network: miners (use CPU& electricity to run chain)

Bitcoin blockchain: ownership of bitcoins.

Blockchain can be used for other assets & contracts (Ethereum)

Public blockchain: transparent, open, no central authority (≠ private blockchain)

Mining pools Bitcoin (by CPU)



The blockchain folk theorem

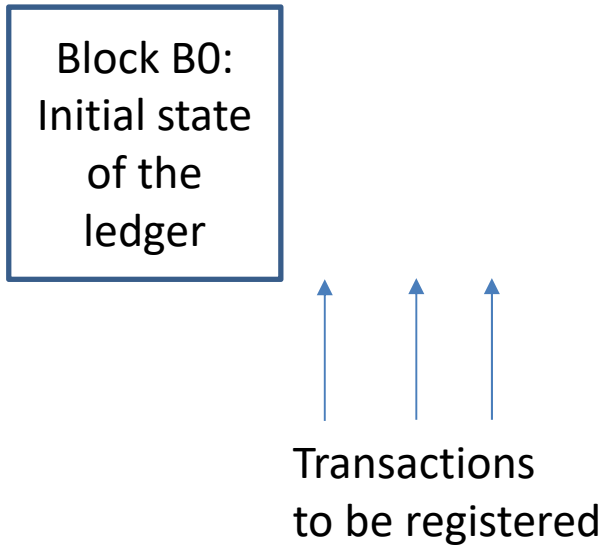
Unproven but often claimed:

“Blockchain is trust machine, automatically building immutable and unique consensus”

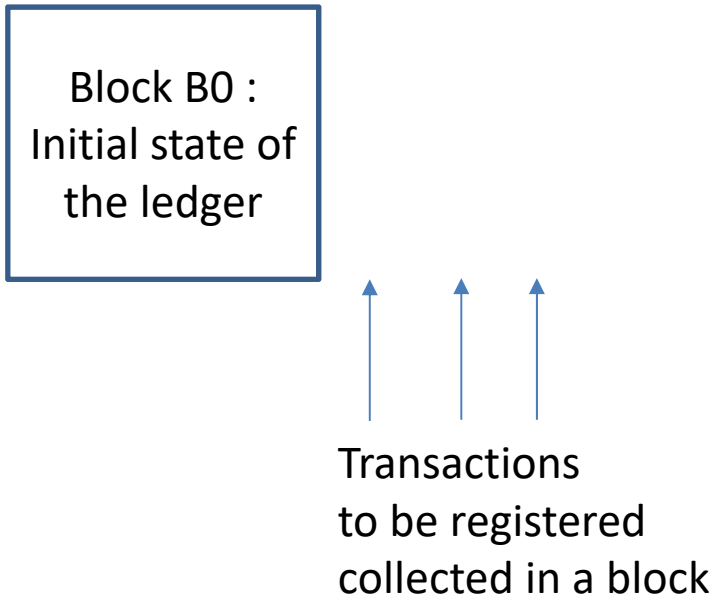
If that “theorem” is true, then blockchain = cost effective way to record transactions and ownership

Is it?

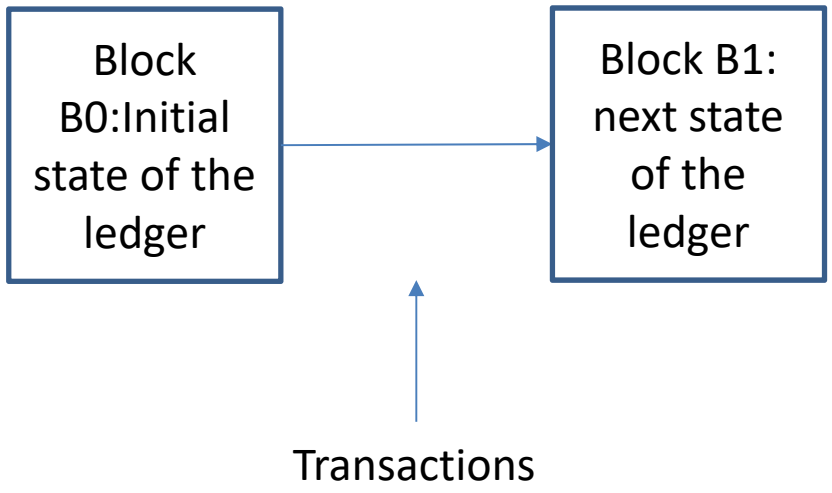
An ideal blockchain (1)



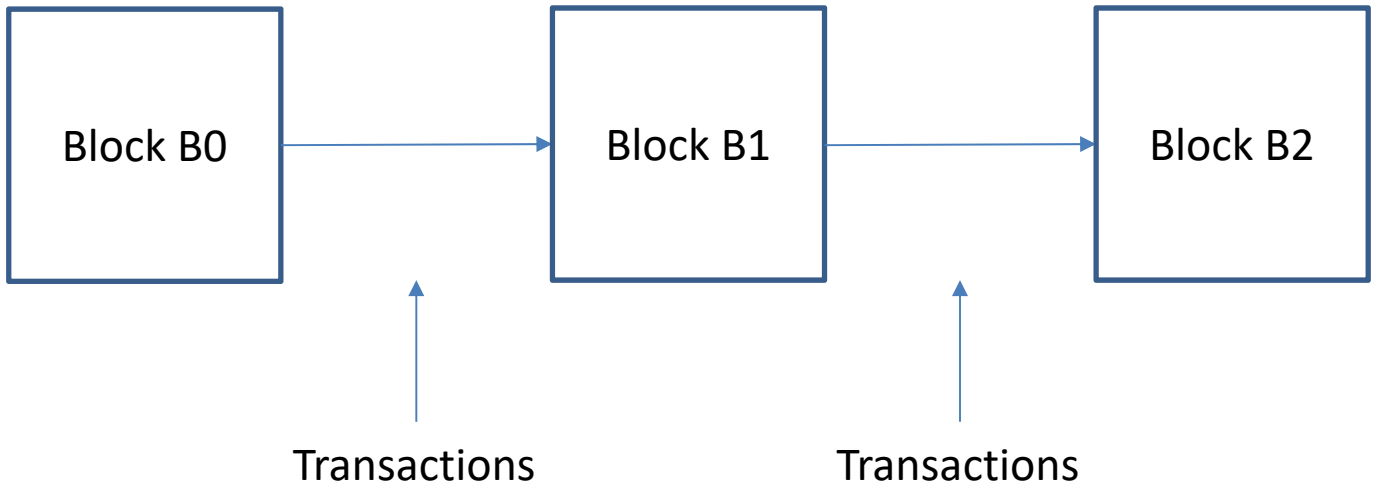
An ideal blockchain (2)



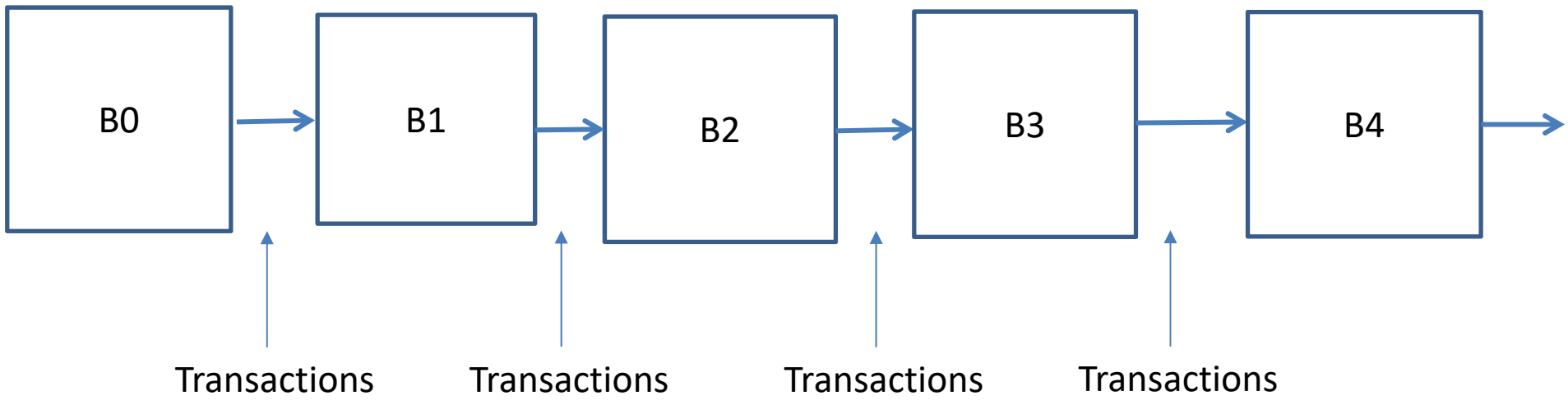
An ideal blockchain (3)



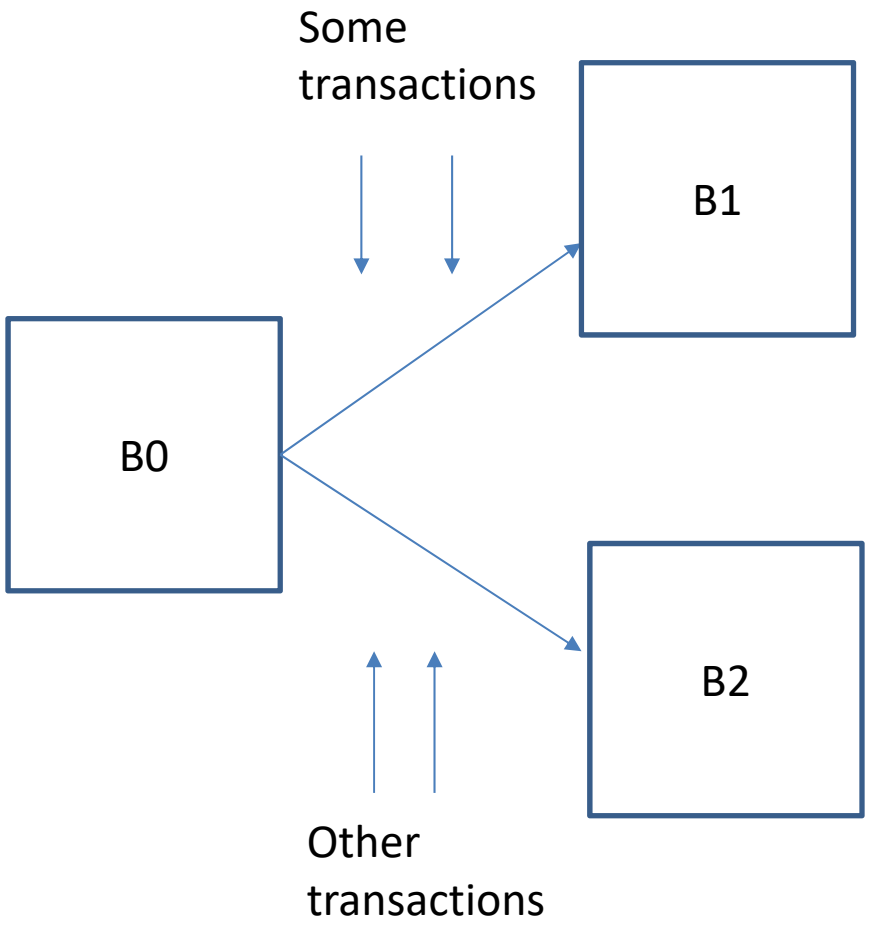
An ideal blockchain (4)



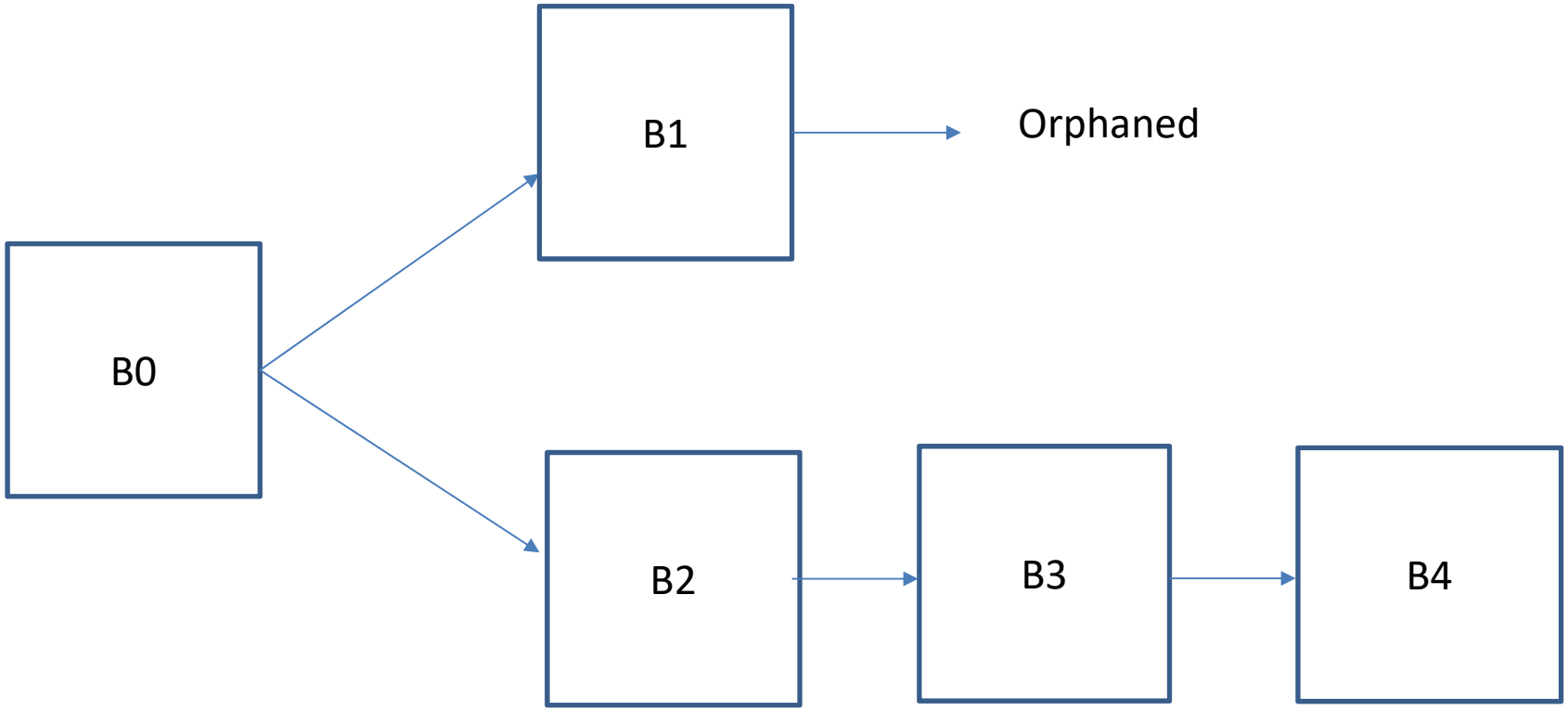
An ideal blockchain (5)



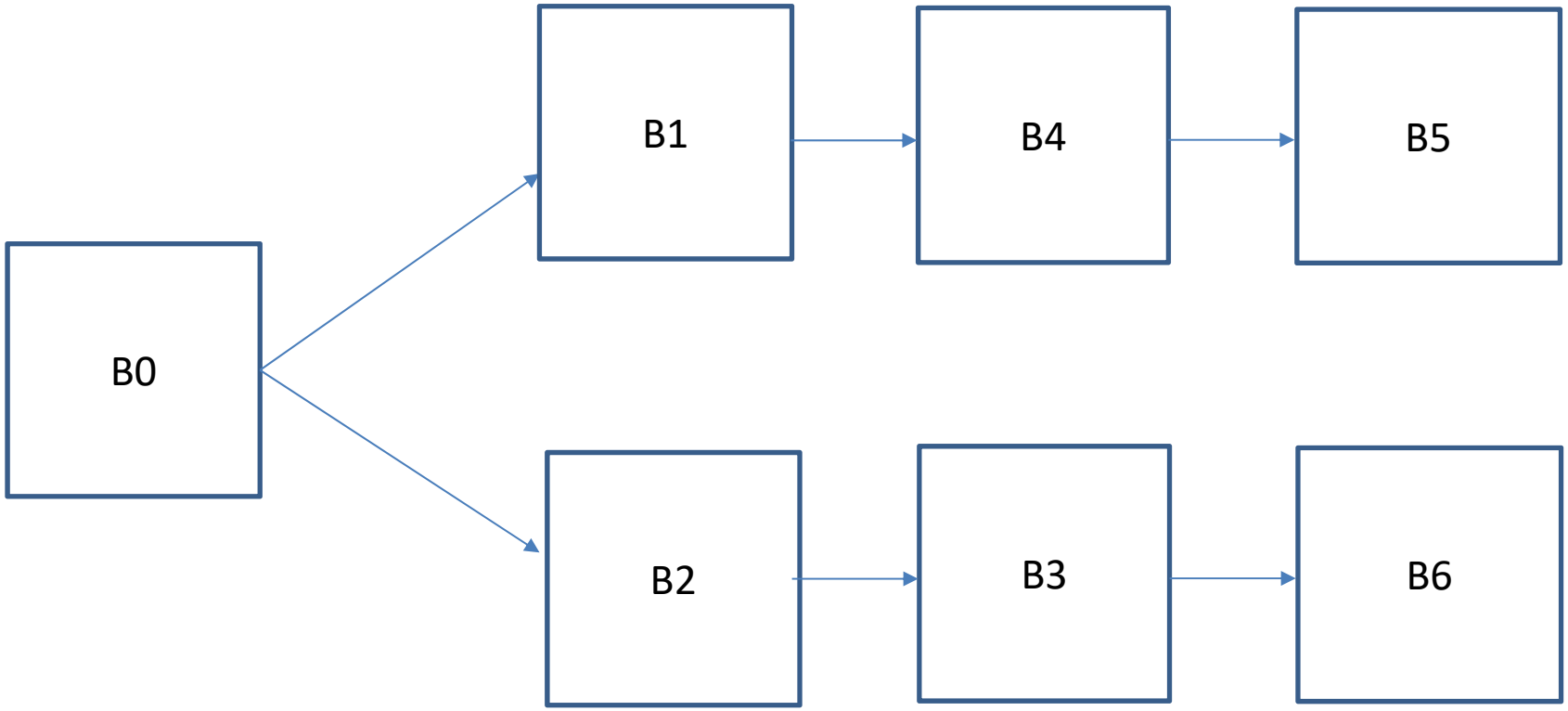
What could go wrong? Fork



What could go wrong? Orphan



What could go wrong? Persistent fork



What’s wrong with that?

Fork: Uncertainty about state of ledger, especially if persistent

Orphaned block: Mined in vain (waste of resources spent on mining) + uncertainty about transactions in orphaned block

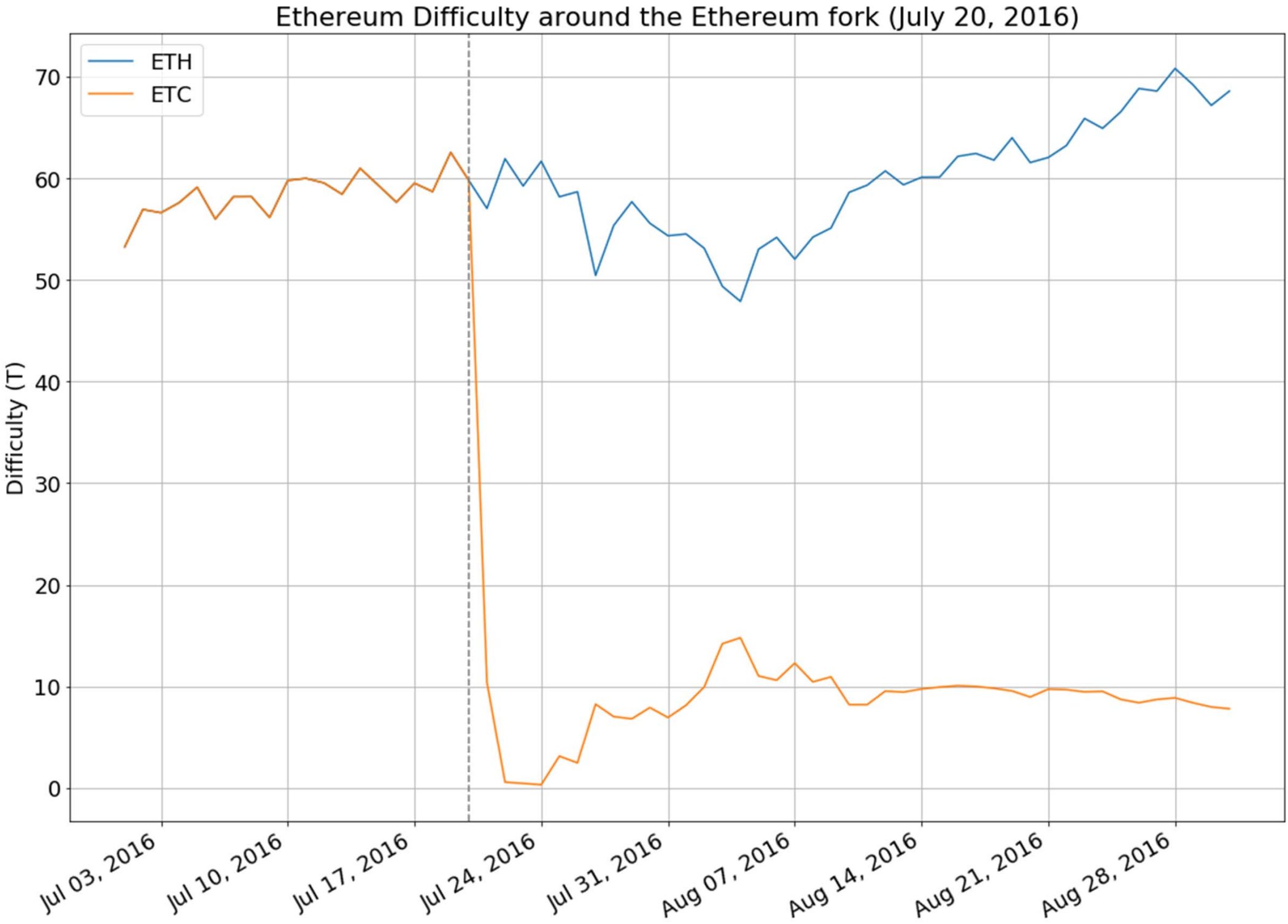
Does blockchain ever go wrong?

Transient fork with orphaned blocks:

- March 11, 2013, Bitcoin blockchain split
- ½ network adding blocks to 1 branch, other ½ to other
- Lasted 6 hours
- Eventually one branch (24 blocks) abandoned

Persistent forks:

- Summer 2016: Ether / Ether classic
- Summer 2017: Bitcoin / Bitcoin cash.


Ethereum Difficulty around the Ethereum fork (July 20, 2016)
ETH
ETC
Difficulty (T)
70
60
50
40
30
20
10
0
Jul 03, 2016
Jul 10, 2016
Jul 17, 2016
Jul 24, 2016
Jul 31, 2016
Aug 07, 2016
Aug 14, 2016
Aug 21, 2016
Aug 28, 2016

Issue

What can jeopardise blockchain consensus, trigger forks, orphan blocks?

→ Miners' actions

To analyse miners' actions:

describe rules of blockchain → game

analyse equilibrium of that game

What do miners do?

Collect transactions in a block

Decide to which previous block to chain their current block

Use computers and electricity to solve difficult cryptographic problem (purely numerical problem, completely unrelated to nature and quantity of economic transactions in block)

Once block solved: broadcast to network. Nodes can easily check if solution correct

Express acceptance of this block by chaining theirs to it

What do they mine for?

In the block she mines, the miner includes the creation of units of cryptocurrency (12.5 for bitcoin) which she allocates to herself (crypto money creation/seigneuriage)

Reward valuable only if block included in consensus chain by other miners (who do so by chaining their own blocks to it)

→ mining = coordination game: I want to mine in the branch in which I anticipate the others will mine

→ equilibrium multiplicity

The longest chain rule

When the chain splits in several branches (i.e. when there is a fork)

Nakamoto (2008) advises miners should chain to the longest branch

If miners always do so, then there is a single chain; and we show there is a Markov perfect equilibrium, in which miners do so

But miners must coordinate on deciding to follow that rule; and we show there are equilibria, in which miners don't follow the longest chain rule → equilibria with forks

Coordination issues during March 2013 Bitcoin fork

Two competing chains (0.7 versus 0.8): Miners didn't know which chain to coordinate on (uncertainty → Bitcoin price dropped 25%)

"Gavin Andresen: the 0.8 fork is longer, yes? so majority haspower is 0.8 ... first rule of bitcoin: majority haspower wins

Luke Dashjr: if we go with 0.8 we are hard forking

BTC Guild: I can single handedly put 0.7 back to the majority hash power. I just need confirmation that that's what should be done.

Pieter Wuille: that is what should be done, but we should have consensus first"

Eventually, BTC guild chose 0.7, 0.8 orphaned, 24 blocks lost

Miners and blocks

Miners, $m \in \mathcal{M} = \{1, ...M\}$, risk neutral, rational

Exogenous initial state of ledger at time 0 = block B_0

Starting from B_0 participants start mining first block B_1

As time goes by, miners observe the set of solved blocks

At any time miners decide to which previously solved block they want to chain the block they currently mine (action space = set of previously solved blocks.)

Solving blocks

Difficulty of cryptographic problem and computing power $\longrightarrow$ Time it takes miner to solve his block = exponential

$$\Pr(m \text{ solves block between } t \text{ and } t + dt) = \theta_m dt$$

independent of

- how long m has been mining the block
- which block m is mining
- which block the others are mining

$\longrightarrow$ Suppose m has been mining block B_n, and another miner solves: duration until next time m solves independent of whether m continues to mine B_n or any other block
$\longrightarrow$ Number blocks solved by m at t = exogenous random variable

$$N_m(t) = \int_{s=0}^{t} dN_m(s)$$

Stochastic maturity

At time z_m (exponential with intensity λ) miner m hit by liquidity shock $\longrightarrow$ must consume real goods $\longrightarrow$ sells cryptocurrency earned as reward to a new miner, who also inherits his beliefs

Exit compensated by entry $\longrightarrow$ stationary environment

Before z_m miner m keeps cryptocurrency obtained as reward (k-blocks rule: on Bitcoin $k = 100$)

Rewards for mining blocks

In block he mines, m includes his reward: G units of cryptocurrency (eg Bitcoins or ETH) (+ transaction fee, much smaller): miners earn *Seigneuriage* on cryptocurrency

Value of reward for mining block in chain $\mathcal{B}$ depends on number of miners active in $\mathcal{B}$

Key assumptions:

- G increasing in number of miners active in $\mathcal{B}$
- $G = 0$ if 0 or only 1 miner in $\mathcal{B}$ (orphaned)

State, strategy and equilibrium

At any time τ, state, ω_τ, includes

- tree of previously solved blocks
- number of miners active on different branches
- public randomisation device

Strategy: maps state ω_τ into action: where, in tree of previously solved blocks, to chain current block

We look for Markov Perfect Equilibria (MPE) (subgame perfect Nash in which action depends only on current state)

Maximum miner's gains

Upper bound on m's lifetime payoff

$$\mathcal{G}_m^{\max} = \left[\int_{s=0}^{z_m} dN_m(s)\right] G(M).$$

- total number of blocks solved by m before z_m: $\int_{t=0}^{z_m} dN_m(t)$, irrespective of mining strategy of m and $-m$
- m cannot earn more than $G(M)$ each time he solves a block.

At t, expectation of $\mathcal{G}_m^{\max}$, *conditional on* $z_m > t$

$$E_t\left[\int_{s=0}^{t} dN_m(s) + \int_{s=t}^{z_m} dN_m(s)\right] G(M).$$

LCR is Nash equilibrium

Proposition 1:
There exists a Markov Perfect Equilibrium in which, on the equilibrium path, there is a single chain and all miners follow the longest chain rule (LCR), thus obtaining the maximum expected payoff $E(\mathcal{G}_m^{\max})$

If m follows LCR, like the others
$\Longrightarrow$ at z_m only one chain (with M active miners)
$\Longrightarrow$ each block mined by m earns $G(M)$
$\Longrightarrow$ m obtains maximum possible gain: $\mathcal{G}_m^{\max}$
$\Longrightarrow$ no deviation yields strictly greater expected payoff
$\Longrightarrow$ optimal (at least weakly) for m to follow LCR

Coordination rather than competition

Miners are not really competing to solve their block before the others

That someone else solves his block before me, does not, in itself, reduce my gains

The only thing that matters is that we all coordinate well, and all work on the same chain, so that we all obtain maximum rewards for the blocks we solve

Proposition 1, entirely driven by coordination effects, does not depend on number of miners, also holds in large number of miners limit

Sunspot equilibrium fork

LCR not unique equilibrium. Denote $B_{n(\tau)}$ last block solved by τ

Proposition 2:
There exists a MPE in which at a random time τ (sunspot time) all miners fork, chaining to $B_{n(\tau)-f}$, and following LCR on resulting chain.

Intuition: expect all to fork
$\longrightarrow$ expect only blocks on fork to be rewarded
$\longrightarrow$ also fork

Coordination game - strategic complementarity (again does not depend on number of miners)

Consequence: Fork becomes only active chain, blocks $B_{n(\tau)-f+1}$ to $B_{n(\tau)}$ orphaned, not rewarded $\Longrightarrow$ fork equilibrium Pareto dominated by LCR equilibrium

Persistent forks

Candidate equilibrium: Some fork at τ^f to new chain whose parent is $B_{n(\tau^f)-f}$, while others remain on original chain

Vested interest on original chain at time τ^f : $v^o(m, \tau^f) =$ number of blocks solved by m on original after $B_{n(\tau^f)-f}$

Rank miners by vested interest in original chain

$$v^o(m+1, \tau) \geq v^o(m, \tau)$$

Proposition 3:
Under some conditions, $\exists$ integer K s.t $\exists$ a MPE in which, at random time τ^f, miners $m \leq K$ (with low $v^o(m, \tau)$) fork to new chain (and hereafter remain there), while miners $m > K$ (with large $v^o(m, \tau)$) forever remain on original chain.

Intuition for Proposition 3

$K \geq \frac{M}{2}$: Persistent forks can occur only if majority of miners fork

Benefit from forking = blocks mined on new chain more valuable (because majority mines it)

Cost of forking = reduces value of blocks already mined on old chain: large if $v^{old}(m, t_{B_n})$ large

Persistent fork equilibrium Pareto dominated

LCR equilibrium: each block rewarded by $G(M)$

Persistent fork equilibrium: blocks solved after $t_{B_{n-f}}$ rewarded by $G(M-K)$ or $G(K) < G(M)$

Information transmission delays

Suppose m does not immediately observe that B_n was solved, and continues to mine from B_{n-1}: If m solves his block before the others, there are now two competing chains of same length

Suppose that from that point on all observe all blocks solved, there are 3 possible equilibria:

- All ignore m and stick to original chain $\longrightarrow$ m's block quickly orphaned, only transient one-block fork
- All focus on m and abandon the original chain $\longrightarrow$ B_n orphaned, only transient one-block fork
- All but m, stick to original chain, while m sticks to his block: If m first to solve, then all chain to this blocks (original chain block B_n orphaned), otherwise all revert to original chain (m's block orphaned)

As in basic model: multiple equilibria. Information delay offers an interpretation for sunspot in Proposition 2.

Upgrade

Miners must choose whether to adopt upgrade or reject it

Hardfork: upgrade and original version not compatible

Again, multiple equilibria:

- All adopt the upgrade (because all anticipate others will adopt)
- None adopts (because all anticipate none will adopt) // Segwit2X
- If some miners derive private benefits from adopting, or from rejecting, there can exist equilibria in which some adopt and others don't, so that chain splits and fork occurs // ETH vs ETC

Endogenous computing power

Mining = just a way to randomise who wins in decentralised manner

Planner would prefer all miners to acquire only very small computing power ε, so that randomisation can be achieved without wasting too much resources (electricity, hardware)

But, if I anticipate others to acquire only ε, I find it optimal to acquire larger computing power, to increase my chances to win

By doing so I exert an negative externality: I increase total computing power, hence difficulty, hence I reduce the frequency with which others solve

In equilibrium: all acquire computing power $>> \varepsilon$: equilibrium not socially optimal, due to negative externalities [electricity wasted, CO2 emitted]

Conclusion

LCR equilibrium and single persistent chain with no fork cannot be ruled out... but cannot be taken for granted

Number of miners/computing power (and end users) on a chain $\longrightarrow$ Credibility of chain $\longrightarrow$ Value of rewards for blocks mined on that chain $\longrightarrow$ Attractiveness of that chain

Coordination game $\longrightarrow$ Multiple equilibria

- Instability?
- Pareto dominated (waste of resources)
- Forks