The Economic Limits of Bitcoin and the Blockchain

Eric Budish
Chicago Booth

NBER Summer Institute, Monetary Economics
July 2018

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from “Proof-of-Work” Consensus Mechanism
- Amount of computational work must simultaneously:

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"
  - (2) deter "majority attack"

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"
  - (2) deter "majority attack"
- Together, (1)+(2) imply:

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from “Proof-of-Work” Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain “miners”
  - (2) deter “majority attack”
- Together, (1)+(2) imply:
  - (3) recurring, “flow” costs of maintaining the blockchain must be large relative to one-off, “stock” benefits of attacking it
  - Very expensive! Like a large implicit tax.

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"
  - (2) deter "majority attack"
- Together, (1)+(2) imply:
  - (3) recurring, "flow" costs of maintaining the blockchain must be large relative to one-off, "stock" benefits of attacking it
  - Very expensive! Like a large implicit tax.
- Way out (i.e., why has Bitcoin not been attacked yet)
  - (i) mining technology is both scarce and non-repurposable, <u>and</u>
  - (ii) any majority attack is a "sabotage" in that it causes a collapse of economic value of the blockchain

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"
  - (2) deter "majority attack"
- Together, (1)+(2) imply:
  - (3) recurring, "flow" costs of maintaining the blockchain must be large relative to one-off, "stock" benefits of attacking it
  - Very expensive! Like a large implicit tax.
- Way out (i.e., why has Bitcoin not been attacked yet)
  - (i) mining technology is both scarce and non-repurposable, <u>and</u>
  - (ii) any majority attack is a "sabotage" in that it causes a collapse of economic value of the blockchain
- But: vulnerability to sabotage is a serious concern, and (i) also points to specific collapse scenarios

Overview of the Argument

- Nakamoto (2008) Blockchain Innovation: Anonymous, Decentralized Trust from "Proof-of-Work" Consensus Mechanism
- Amount of computational work must simultaneously:
  - (1) satisfy zero-profits condition for blockchain "miners"
  - (2) deter "majority attack"
- Together, (1)+(2) imply:
  - (3) recurring, "flow" costs of maintaining the blockchain must be large relative to one-off, "stock" benefits of attacking it
  - Very expensive! Like a large implicit tax.
- Way out (i.e., why has Bitcoin not been attacked yet)
  - (i) mining technology is both scarce and non-repurposable, <u>and</u>
  - (ii) any majority attack is a "sabotage" in that it causes a collapse of economic value of the blockchain
- But: vulnerability to sabotage is a serious concern, and (i) also points to specific collapse scenarios
- Overall take: ingenious, but economically may be limited. If it gets economically important enough, it will get attacked

What is Blockchain (1/4)

- **Transactions** consist of
  - Sender address
  - Receiver address
  - Amount
  - Sender's signature
- Signature:
  - Can only be generated by holder of sender's private key (presumably the sender!)
  - Yet does not reveal the key
  - Encodes the transaction information too — so can't tamper with amount, destination, etc., without key
  - Magic, but completely standard cryptography. Not new to cryptocurrencies.

What is Blockchain (2/4)

- Imagine transactions on a google spreadsheet
  - Signature: only I can initiate transactions in which I send money
  - But:
    - I can send money I don't have
    - I can send money I do have but to multiple parties at the same time.
    - I can delete previous transactions (mine or others')
  - Works fine if we trust each other, not if we don't
- Imagine transactions through a trusted party that keeps track of balances
  - That works just fine re: security issues listed above
  - But: requires a trusted party

What is Blockchain (3/4)

Nakamoto (2008) Blockchain Innovation

- Users submit transactions to a pending transactions list
- Every $\sim$ 10 minutes, "miners" engage in a computational tournament for the right to add a new block of transactions to a chain
  - Each new block "chains" to previous block
  - Transactions can only be added to a block if valid given previous blocks, other transactions in this block
- Computational tournament:
  - Find a "lucky hash" that is a function of
    - New block of transactions
    - Previous block of transactions
  - Called "proof of work" – hard to find, easy to check
- Miner who finds a lucky hash reports new block, previous block it chains to, and the lucky hash
- Successful miner earns "block reward"

What is Blockchain (4/4)

- Nakamoto (2008): "[miners] express their acceptance of the [new] block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash."
- Nakamoto (2008) convention, in case there are multiple chains: longest-chain as measured by amount of computational work
- From the abstract:

  "The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain serves not only as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power.

Clarification

- As interest in Bitcoin and its blockchain have surged, some have started to use the phrase "blockchain" to describe distributed databases among *known, trusted parties* – that is, *without* the central innovation of Nakamoto (2008)

Clarification

- As interest in Bitcoin and its blockchain have surged, some have started to use the phrase "blockchain" to describe distributed databases among *known, trusted parties* – that is, *without* the central innovation of Nakamoto (2008)

  "If you announce that you are updating the database software used by a consortium of banks to track derivatives trades, the New York Times will not write an article about it. If you say that you are blockchaining the blockchain software used by a blockchain of blockchains to blockchain blockchain blockchains, the New York Times will blockchain a blockchain about it." (Matt Levine, 2017)

Clarification

- As interest in Bitcoin and its blockchain have surged, some have started to use the phrase "blockchain" to describe distributed databases among *known, trusted parties* – that is, *without* the central innovation of Nakamoto (2008)

  "If you announce that you are updating the database software used by a consortium of banks to track derivatives trades, the New York Times will not write an article about it. If you say that you are blockchaining the blockchain software used by a blockchain of blockchains to blockchain blockchain blockchains, the New York Times will blockchain a blockchain about it." (Matt Levine, 2017)

- I use blockchain in sense of Nakamoto (2008) innovation, not distributed databases more broadly

Outline of Talk

1. Blockchain: A Critique in 3 Equations
   1.1 Rent-Seeking Competition (Miners)
   1.2 Incentive Compatibility (Majority Attack)
   1.3 Economic Constraint on the Blockchain: Flow vs. Stock
2. Majority Attack Scenarios
   2.1 Attack I: Double Spending
   2.2 Attack II: Sabotage ("Pick your poison")
3. A Way Out: Sabotage + Blockchain-Specific Mining Technology
   3.1 Softer Constraint: Stock vs. Stock. May explain why Bitcoin not attacked yet.
   3.2 Suggests collapse scenarios

Outline of Talk

1. **Blockchain: A Critique in 3 Equations**
   1.1 **Rent-Seeking Competition (Miners)**
   1.2 **Incentive Compatibility (Majority Attack)**
   1.3 **Economic Constraint on the Blockchain: Flow vs. Stock**
2. Majority Attack Scenarios
   2.1 Attack I: Double Spending
   2.2 Attack II: Sabotage ("Pick your poison")
3. A Way Out: Sabotage + Blockchain-Specific Mining Technology
   3.1 Softer Constraint: Stock vs. Stock. May explain why Bitcoin not attacked yet.
   3.2 Suggests collapse Scenarios

Rent-Seeking Competition (Miners)

- P_{block} : economic reward to miner who wins computational tournament
  - Assume exogenous; will place constraints below
- c: per-block cost of one unit of computational power
  - Per-block electricity costs + Per-block cost of capital, incl. depreciation. Notationally: $c = rC + e$
- Assume for now capital easily repurposable
  - *Not* true for Bitcoin at present (ASICs)
  - Does capture Nakamoto ideal of "one-CPU-one-vote"
  - Will revisit in detail later
- N units of computational power $\rightarrow \frac{1}{N}$ prob of winning P_{block}
- Honest mining, Free entry equilibrium

$$N^* c = P_{block} \tag{1}$$

- Note: (1) widely known (many papers, Bitcoin Wiki)

Incentive Compatibility (Majority Attack)

- Well-known that blockchain vulnerable to majority attack
- Abstract of Nakamoto (2008):

  "The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain serves not only as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers." (Emphasis added)

- Bitcoin Wiki:

  "Bitcoin's security model relies on no single coalition of miners controlling more than half the mining power"

Incentive Compatibility (Majority Attack)

- What is cost of a majority?
- Outside attacker, simple majority: $N^* c + \epsilon$ per block
- Inside attacker: as little as $\frac{N^* c}{2}$ per block
- Outside attacker with $\frac{A}{A+1}$ majority: $AN^* c$ per block
- Assume exists attack with
  - Payoff V_{attack} (discuss more below)
  - Takes A attacker t periods in expectation (simulated below)
- Cost net of block rewards: $At \cdot N^* c - tP_{block}$
- Using (1) and defining $\alpha = (A-1)t$, cost is $\alpha \cdot N^* c$
- Incentive constraint:

$$\alpha \cdot N^* c > V_{attack} \tag{2}$$

Incentive Compatibility (Majority Attack)

$$\alpha \cdot N^{*} c > V_{attack} \tag{2}$$

- (2) captures that what enables "decentralized trust" of the blockchain is the computing power devoted to maintaining it
- Economically
  - LHS is related to *flow* cost of maintaining the blockchain
  - Contrast: mutually-beneficial cooperation in a relationship and temptation to cheat, or trusted brand tempted to shirk on quality
  - Cost of cheating: stock value of relationship or brand, not flow cost of maintenance
- Computer security
  - Security is *linear* in amount of computational power
  - Many other IT security investments yield convex returns (e.g., traditional crypto)
  - Analogy: lock on door

Critique

- In hoped-for eqm with honest mining, amount of computational power characterized by (1), $N^* c = P_{block}$
- Combine with incentive compatibility (2), $\alpha \cdot N^* c > V_{attack}$
- Yields:

$$P_{block} > \frac{V_{attack}}{\alpha} \tag{3}$$

- In words: *the eqm per-block payment to miners for running the blockchain has to be large relative to the one-off benefits of attacking it*
- Flow payment to miners > Stock value of attack
- Imagine if users of Visa network had to pay fees to Visa, every 10 minutes, large relative to successful one-off attack

Outline of Talk

1. Blockchain: A Critique in 3 Equations
    1.1 Rent-Seeking Competition (Miners)
    1.2 Incentive Compatibility (Majority Attack)
    1.3 Economic Constraint on the Blockchain: Flow vs. Stock
2. **Majority Attack Scenarios**
    2.1 **Attack I: Double Spending**
    2.2 **Attack II: Sabotage ("Pick your poison")**
3. A Way Out: Sabotage + Blockchain-Specific Mining Technology
    3.1 Softer Constraint: Stock vs. Stock. May explain why Bitcoin not attacked yet.
    3.2 Suggests collapse Scenarios

What Can An Attacker Do?

- A majority attacker can
  - Solve computational puzzles faster, in expectation, than the honest minority
  - Create an alternative longest chain, replace the honest chain at a strategically opportune moment
  - This allows the attacker to:
    - Control what transactions get added to the blockchain
    - Remove recent transactions from the blockchain
  - The attacker also earns the block rewards, for each period of his alternative chain
- A majority attacker cannot
  - Create new transactions that spend other participants' Bitcoins ("steal all the Bitcoins")
  - This would require not just >50% majority, but breaking modern cryptography

(Good source: Bitcoin Wiki, "Attacker Has a Lot of Computing Power")

Attack I: Double Spending

- Double spending attack
  - (i) spend Bitcoins — i.e., engage in a transaction in which he sends Bitcoins to a merchant in exchange for goods or assets
  - (ii) allow that transaction to be added to the blockchain
  - (iii) subsequently remove the transaction from the blockchain, perhaps after an escrow period
- To translate into values for V_{attack} and α, assume:
  1. k transactions in a block
  2. attacker engages in 1 block worth of transactions, i.e., k distinct transactions
  3. average value: $\bar{v}_{transaction}$
  4. escrow period of e blocks
  5. attack does *not* affect subsequent value of Bitcoins (will be relaxed in a moment)
- Under these assumptions, (3) becomes [$p_{trans} = P_{block}/k$]:

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$
- These α's interpretable as $2\% - 60\%$ "tax" on *largest* possible transactions
  - $\bar{v}_{transaction} = \$1000$: current $p_{transaction}$ completely plausible
  - $\bar{v}_{transaction} = \$1000000$ ("store of value"): need $p_{transaction}$ btwn \$20k-\$100k. Even with some "slippage", this seems high.

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$
- These α's interpretable as $2\% - 60\%$ "tax" on *largest* possible transactions
  - $\bar{v}_{transaction} = \$1000$: current $p_{transaction}$ completely plausible
  - $\bar{v}_{transaction} = \$1000000$ ("store of value"): need $p_{transaction}$ btwn \$20k-\$100k. Even with some "slippage", this seems high.
- Takeaways:

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$
- These α's interpretable as $2\% - 60\%$ "tax" on *largest* possible transactions
  - $\bar{v}_{transaction} = \$1000$: current $p_{transaction}$ completely plausible
  - $\bar{v}_{transaction} = \$1000000$ ("store of value"): need $p_{transaction}$ btwn \$20k-\$100k. Even with some "slippage", this seems high.
- Takeaways:
  - Consistent with early use cases of Bitcoin

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$
- These α's interpretable as $2\% - 60\%$ "tax" on *largest* possible transactions
  - $\bar{v}_{transaction} = \$1000$: current $p_{transaction}$ completely plausible
  - $\bar{v}_{transaction} = \$1000000$ ("store of value"): need $p_{transaction}$ btwn \$20k-\$100k. Even with some "slippage", this seems high.
- Takeaways:
  - Consistent with early use cases of Bitcoin
  - Casts doubt on "store of value" story, major component of global financial system

Double Spending Attack

$$p_{transaction} > \frac{\bar{v}_{transaction}}{\alpha}$$

- Computational simulations to find α for different majority power A and escrow periods e:
  - $A = 1.25, e = 0$: duration 6.54 blocks, net cost $\alpha = 1.64$
  - $A = 1.25, e = 6$: duration 13.41 blocks, net cost $\alpha = 3.35$
  - $A = 1.05, e = 100 \rightarrow \alpha = 9.2$; $e = 1000 \rightarrow \alpha = 53.5$
- These α's interpretable as $2\% - 60\%$ "tax" on *largest* possible transactions
  - $\bar{v}_{transaction} = \$1000$: current $p_{transaction}$ completely plausible
  - $\bar{v}_{transaction} = \$1000000$ ("store of value"): need $p_{transaction}$ btwn \$20k-\$100k. Even with some "slippage", this seems high.
- Takeaways:
  - Consistent with early use cases of Bitcoin
  - Casts doubt on "store of value" story, major component of global financial system
  - For the system to be secure for large transactions requires implicit tax rates that render it unusable for small ones

Attack II: Sabotage

- Obvious response: double spending would be "noticed"
- Cause decline in value of Bitcoin, which attacker needs to hold
- Bitcoin Wiki classifies majority attack "Probably Not a Problem" for this reason
- Formally: suppose Bitcoin value declines by proportion Δ_{attack}
- Constraint is now:

$$p_{transaction} > \frac{(1 - \Delta_{attack})}{(A - 1 + \Delta_{attack})t} \bar{v}_{transaction}$$

- If Δ_{attack} large enough, then indeed deter double spending
- However, "pick your poison":
  - Need to concede possibility of sabotage/collapse
  - Then should worry about attacker motivated by sabotage per se: $V_{sabotage}$
  - Either: high implicit tax rates or risk of collapse

Attack II: Sabotage

- What is $V_{sabotage}$?
- Hard to say of course, but easy to imagine that the magnitudes are already large, and would be larger still if Bitcoin / blockchain live up to the hype
  - Market cap: $100B-$150B (Gold: $7.5T)
  - Open interest on CME, CBOE futures: $150M (Gold: $65B)

Attack II: Sabotage

- What is $V_{sabotage}$?
- Hard to say of course, but easy to imagine that the magnitudes are already large, and would be larger still if Bitcoin / blockchain live up to the hype
  - Market cap: $100B-$150B (Gold: $7.5T)
  - Open interest on CME, CBOE futures: $150M (Gold: $65B)
- Goldman Sachs (2018): "Blockchain technology [that] was originally developed as part of the digital currency Bitcoin" is "The New Technology of Trust"
  - Applications include: "An international ID blockchain, accessible anywhere in the world, [that] allows people to prove their identify, connect with family members, and even receive money without a bank account."
  - Others have discussed blockchain for land provenance, medical records, and voting
- May be using "blockchain" as marketing term for older ideas from CS. But, to extent Nakamoto (2018) blockchain is used in these domains, we should worry about $V_{sabotage}$

Outline of Talk

1. Blockchain: A Critique in 3 Equations
   1.1 Rent-Seeking Competition (Miners)
   1.2 Incentive Compatibility (Majority Attack)
   1.3 Economic Constraint on the Blockchain: Flow vs. Stock
2. Majority Attack Scenarios
   2.1 Attack I: Double Spending
   2.2 Attack II: Sabotage ("Pick your poison")
3. **A Way Out: Sabotage + Blockchain-Specific Mining Technology**
   3.1 **Softer Constraint: Stock vs. Stock. May explain why Bitcoin not attacked yet.**
   3.2 **Suggests collapse Scenarios**

Blockchain-Specific Mining Technology

- Analysis so far has assumed attacker's cost is proportional to per-block "flow" cost of mining the blockchain
  - Formally, cost was $\alpha N^* c$ where $c = rC + e$ includes rental cost of capital, not fixed cost
- However, if both:
  - (i) technology necessary for mining the blockchain is specific (i.e., non-repurposable)
  - (ii) attack harms subsequent value of that technology (i.e., sabotage)
- Then it may be appropriate to charge the attacker a stock cost rather than a flow cost
- Importantly, (i) and (ii) both seem likely to hold for the Bitcoin blockchain at present

Blockchain-Specific Mining Technology

Flow cost approach appropriate under four cases:

- **Case 1:** The most efficient chips are re-purposable
  - Original Nakamoto (2008) vision: "one-CPU-one-vote"
  - Not true for Bitcoin at present: ASICs
  - Note: some cryptocurrency proof-of-work protocols designed to be "ASIC resistant" (e.g., Ethereum)
- **Case 2:** The most efficient chips are specialized, but there are repurposable chips that are efficient enough for an attack
  - Not true for Bitcoin at present: ASICs are 1000s times more economically efficient than GPUs/FPGAs
  - May become true in future, e.g., improvements in FPGA-like technology

Blockchain-Specific Mining Technology

Flow cost approach appropriate under four cases:

- **Case 3:** The most efficient chips are specialized, and there are previous-generation specialized chips that are not economically efficient for mining, but are efficient enough for an attack, and exist in large quantity
  - Formally: suppose efficient chip is $c^* = rC^* + e^*$ and there exists a previous gen chip with $\tilde{e} > c^*$.
  - If $\tilde{e}$ within a reasonable factor of e^*, then could be used for attack, even though not economical for mining even if free.
- **Case 4:** The attack isn't a sabotage
  - Insider could attack, pay flow cost, then go back to mining as usual.
  - Outsider could attack repeatedly, pay flow cost each time.

Blockchain-Specific Mining Technology

Flow cost approach is not appropriate, should instead charge attacker a stock cost, if:

- **Case 5:** The most efficient chips are specialized, there are neither reasonably efficient repurposable chips nor previous-gen specialized chips, and the attack is a sabotage
  - Likely satisfied for Bitcoin at present
  - ASICs 1000s times more efficient than repurposable alternatives
  - ASIC market seems mostly to be catching up with demand (e.g., Samsung recently announced entry)
  - ASIC technology has been improving dramatically, so previous-gen ASICs poor substitutes

Blockchain-Specific Mining Technology

- To analyze case 5, consider the extreme of total collapse of the economic value of the blockchain, including the specialized equipment
- This is the case for which the incentive constraint against the attack is least constraining
- Now IC constraint is

$$N^* C > V_{sabotage} \quad (2')$$

- Stock value on LHS, not flow. \$1.5B-\$2B vs. <\$1M-\$5M.

Blockchain-Specific Mining Technology

- To analyze case 5, consider the extreme of total collapse of the economic value of the blockchain, including the specialized equipment
- This is the case for which the incentive constraint against the attack is least constraining
- Now IC constraint is

$$N^{*} C > V_{sabotage} \tag{2'}$$

- Stock value on LHS, not flow. \$1.5B-\$2B vs. <\$1M-\$5M.
- Still, a meaningful economic constraint:
  - Still linear
  - Must concede both (i) possibility of sabotage, (ii) security relies on specialized equipment
- Amounts still small if Bitcoin becomes major "store of value" akin to gold, or major component of global financial system
  - \$Attack blockchain <<< \$Attack Fort Knox
  - \$Attack blockchain <<< \$Attack Federal Reserve

Collapse Scenarios

- Suppose, for purpose of discussion
  - Bitcoin blockchain *does* satisfy (2'): $N^* C > V_{attack}$
  - Bitcoin blockchain *does not* satisfy (2): $\alpha N^* c > V_{attack}$
- Model then suggests 3 possible scenarios that could precipitate collapse

1. Ultra-cheap specialized ASICs
   - As tech matures: cheap previous-gen versions, or current-gen version becomes cheap enough that electricity the predominant component of cost
   - If Bitcoin value falls (for other reasons): glut of ASICs relative to amt needed for mining eqm (1)
2. Efficient-enough repurposable chips
   - If blockchain grows in importance and repurposable chips get better at hashing then flow cost.
   - Improvements in FPGA-like technology
3. Economic sabotage becomes sufficiently tempting
   - Futures markets grow
   - Bitcoin grows in economic importance

Conclusion: Summary

- Anonymous, decentralized trust enabled by Nakamoto (2008) blockchain: *ingenious but expensive*
- Eq. (3): for trust to be meaningful, flow cost of running the blockchain > one-shot value of attacking it
  - Double spending attack: payments to miners must be large relative to the highest-value possible uses of the blockchain
  - Like a large implicit tax
- Argument that attack costs more than this flow requires one to concede both
  1. Security relies on use of scarce, non-repurposable tech (contra "one-CPU-one-vote")
  2. Vulnerable to sabotage, linear in amount of specialized computational equipment ("pick your poison")
- This then points to specific collapse scenarios
  - Conditions change in the chip market
  - Bitcoin becomes sufficiently economically important to tempt a saboteur

Conclusion: Remark

- Emphasize: model consistent with earliest uses of Bitcoin and blockchain
- Skepticism:
  - Bitcoin as "store of value" akin to gold
  - Bitcoin as a major component of the global financial system
  - Use of Nakamoto blockchain by businesses, governments
- Note: <u>not</u> skeptical re: use of distributed databases more broadly
- What this paper highlights is that it is exactly the aspect of Bitcoin and Nakamoto (2008) that is so innovative relative to traditional distributed databases — the anonymous, decentralized trust that emerges from proof-of-work — that is so economically constraining

Conclusion: Open Question

- Open question: are there other ways to generate anonymous, decentralized trust that make this paper's arguments less constraining?
  - More precisely: versions of (1)-(3) seem intrinsic to any anonymous, decentralized blockchain protocol
  - But is there a way to either reduce V_{attack} or raise α relative to a given level of P_{block}?
- Interesting in this regard: "proof of stake"
  - Usual motivation: reduce mining expense and environmental harm (Bitcoin is 0.3% of *global* electricity consumption)
  - Environmental issue is orthogonal to the concerns raised in this paper. Just conceptualize c as per-block opportunity cost of holding one unit of stake
  - But: use of stakes rather than work may open up new possibilities for thwarting attacks.
- Active area ... will wait and see if there is a breakthrough.
- Or, perhaps there is a theorem waiting to be proved that no such breakthrough exists.