**WORKING PAPER** · NO. 2022-83

# Trust at Scale: The Economic Limits of Cryptocurrencies and Blockchains

*Eric Budish*

DECEMBER 2023

Becker Friedman Institute
FOR ECONOMICS AT UCHICAGO

5757 S. University Ave.
Chicago, IL 60637
Main: 773.702.5599
bfi.uchicago.edu

# Trust at Scale: The Economic Limits of Cryptocurrencies and Blockchains*†

Eric Budish‡

December 12, 2023

**Abstract**

Satoshi Nakamoto (2008) invented a new kind of economic system that does not need the support of government or rule of law. Trust and security instead arise from a combination of cryptography and economic incentives, all in a completely anonymous and decentralized system. This paper uses a simple three-equation argument to show that this new form of trust, while ingenious, is deeply economically limited. A zero-profits condition on the providers of blockchain trust and an incentive-compatibility condition on the system's security against attack together imply an equilibrium constraint that the recurring, "flow" cost of blockchain trust must be large relative to the one-off, "stock"-like benefit of attacking the system. Moreover, this equilibrium cost of blockchain trust scales linearly with the value secured — which means that if cryptocurrencies and blockchains were to become more economically useful than they have been to date, then their costs would have to grow to absurd levels. There is a way out of the flow-stock argument but it is premised on the risk of economic collapse, which is itself a serious problem – pick your poison. The key contrast between Nakamoto trust and traditional trust grounded in rule of law, and complementary sources such as reputations, relationships and collateral, is the economies of scale that arise from credible deterrence as in Hayek (1960) and Becker (1968): Society or a firm pays a fixed cost to enjoy trust over a large quantity of economic activity at low or zero marginal cost.

---

[*] This paper originally circulated in June 2018 in a shorter form as Budish (2018).

[†] Acknowledgments: I am grateful to the co-editor Andrei Shleifer and five anonymous referees for advice that greatly improved the paper. Thanks are also due to Susan Athey, Vitalik Buterin, Glenn Ellison, Gene Fama, Alex Frankel, Joshua Gans, Edward Glaeser, Austan Goolsbee, Hanna Halaburda, Zhiguo He, Joi Ito, Steve Kaplan, Anil Kashyap, Judd Kessler, Scott Kominers, Randall Kroszner, Robin Lee, Jacob Leshno, Andrew Lewis-Pye, Shengwu Li, Neale Mahoney, Gregor Matvos, Sendhil Mullainathan, Vipin Narang, Neha Narula, David Parkes, Tim Roughgarden, John Shim, Scott Stornetta, Adi Sunderam, Chad Syverson, Alex Tabarrok, Rakesh Vohra, Aviv Zohar, and seminar participants at Chicago Booth, the MIT Digital Currency Initiative, NBER Monetary Economics, Harvard, Carnegie Mellon, UPenn, Virtual Market Design, UIC, University of Tokyo, Northwestern, Iowa State Market Design Conference, Columbia, Stanford and NBER Market Design. Ethan Che, Natalia Drozdoff, Matthew O'Keefe, Anand Shah, Peyman Shahidi, Jia Wan and Tianyi Zhang have provided excellent research assistance. Disclosure: the author is a technical advisor to a project pursuing frequent batch auctions for decentralized finance. The author does not have any other financial interests that relate to this research.

[‡] University of Chicago Booth School of Business, eric.budish@chicagobooth.edu

# 1 Introduction

Economists have long widely agreed that the market system requires some form of government and rule of law for support. This is uncontroversial among even the most free-market oriented thinkers. Adam Smith (1776) mostly argues for reducing government interference in markets, but he does not go all the way to zero, writing that "commerce and manufactures can seldom flourish long in any state" without a legal system, property rights and contract enforcement, as well as certain public goods. Hayek (1960) grapples at length with the paradox that to maximize freedom—which he defines as the absence of coercion—it is necessary to have a government that has the power to coerce. Friedman (1962) famously described the government's role establishing the "rules of the game" for the market system and acting as its "umpire." There is significant debate within modern economics about what else government should do beyond these basic supports for the market system (e.g., social insurance, correcting externalities), but that there is some role for government and rule of law is essentially taken for granted.

Satoshi Nakamoto (2008) invented a new kind of economic system that does not need the support of government or laws.[1] Trust and security instead arise from a combination of cryptography and economic incentives, all in a completely anonymous and decentralized system. The details are complex and will be described below in Section 2, but at a high level, Nakamoto (2008) invented a way to achieve what computer scientists call "permissionless consensus": a large, anonymous, decentralized, freely-entering and -exiting mass of computational power around the world is incentivized to pay attention to and collectively maintain a common data set, enabling trust in this data set without the need for rule of law or any specific trusted party. The parties involved do not even need to know the identity or number of other parties involved. This invention enabled cryptocurrencies, including Nakamoto's own creation Bitcoin. The data structure maintained by the large mass of computational power is called a blockchain.[2]

It is no understatement to say that Nakamoto's invention captured the world's attention. One oft-cited figure is the $3 trillion of market capitalization of Bitcoin and other crypto assets at their 2021 peak, but even this figure seems to understate the amount of cultural, political,

---

[1] The anti-government views of Nakamoto and other early Bitcoin enthusiasts have been widely documented. See Popper (2015) for one early account and satoshi.nakamotoinstitute.org as a primary source. A few example quotes from these sources give a sense: "It's completely decentralized with no server or central authority" (Nakamoto email, 1/8/2009); "a new territory of freedom" (Nakamoto email, 11/6/2008); "outside the reach of any government" (Popper, pg 48).

[2] Not widely appreciated is that the blockchain data structure, *without* the novel method of trust, significantly predates Nakamoto, at least in terms of the core scientific ingredients if not popular and commercial appreciation of its usefulness (Haber and Stornetta, 1991; Bayer, Haber and Stornetta, 1993). This data structure is sometimes called a permissioned blockchain, and is in essence a well-architected database that is append-only, has clear rules about what parties can add what data, and uses cryptography to prove that past data has not been deleted or tampered with. See Section 2.5.1 and Budish and Sunderam (2023) for further discussion.

and commercial attention that has been paid to blockchains and cryptocurrencies. This level of attention is understandable given the nature of the idea—the premise of an economic system without government or rule of law is radical. Yet, at the same time, the economic usefulness of Nakamoto's invention remains an open question. To date, the majority of cryptocurrency volume appears to be speculative, with the other most widely documented use case being black-market transactions (Makarov and Schoar, 2021; Gensler, 2021; Buterin, 2022).[3] Moreover, the majority of this speculative volume has been through cryptocurrency exchanges—which are, at least in principle, centralized, trusted financial intermediaries. That is, the largest volume use of cryptocurrencies to date does not even take advantage of the novel form of trust.

So which view is correct? Can trust and security be engineered from cryptography and incentives alone? Or is rule of law essential for the market system?

This paper will show that Nakamoto's novel form of trust—while undeniably ingenious—is economically implausible, at least in its literal form without any implicit support from government or rule of law. It is too expensive in absolute terms relative to the stakes involved, and its expense *scales linearly* with the stakes involved. Put differently, if Nakamoto trust were to become more economically useful, then the costs of securing its trust would become preposterous. The analysis serves both as an explanation for why cryptocurrencies and blockchains have not been very economically useful to date, and as a reason to be skeptical that Nakamoto's anonymous, decentralized trust will play a major role in the global economy and financial system in the future. In so doing, this paper will also sharpen our conceptual understanding of what is special about traditional forms of trust that are grounded in rule of law and other complementary sources such as reputations, relationships and collateral. The key distinction will prove to be *economies of scale in the production of trust.*

The core of the paper's argument is just three simple equations. The first equation is a zero-profits condition that says that the amount of computing power devoted to maintaining Nakamoto trust will directly reflect the compensation paid to this computing power. For a sense

---

[3] Makarov and Schoar (2021) find that about 75% of Bitcoin transaction volume since 2015 involves cryptocurrency exchanges or exchange-like entitites, once the data are cleaned to account for spurious volume (such as a user moving their own funds from one address to another). They conclude that "the vast majority of Bitcoin transactions between real entities are for trading and speculative purposes." In a dataset from an earlier time period and using a different data cleaning and classification methodology, Foley, Karlsen and Putniņš (2019) find that 46% of Bitcoin transactions that do not involve cryptocurrency exchanges relate to illegal activity. Many credible public observers have also described cryptocurrency activity to date as mostly speculative or black-market. For example, Treasury Secretary Janet Yellen said in Feb 2021 "I don't think that bitcoin ... is widely used as a transaction mechanism ...To the extent it is used I fear it's often for illicit finance. ... It is a highly speculative asset." (Cox, 2021). SEC Chair Gary Gensler said in Aug 2021 "Primarily, crypto assets provide digital, scarce vehicles for speculative investment. ... These assets haven't been used much as a unit of account. We also haven't seen crypto used much as a medium of exchange. To the extent that it is used as such, it's often to skirt our laws ..." (Gensler, 2021). Ethereum founder Vitalik Buterin wrote in a Dec 2022 essay that he is most excited about applications still to come in the future, not the ones that already exist which he describes as "hyperfinancialized" (Buterin, 2022).

of magnitudes, in 2021/2022 the compensation to Bitcoin computing power (known as "miners") averaged about \$250,000 per block of data, or about \$36M per day, and Bitcoin miners performed an average of about 200 million trillion calculations per second as an equilibrium response to this compensation. The computer science details behind this process are complicated, and vary to some extent by blockchain protocol, but the economics is standard free-entry logic. Variations of this first equation have appeared in numerous other prior papers.

The second equation is an incentive compatibility condition: how much trust does a given level of computational power produce? The Achilles' heel of the form of trust invented by Nakamoto (i.e., permissionless consensus) is that it is vulnerable to what is known as a "majority attack." Nakamoto's method for creating an anonymous, decentralized consensus about the state of a dataset relies on a majority of the computing power devoted to maintaining the data to behave honestly. This is not an obscure point; it is in the *abstract* of the famous Nakamoto (2008) paper. Intuitively, permissionless consensus, whether in the form invented by Nakamoto or other subsequent variations such as proof-of-stake, always relies on some implicit version of majority or super-majority voting to adjudicate what the state is in case there is a dispute. The second equation captures that it must not be economically profitable for a potential attacker to acquire a 51% or greater majority of the total computational power to manipulate the state. Prior economic analysis of cryptocurrencies had mostly abstracted away the possibility of attack, focusing instead on other economic issues.

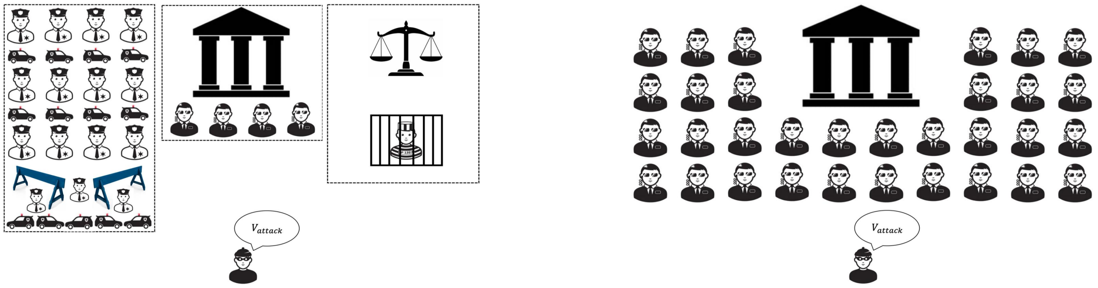
Equation (3) connects equations (1) and (2), i.e., connects the free-entry/zero-profits condition to the incentive compatibility condition. The reason these two equations can be linked is that the amount of honest computing power appears in both. In equation (1), the amount of honest computing power reflects the recurring payments to this computing power. In equation (2), the amount of honest computing power determines the cost of attack. Equation (3) then tells us that the payments to the honest computing power in the zero-profit equilibrium must be large relative to the value of attacking the system.

This is a very expensive form of trust! The recurring payments to miners are a "flow", whereas the value of attacking the system is more like a "stock." So equation (3) tells us that the flow-like costs of maintaining the trust must exceed the stock-like value of breaking the trust. A base case numerical calculation shows that this implies the annual cost of blockchain trust must be 400,000% of the value secured against attack — so it would take all of global GDP to secure against attacks of \$25 billion.[4]

---

[4] A 2016 blog post by Ethereum founder Vitalik Buterin contains an informal statement of the flow-stock problem: "The size of the mining network has to be so large that attacks are inconceivable. Attackers of size less than X are discouraged from appearing by having the network constantly spend X every single day. I reject this logic because (i) it kills trees, and (ii) it fails to realize the cypherpunk spirit." This observation, in my view, deserves a share of the credit for the flow-stock idea even though it is informal and incomplete. See Section 3.3 for the full quotation

Figure 1: Comparison of Security Models: Bitcoin versus Traditional



| Traditional Security | | Bitcoin Security |
|---|---|---|
| cost of overcoming guards + | | |
| cost of overcoming police reinforcements + | $> V_{attack}$ | cost of overcoming guards $> V_{attack}$ |
| risk × punishment if caught | | |

The essential difference between the Nakamoto trust model and the traditional model grounded in rule of law is depicted in Figure 1. A criminal is thinking of robbing a bank. In the traditional model, the criminal must first consider how many security guards he will need to overcome. Then he will have to take into account that the bank will call in reinforcements from police, and that, if caught, he will go to jail. Similarly, consider a country thinking of whether to invade another country. They will have to consider the soldiers at the border (analog of security guards), but also that the invaded country will call in military reinforcements (analog of police), and that the invaded country may launch a counter-attack in retaliation (analog of Beckerian deterrence from courts).[5] In contrast, the Nakamoto model is just to have a very large number of security guards at the bank, or soldiers at the border. This works, but it is very expensive and scales terribly with the stakes.

Notice two sources of scale economies in the traditional model. First, the police do not have to be present at the particular bank to provide security—they can provide security to a large number of locations at once as long as they are not all attacked simultaneously. Second, the courts can deter crime with just the credible threat to prosecute and imprison—a fixed cost investment in court capacity can deter a large quantity of potential criminal activity. This is the essence of Becker's (1968) model of optimal deterrence, and is central to Hayek's (1960) resolution of the paradox noted above that freedom requires a government with the power to coerce.[6]

---

and how it can be translated into this paper's model.

[5] I thank Edward Glaeser for drawing this connection to military strategy and Vipin Narang for a helpful discussion about the topic.

[6] Hayek's resolution is that a government's credible threat of coercion, in response to violations of clear, predictable, and symmetrically enforced laws, is both (i) not a violation of freedom, and (ii) sufficient to secure freedom. "The *threat* of coercion has a very different effect from that of *actual and unavoidable* coercion … The great ma-

Notice as well a subtle additional source of inefficiency in the Nakamoto trust model—the full scale of the computational power is present for all transactions, whether for large sums or small. This is like having the same number of security guards outside the local bank branch as outside Fort Knox or the Federal Reserve.

There are many other sources of trust that are familiar to economists besides rule of law, such as reputations, brands, relationships, collateral, and cultural or organizational norms.[7] The contrast versus Nakamoto trust is similar: in each of these cases the trust is more secure than the flow level of investment in maintaining the trust. It also bears emphasis that these sources of trust often work in conjunction with rule of law, sometimes implicitly. For example, a customer trusts Starbucks to provide good coffee because of its brand, but also because it is illegal for a different entity to impersonate Starbucks' name and imagery. In a Levin (2003) relational contract, the employee trusts that if they put in high effort they will get paid a performance bonus, but in the background, it is also the case that the employee knows the employer will pay at least the promised minimum because of rule of law, and the employer trusts the employee not to rob the company because of the rule of law.[8]

There are also many alternative models for creating data security that are familiar to computer scientists and related experts. This includes traditional cryptography. To attack the Nakamoto blockchain requires that the attacker has more computing power than the honest miners; to attack data that is secured by traditional cryptography requires more computing power than a trillion Amazon Web Services, run for more time than the age of the entire universe.[9]

---

jority of the threats to coercion that a free society must employ are of this avoidable kind ... The sanctions of the law are designed only to prevent a person from doing certain things or to make him perform obligations that he has voluntarily incurred. ... Provided that I know beforehand ... I need never be coerced." (Pgs 209-210, emphasis added). "It is the cases that never come before the courts, not those that do, that are the measure of the certainty of the rule of law." (Pg. 316). "There is little difference between the knowledge that if he builds a bonfire on the floor of his living room his house will burn down, and the knowledge that if he sets his neighbor's house on fire he will find himself in jail. Like the laws of nature the laws of the state provide fixed features in the environment in which he has to move." (Pg. 221)

[7] Foundational work on trust from rule of law includes Schelling (1960), Becker (1968), Hart (1995), La Porta et al. (1998). Important work on other sources of trust includes Nelson (1974), Kreps et al. (1982), Fudenberg, Levine and Maskin (1994), Tadelis (1999) on brands and reputations; Baker, Gibbons and Murphy (2002), Levin (2003) on relationships; Kandori (1992), Holmstrom and Milgrom (1994), La Porta et al. (1997), Guiso, Sapienza and Zingales (2006) on norms. Also closely related is work on trust specifically in the context of financial markets, including Shleifer and Vishny (1997), La Porta et al. (1998), Sapienza and Zingales (2012), Gennaioli, Shleifer and Vishny (2015) and Zingales (2015).

[8] Formally, in Levin (2003)'s model, the employer pays the employee at least the fixed salary $w_t$ no matter what, and the employee's lowest action, denoted $e_t = 0$, harms the firm only through poor effort, not theft. If the firm could pay the worker nothing or the worker could rob the firm, the scope for cooperation in the relational contract would be far worse (formally, each party's "renege" option would be much more attractive, undermining the relational contract's ability to be self-enforcing.)

[9] Bitcoin's current level of computing power is about $3 \times 10^{20}$ hashes per second. As I will discuss below, this level of computing power has a flow cost of about \$40 million per day and requires about \$10 billion of specialized capital. To break an SHA-256 encrypted data set through brute force would require $2^{256} \approx 10^{77}$ calculations. I

There is a way out of my flow-stock argument that may explain why major cryptocurrencies such as Bitcoin and Ethereum have not already been attacked given this paper's analysis. If both (i) the technology used to maintain the blockchain is specialized (as opposed to repurposable), and (ii) an attack not only allows the attacker to steal money but also causes a collapse in the value of the cryptocurrency, then the attacker's cost becomes a stock, not a flow, because their specialized capital will collapse in value too.[10] However, vulnerability to collapse is itself a serious problem, and raises the possibility of an attack motivated by this collapse per se ("sabotage"). This part of the analysis thus suggests a "pick your poison" critique of Nakamoto trust: it is either extremely expensive relative to its economic usefulness, or it is vulnerable to sabotage and collapse.[11] The model also identifies specific collapse scenarios.

The remainder of this paper is organized as follows. Section 2 provides a description of the Nakamoto (2008) blockchain. Section 3 presents the heart of the economic critique of Nakamoto, equations (1)-(3). Section 4 uses the model to quantify the cost of keeping Nakamoto trust secure against double-spending attacks. Section 5 considers the possibility of a "sabotage" attack, derives a pick your poison result, and considers collapse scenarios. Section 6 contrasts Nakamoto trust with traditional trust grounded in rule of law. Section 7 concludes. Appendix A discusses responses to this paper's argument since it first circulated in 2018. Appendix B provides technical results in support of the double-spending attack analysis. Appendix C compiles lists of cryptocurrency majority attacks, thefts of cryptocurrency financial institutions, and collapses of cryptocurrency financial instutions.

---

estimate that if you had a trillion Amazon Web Services' worth of compute power (about \$65 billion trillion of capital), running for 14 billion years, that would get you to about $10^{45}$ hashes.

[10] Ethereum's recent adoption of proof-of-stake consensus with "slashing", which is the confiscation of a participant's capital under pre-defined conditions, is an attempt to make a double-spending attack cost a stock not a flow without needing the whole system to a collapse (see Buterin, 2016). Unfortunately, this approach too bumps up against serious economic limits. Most significantly, an impossibility theorem of Tas et al. (2023) shows that it is impossible to guarantee that a protocol can confiscate an attacker's capital before they spend it elsewhere, which defeats the purpose. Work in progress of Lewis-Pye, Roughgarden and Budish (2023) derives a possibility result for this kind of security assuming both a strong network reliability assumption and that the attacker is smaller than 5/9 of the total stake; if the attacker is large enough then the Tas et al. (2023) impossibility obtains and rule of law or some other external source of trust is needed for security. See discussion in Section 2.5.3 and Appendix A.1.

[11] Work in progress of Lewis-Pye, Roughgarden and Budish (2023) shows that the pick your poison critique of Nakamoto trust generalizes to the class of "dynamically available" permissionless consensus protocols. Dynamic availability is a technical condition in the distributed consensus literature that is somewhat like a free-entry / free-exit condition.

# 2 Overview of the Nakamoto Blockchain

Sections 2.1-2.4 provide an overview of Bitcoin and the Nakamoto (2008) blockchain. The goal is to provide an overview that is self-contained and at a sufficient level of detail to justify the economics analysis in the rest of the paper.[12] Section 2.5 clarifies the relationship between the Nakamoto blockchain and three other ideas: permissioned blockchains, smart contracts, and proof-of-stake consensus. Readers already familiar with the relevant background may skip this section without much loss.

## 2.1 Transactions

The first step in describing Bitcoin and the Nakamoto blockchain is to describe transactions, and the limitations of other methods of keeping track of transactions.

**Elements of a Bitcoin Transaction.** The key elements of a transaction are the sender of funds, the receiver of funds, the transaction amount, and a cryptographic signature. The sender and receiver are represented as alphanumeric strings called addresses; addresses are somewhat analogous to account numbers. The cryptographic signature uses standard ideas from public-key cryptography to prove that the transaction was initiated by the sender; that is, the signature could only be created by someone who knows the sender's private key for that address. The cryptographic signature also encodes the other transaction details, including the receiver and the transaction amount; it is like not only signing a check but also signing the seal of the envelope that contains the check, so the recipient and amount cannot be subsequently altered.

**Limitations of a Shared Public Spreadsheet of Transactions.** Imagine keeping track of such transactions on a shared public spreadsheet, such as a Google Doc. The cryptographic signature provides a certain level of trust in the data, in that only Alice, or someone in possession of Alice's private key, can add correctly-signed transactions in which Alice is the sender of funds. However, there are three vulnerabilities:

1. Alice could add a transaction in which she sends money she does not have.
2. Alice could add multiple transactions at the same or similar time, in which she sends money she does have but to multiple parties at the same time.

---

[12] Readers interested in additional computer science detail should consult sources such as the textbook treatment of Narayanan et al. (2016), the website Bitcoin.Org (especially its Bitcoin Developer Guide), Tim Roughgarden's (2023) online course, and the original Nakamoto (2008) paper. Lewis-Pye and Roughgarden (2023) survey the computer science literature on permissionless consensus more broadly. There are several surveys aimed specifically at economists including Halaburda et al. (2022) and Böhme et al. (2015).

3. Alice could delete previous transactions from the shared public spreadsheet; either her own or others'.

Thus, while a shared public spreadsheet of transactions could be utilized among parties that trust each other — e.g., a modern version of the babysitting co-op parable in Krugman (1998) — this system is not suitable for tracking transactions among parties that do not have such a level of trust.

**Limitations of a Trusted Party.** Imagine keeping track of transactions through a widely trusted party that keeps track of balances, such as a central bank. This approach addresses the three vulnerabilities described above with respect to the shared public spreadsheet: the trusted party can ensure that only valid transactions are added to the ledger and that previous transactions are not deleted. However, the limitation is that it requires such a trusted party. The central goal of Nakamoto (2008) is to have a trusted ledger of transactions that does not require any specific trusted party.

## 2.2 What is the Nakamoto Blockchain?

This section describes the Nakamoto blockchain in four steps.

**I: Pending Transactions List.** Users submit transactions to a pending transactions list, called the mempool. One can think of the mempool as in essence the shared public spreadsheet discussed above. However, transactions in the mempool are not considered official yet.

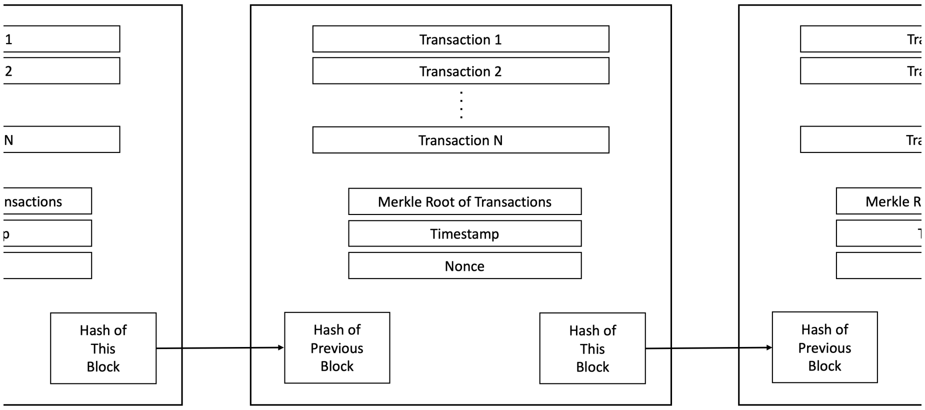
**II: Valid Blocks.** Any computer around the world can compete for the right to add transactions from the mempool to a data structure called the *blockchain*. The computational competition will be described in the next step.

The phrase blockchain references that transactions are added in blocks (for Bitcoin, consisting of about 1000-2000 transactions), and each block of transactions "chains" to the previous block by including a hash of the data in the previous block. See Figure 2. This use of hashes to chain together a sequence of blocks of data was invented by Haber and Stornetta (1991) and Bayer, Haber and Stornetta (1993). Since the hash of the current block depends on the data in the previous block, which in turn includes its hash of the block before that, etc., any change to any element in the history of transactions affects the value of the hash of the current block.

For a block of transactions to be valid, the following three criteria must all be true:

1. Each individual transaction must be properly signed: the cryptographic signature could only be generated by a user in possession of the sender's private key.

Figure 2: Illustration of the Blockchain Data Structure



*Notes:* See the text of Section 2.1 for a description of transactions and the text of Section 2.2 for a description of the overall blockchain data structure and the other elements in the diagram.

2. Each individual transaction must be properly funded: given all transactions in previous blocks in the chain, the sender must be in possession of the Bitcoins she or he is sending.
3. The transactions in a block must not contradict each other: there cannot be two or more transactions in a block in which a common sender sends the same Bitcoins to multiple receivers.

**III: Bitcoin Mining Computational Tournament.** The competition to add new blocks boils down to a massive, brute-force search for a lucky random alphanumeric string. More precisely, Bitcoin miners — where "miners" is just the terminology for computational power that attempts to add new blocks of transactions to the Bitcoin blockchain — choose a valid block of Bitcoin transactions from the mempool that they wish to chain to the previous block of transactions, and search for an alphanumeric string (called a nonce) such that when that alphanumeric string, in combination with all of the data in the new block of transactions they are adding (summarized by its Merkle Root), and the hash of the previous block of transactions that they are chaining to, is all hashed together using the hash function SHA-256, the result has a very large number of leading zeros.

For readers unfamiliar with hash functions, it is highly recommended to go to a website like https://www.movable-type.co.uk/scripts/sha256.html to get a feel for how they work. For example, the hash of the title of this paper is 09b23bf1eb4b7cda... which has one leading zero. A block added to the Bitcoin blockchain in April 2022, block 729,999 has the hash

0000000000000000008b6f6fb83f8d74512ef1e0af29e642dd20daddd7d318f which has 19 leading zeros. Since each digit in the hash can take on values 0-9 and a-f, and the SHA-256 hash function is pseudorandom, the likelihood of finding an alphanumeric string that produces a hash with 19 leading zeros is 1 out of $16^{19}$, which is about 1 out of 75 billion trillion. The number of leading zeros required is calibrated by the Bitcoin system every roughly two weeks, based on the current amount of computational power devoted to Bitcoin mining, to ensure that blocks are successfully mined on average every 10 minutes. (This calibration can be finer than is possible using just zeros; for instance the hash might have to have 19 leading zeros and a 20th digit weakly less than 9.) As of this writing, the amount of computational power devoted to Bitcoin mining is about 375 million trillion hashes per second.

When a miner finds a lucky alphanumeric string, they publicly broadcast their block — consisting of the transactions, the hash of the previous block, their lucky alphanumeric string, and their block's hash — to all of the other Bitcoin miners. Other Bitcoin miners can quickly check whether the block is valid; that is, does the set of transactions in the block meet the criteria listed above in Step II, and does the alphanumeric string indeed produce a valid hash with enough leading zeros. Note, critically, that while finding a lucky alphanumeric string is extremely computationally intensive, checking the validity of a given block is computationally trivial. For this reason, a valid block is "proof-of-work" — proof that the miner who found the block did a large amount of computational work in expectation.

The lucky miner who broadcast the valid block gets compensated in two ways. First, the miner is compensated with new Bitcoins. This is called the "block reward", which was originally 50 Bitcoins per block, and halves every roughly four years, most recently in May 2020 to 6.25 Bitcoins per block. Second, the miner earns transactions fees associated with the transactions they included in their block. The economics of these transactions fees are considered in depth in Huberman, Leshno and Moallemi (2021); users who place a high value on getting their transaction added to the blockchain quickly can ensure faster service by offering a larger transaction fee, so there is an auction-theoretic flavor to the fees, as well as queueing and congestion issues.

**IV: Longest-Chain Convention.** Once a valid block is broadcast and the other miners have checked its validity, miners are supposed to move on to mining the next block. To induce this behavior, Nakamoto proposed the *longest-chain convention* — the convention that, if there are multiple chains of blocks, the longest chain, as measured by the amount of computational work, is the official consensus record of transactions.

Intuitively, Nakamoto's longest-chain convention provides a decentralized way to coordinate miners' efforts. If miners focus their attention on the current longest chain, and they find a lucky alphanumeric string and mine a block, then their new block will be part of the new longest chain, and hence new official record, and the miner will earn the block reward. Nakamoto (2008) shows formally that as long as a majority of computational power follows the longest-chain convention, then the longest chain will outpace attackers with probability that converges to one exponentially in the honest-majority's share and the deficit the attacker must overcome.

A related intuition is that the longest-chain convention provides a decentralized way to adjudicate disputes — computational power "votes" on the true state, and the majority rules.

The game-theoretic validity of longest-chain consensus has received considerable academic attention. The most general treatment to date is Biais et al. (2019), who show that honest mining on the longest chain is indeed a Nash equilibrium, though there can be other equilibria as well. Carlsten et al. (2016) show that longest-chain mining is an equilibrium only if the block reward component of miner compensation is large enough. Kroll, Davey and Felten (2013) provide credible intuition for why longest-chain mining is a Nash equilibrium, though without a formal game-theoretic model.

However, all of these prior works explicitly assume that all miners are "small" — that is, they assume away the possibility of majority attack. Majority attack, discussed next, will be at the heart of this paper's analysis.

## 2.3 Vulnerability to Majority Attack

Nakamoto's blockchain is vulnerable to attack by an adversary with 51% or more of the computational power. This is because the adversary, whenever they like, can create an alternative chain of blocks that will outpace the honest chain of blocks with probability one, and hence become the new consensus. This vulnerability of Nakamoto consensus is widely understood — it is even in the abstract of the Nakamoto (2008) paper (excerpted below). Moreover, that Nakamoto consensus is vulnerable to attack is not surprising to computer scientists in the sense that previous approaches to distributed consensus were also vulnerable to attacks by a too-large adversary. For example, the Byzantine Fault Tolerance (BFT) paradigm for consensus has been known to be vulnerable to attack by an adversary with $> \frac{1}{3}$ of the power since the 1980's, except under very restrictive assumptions (Pease, Shostak and Lamport 1980; Lamport, Shostak and Pease 1982; Dolev and Strong 1983; Fischer, Lynch and Paterson 1985; Dwork, Lynch and Stockmeyer 1988).[13]

---

[13] The exception in which communication among honest parties is secure even in the presence of an unbounded adversary requires that the honest parties have access to a communication network that never experiences delays longer than a known bound (the "synchronous model") and have access in advance of communication to all honest parties' cryptographic public keys (the "public key infrastructure" assumption). These assumptions are frequently satisfied in practical applications with pre-existing trust (e.g., secure military communications) but are widely viewed to be incompatible with the kind of communication Nakamoto (2008) is trying to facilitate over the internet among parties without pre-existing trust. See Lecture 3 of Roughgarden (2023) for an accessible treatment and the papers cited in the text for some key historical results on BFT consensus.

The canonical attack Nakamoto (2008) worried about is called a double-spend: the attacker sends Bitcoins in transactions on the original honest chain, and then deletes those transactions from the consensus record with their alternative chain, allowing them to spend the same currency twice. Section 4 will describe double-spending attacks in detail and analyze their economic implications.

Eyal and Sirer (2014) show that Bitcoin is also vulnerable to a form of minority attack, in which a large-enough miner can sometimes profit, in expectation, from holding back a solved block so that they can work on extending it in private, while other miners therefore focus their attention on what is probabilistically not the longest chain. However, the purpose of the Eyal and Sirer (2014) minority attack is more circumscribed in that its goal is to obtain a disproportionate share of mining rewards, not to manipulate the blockchain to double spend.

## 2.4 Nakamoto Blockchain: Summary

The abstract of Nakamoto (2008) succinctly summarizes the accomplishment and its vulnerability:

> "A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. 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.* The network itself requires minimal structure. Messages are broadcast on a best effort basis and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone." (Emphasis added)

The accomplishment is a "purely peer-to-peer version of electronic cash" without the use of a "trusted third party." Trust in the integrity of the data emerges from the hash-based proof-of-work, conducted by an unstructured network with free entry and exit. The longest chain is the official record of "what happened" — i.e., is the (permissionless) consensus.

The vulnerability is majority attack — the construction relies on the assumption that "a majority of CPU power is controlled by nodes that are not cooperating to attack the network."

The economic limits that majority attack places on Nakamoto's novel form of trust are at the heart of this paper's analysis.

## 2.5 Clarifications and Discussion

### 2.5.1 Permissioned Blockchains

As interest in Bitcoin and Nakamoto's blockchain surged, many started to use the phrase "blockchain" to describe similarly-architected databases maintained by *known, trusted parties* — that is, *without* the central innovation of Nakamoto (2008). This concept is sometimes known as a permissioned or private blockchain, or sometimes as distributed ledger technology (see, e.g., Bakos and Halaburda, 2021). An IBM marketing campaign called it "Blockchain for Business." Goldman Sachs called such blockchains "The New Technology of Trust." (Goldman Sachs, 2018)

Many researchers and observers view this use of the phrase "blockchain" as hype for what is in essence just an append-only distributed database with well-defined permissions, and with cryptography to protect against data tampering as in Haber and Stornetta (1991). The financial columnist Matt Levine memorably wrote:

> "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." (Levine, 2017)

As should be clear, this paper's critique is of blockchain in the sense of Nakamoto (2008), not of distributed databases with trust grounded in traditional sources. It should be uncontroversial that well-architected databases are economically useful, even if there is debate about what to call them. See Budish and Sunderam (2023) for further discussion in the context of traditional finance.

### 2.5.2 Smart Contracts

Notice that Nakamoto's novel form of trust is not specific to currency transactions. One can replace "Alice sends Bob 10 Bitcoins, signed by Alice" with any executable computer instruction signed by Alice. This idea is often called a "smart contract" (see Buterin, 2014*a*).

The analysis framework of this paper applies analogously to blockchains that allow smart contracts though the attack possibilities may differ.

### 2.5.3 Proof-of-Stake

The computational work Bitcoin miners must perform to add a new block serves the role of sybil resistance, i.e., making it expensive to add new identities to the permissionless system. Without sybil resistance an attacker could create infinitely many identities.

Since Nakamoto (2008) there have been several other approaches taken to sybil resistance for permissionless consensus, the most prominent of which is proof-of-stake. Roughly, instead of voting for the correct chain with computational work, participants vote for the correct chain with stake in the cryptocurrency. Ethereum, the second-most valuable cryptocurrency project after Bitcoin, switched from proof-of-work to proof-of-stake in Fall 2022, and its founder has been discussing the potential benefits of proof-of-stake since as early as 2014 (Buterin, 2014*b*, 2016).

One motivation for proof-of-stake over proof-of-work is to reduce environmental externalities. The computational work that powers Bitcoin consumes on the order of 0.3-0.8% of all global electricity, which is a fairly astonishing figure.[14] As will become clear, however, the environmental issue is orthogonal to the concerns raised in this paper about Nakamoto (2008) — just replace the opportunity cost of mining with the opportunity cost of holding stake and the arguments go through relatively unchanged.

What is interesting about proof-of-stake for the purpose of this paper's argument is that stakes are not memory-less like computational work: stakes are locked up on chain, like collateral, and observably persist over time, like reputation. This opens up the possibility of punishing attackers by confiscating their stakes, which makes attacks more expensive and hence the blockchain more secure. Intuitively, this is an attempt to algorithmically mimic the traditional trust that is created by law in combination with financial collateral (Buterin, 2014*b*, 2016; Buterin and Griffith, 2019).

Recent research suggests that this approach to security, while intuitively compelling, may not work. Tas et al. (2023) show as their Theorem 1 that it is impossible to guarantee that a large-enough attacker can be successfully slashed by any positive amount before the attacker withdraws their stake. Lewis-Pye, Roughgarden and Budish (2023) derive a possibility result for slashing the attacker's stake without punishing honest participants, but the result requires both a strong assumption about the nature of the networking environment and the assumption that the attacker's majority is smaller than 5/9 of the total locked-up stake. If the attacker is too large then the attacker can circumvent the punishment; a rough intuition is that a large-enough attacker controls the protocol's legal system.[15] An interpretation of these results is that a proof-of-stake

---

[14] De Vries (2018); Digiconomist (2022). The 0.8% figure is Digiconomist (2022)'s main estimate, whereas the 0.3% figure is based on its best-case analysis under the assumption that all Bitcoin mining equipment is maximally energy efficient.

[15] The network reliability assumption in Lewis-Pye, Roughgarden and Budish (2023) is that it is possible to impose a delay period between when a user (including honest users) asks to unlock their stake and when they

blockchain can successfully mimic traditional trust to deter small attacks, but that rule of law must step in in the case of a large-enough attacker. See discussion of related open questions in the Conclusion and further discussion of proof-of-stake in Appendix A.1.

# 3 Nakamoto Blockchain: A Critique in Three Equations

Sections 3.1-3.3 present the three equation critique of Nakamoto (2008). Section 3.4 presents a result that shows that the net cost of attack may be *zero* under strong assumptions. Section 3.5 presents a one-shot game version of the analysis that captures the essence of the critique while abstracting from many details. This one-shot version will be helpful for the comparison to traditional trust later in the paper and may be simpler to teach.

## 3.1 Zero-Profit Condition (Honest Play)

Our conceptual question here is: how much computational power will maintain Nakamoto's anonymous, decentralized trust, if we restrict all participants to behave honestly?

Let there be a large finite number $I$ of honest participants, who follow the Nakamoto longest-chain protocol automatically. We may think of $I$ as representing all people who could potentially provide part of the decentralized support for Nakamoto trust. For example, $I$ is the number of people connected to the internet around the world.

Each player $i$ chooses a quantity of "trust support" $x_i \in \mathbb{R}^+$, which we may think of as their quantity of computational work in Nakamoto (2008)'s proof-of-work blockchain, or their quantity of some other costly action in another blockchain (stake, storage space, memory, etc.). Let $N = \sum_{i=1}^{I} x_i$ denote the total quantity of trust support. A player can choose a quantity of zero if they like, which is how we can think about people not participating in the decentralized trust. Our equilibrium concept for $N$ will be a zero-profit condition. This is meant to capture the permissionless, free-entry / free-exit nature of Nakamoto trust. Nash equilibrium is studied in the one-shot game analysis of Section 3.5 and is very similar.

Let $c$ denote the cost per unit time to supply one unit of trust support. For example, for proof-of-work, this is the cost per unit time to run one unit of computational power, including variable costs such as electricity and a rental cost of capital for capital equipment. We will sometimes

---

can use it in a transaction, such that the delay period is longer than any feasible attack. Otherwise the attacker could withdraw their stake and spend it before their attack is detected and punished, as in Tas et al. (2023). This implies that, if a proof-of-stake cryptocurrency were to become economically useful (to date, this remains an open question, see Buterin, 2022), then only a fraction of the total stake can be locked up for trust support, with the rest unlocked for actual use.

use the notation $c = rC + \eta$, where $rC$ is the rental cost of capital and $\eta$ is the variable cost of electricity.

Let $p_{block}$ denote the economic reward paid to a participant who successfully mines a new block of transactions, i.e., wins a computational tournament. For the purpose of this paper, we will consider the compensation to the lucky miner in aggregate, without distinguishing between whether this compensation is in the form of newly issued Bitcoins (which are a form of seignorage tax on holders of the currency) or transaction fees. We will treat $p_{block}$ as exogenous and derive constraints on it below. Participants' probability of winning the next reward $p_{block}$ is equal to their share of trust support. Specifically, player $i$ wins the next block with probability $\frac{x_i}{N}$.

Let $D$ denote the block difficulty level, defined as the number of units of trust-support-time needed, in expectation, to mine one block. Assume blocks arrive Poisson. That is, if there are $N$ units of trust support, blocks are solved according to a Poisson point process with mean $\frac{D}{N}$.

Note a potential source of confusion is that costs $c$ are incurred per unit time whereas rewards $p_{block}$ are earned per block. The next two concepts will help map between objects that are per unit time and objects that are per block. First, we can define profits per unit of trust support per unit time as

$$\frac{1}{N}\frac{D}{N}p_{block} - c,$$

because some unit of trust support solves a block every $\frac{D}{N}$ time in expectation and each of the $N$ units is equally likely to be the winner.

Second, define honest equilibrium as follows:

**Definition 1.** A *zero-profit honest mining equilibrium* consists of quantities $\{x_i^*\}_{i=1}^{I}$ and a difficulty level $D^*$ such that participants (i) solve one block per unit time (as a normalization), and (ii) earn zero economic profits in expectation.

**Proposition 1.** *Let* $N^* = \sum_{i=1}^{I} x_i^*$. *In any zero-profit honest mining equilibrium,*

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

*and* $D^* = N^*$.

*Proof.* For participants to solve one block per unit time (condition (i)) requires $D^* = N^*$. This in turn implies that profits per unit of trust support per unit time are $\frac{1}{N}\frac{D}{N}p_{block} - c = \frac{1}{N^*}p_{block} - c$. For these profits to be zero (condition (ii)) in turn implies $N^* c = p_{block}$. □

Proposition 1 is widely known and is the standard characterization of a rent-seeking tournament: the prize in the tournament, $p_{block}$, is dissipated by expenditures aimed at winning the prize, $N^* c$.[16] Prat and Walter (2021) provide empirical support that equation (1) describes actual equilibrium behavior in the Bitcoin mining market, with some additional nuances related to capital adjustment costs. There are also numerous websites that compare current block rewards to current mining costs, and the Bitcoin Wiki discusses the free-entry / zero-profits logic in detail (Bitcoin Wiki, 2020*a*), both of which lend further support to equation (1).

For a sense of magnitudes, in 2021 Bitcoin's block reward averaged roughly $318,000 per 10 minute block, which corresponds to about $2 million per hour, $46 million per day, and about $17 billion per year. Ethereum's block reward in 2021 averaged about $8,000 per block, but since Ethereum's block interval is just 13 seconds on average this corresponds to a similar overall magnitude, of about $50 million per day and about $19 billion per year.

## 3.2 Incentive Compatibilty Condition (Majority Attack)

Our conceptual question here is: how much security is generated by the amount of honest trust-support characterized in equation (1)? As discussed in Section 2.3, it is widely understood that an agent with a majority of computational power could successfully attack Nakamoto's novel form of trust. Specifically, such a player could double spend with probability one.

If there are $N^*$ units of honest trust support, an outsider gains a majority with $N^* + \epsilon$ units, at cost $(N^* + \epsilon)c$ per unit time. An insider gains a majority with as little as $\frac{N^*}{2} + \epsilon$ units of computational power. The analysis will mostly focus on the costs of outside attacks, both to be conservative and because that case most cleanly expresses the conceptual critique of Nakamoto's novel form of trust. That said, readers thinking about the logistics of majority attacks in practice may find insider attacks more worrisome; they are also cheaper.

Consider an additional player, the attacker, not restricted to honest play. This player can attack by choosing $AN^*$ units of trust support, $A > 1$, for an $\frac{A}{A+1}$ majority at cost $AN^*c$ per unit time. Denote the expected duration of the attack by $t(A)$; we will derive a closed form for $t(A)$ under some assumptions in Section 4. Call $AN^*c \cdot t(A)$ the gross cost of attack. The attacker can choose $A$ to minimize $A \cdot t(A)$. Call this optimum $A^* \cdot t(A^*)$; Appendix B studies this optimum numerically.

Let $V_{attack}$ denote the value of attack. For now, let us think about this value of attack in the abstract, but have in mind that the value of an attack will grow if the blockchain's economic

---

[16] See, for example: Kroll, Davey and Felten (2013) pg. 8; Huberman, Leshno and Moallemi (2021) Theorem 1; Easley, O'Hara and Basu (2019) equation (2); Chiu and Koeppl (2022) Lemma 1; Ma, Gans and Tourky (2018) equation (7); and Halaburda et al. (2022) equation (4). It is also straightforward to allow for heterogeneous mining costs. Let $c(\cdot)$ denote a continuous weakly increasing function where $c(n)$ gives the per-block cost of the $n$th unit of computational power. Then (1) becomes $N^* c(N^*) = p_{block}$. The marginal unit of computational power earns zero economic profits.

usefulness and importance grow.

**Definition 2.** The Nakamoto blockchain is *incentive compatible against an outsider attack, on a gross-cost basis*, if the gross cost of attack exceeds the benefits of attack:

$$A^* N^* c \cdot t(A^*) > V_{attack}. \tag{2}$$

Two brief remarks are required. First, as noted, an attack could also come from an insider, i.e., part of the current honest trust support. The outside attacker IC condition (2) seems more attactive conceptually, because it treats the honest participants as small and dispersed per the Nakamoto ideal and in equilibrium with the level of compensation $p_{block}$. That said, an inside attacker might be more realistic in practice, since mining is often concentrated (Makarov and Schoar, 2021; Cong, He and Li, 2021). Second, the left-hand-side of (2) is the gross cost of attack. In the Nakamoto blockchain, the attacker would earn rewards for the blocks in their attacker chain, which subsidizes the attacker. We will come back to the distinction between gross and net costs of attack in Section 3.4.

## 3.3 Flow-Stock Problem

In the hoped-for equilibrium in which participants are honest, the amount of trust support devoted to maintaining the blockchain is characterized by the zero-profit equilibrium (1).[17] The incentive-compatibility condition then relates this amount of trust support to the level of security generated. Since $N^* c$ appears in both the zero-profit condition (1) and the incentive-compatibility condition (2), we can combine the two equations:

**Proposition 2.** *(Flow-Stock Problem) The zero-profit condition (1) and the incentive-compatibility condition (2) together imply the equilibrium constraint:*

$$p_{block} > \frac{V_{attack}}{A^* \cdot t(A^*)} \tag{3}$$

*In words: the equilibrium per-block payment to participants for maintaining the trust on the blockchain must be large relative to the benefits of attacking it.*

*Proof.* (3) follows directly from combining (1) and (2). □

Equation (3) places potentially serious economic constraints on the applicability of the Nakamoto (2008) blockchain. The blockchain can only be used in economic contexts where users are willing

---

[17] If miners earn positive economic profits in equilibrium then the quantity $N^*$ characterized by the zero-profit condition in (1) is an upper bound, and (3) obtains as is.

to pay a per-block transactions cost, $p_{block}$, that is large relative to the value of attacking the system, $V_{attack}$. The "flow" payment for trust support must be large relative to the "stock"-like value of attack.

Economically, this is a very expensive form of trust. In contrast, consider, e.g., mutually-beneficial cooperation in a relationship and the associated temptation to cheat, or a trusted brand that is tempted to shirk on quality. In such cases, the cost of cheating to the cheating party is related to the *stock* value of the relationship or brand they are destroying, not the flow cost of its maintenance.[18] Imagine if a brand were only as trustworthy as its flow investment in advertising, or users of the Visa network had to pay fees to Visa, every ten minutes, that were large relative to the value of a successful attack on the Visa network. Or, imagine that a country were only as secure as its flow expenditure on soldiers at the border.

Another contrast is trust that is supported by rule-of-law. In such cases, the cost of cheating to the cheating party is related not to the direct costs of conducting the crime, but to the costs of potentially getting caught and punished (Becker, 1968). A government able to credibly impose large punishments (the parameter $f$ in Becker's model) can deter large attacks or crimes at comparatively low cost. As emphasized, the ingenious aspect of the Nakamoto (2008) form of trust is that it is completely anonymous and decentralized, without any reliance on rule-of-law, relationships or other traditional sources. But, this aspect also makes the Nakamoto (2008) form of trust much more expensive.

From a computer security perspective, the key thing to note about (3) is that the security of the blockchain is *linear* in the amount of expenditure on trust support. For instance, if attacking the system grows 1000 times more attractive, then the cost of securing the system must grow 1000 times as well for the system to remain secure. In contrast, in many other contexts investments in computer security yield convex returns (e.g., traditional uses of cryptography) — analogously to how a lock on a door increases the security of a house by more than the cost of the lock. It is much more expensive to break modern cryptography than it is to implement it! Imagine if the cost of attacking Visa was that you had to have as much computational power as Visa for a few hours.

A blog post by Ethereum founder Vitalik Buterin (2016) deserves credit for an early informal statement of the flow-stock problem:

> "Because proof of work security can only come from block rewards, and incentives to miners can only come from the risk of them losing their future block rewards, proof

---

[18] An early version of this insight is due to Schelling (1956): "What makes many agreements enforceable is only the recognition of future opportunities for agreement that will be eliminated if mutual trust is not created and maintained, and whose value outweighs the momentary gain from cheating in the present instance." For additional references and literature connections please see the introduction.

> of work necessarily operates on a logic of massive power incentivized into existence by massive rewards. Recovery from attacks in PoW is very hard: the first time it happens, you can hard fork to change the PoW and thereby render the attacker's ASICs useless, but the second time you no longer have that option, and so the attacker can attack again and again. Hence, the size of the mining network has to be so large that attacks are inconceivable. *Attackers of size less than X are discouraged from appearing by having the network constantly spend X every single day.* I reject this logic because (i) it kills trees, and (ii) it fails to realize the cypherpunk spirit – cost of attack and cost of defense are at a 1:1 ratio, so there is no defender's advantage." (Emphasis added).

Buterin's language can be translated into the formal analysis of this paper by dollar-izing both uses of X: "Attackers of size less than X" can be interpreted as attackers with an attack opportunity worth less than $V_{attack}$ (not as the size of the attacker's computational power, else the two X's are not comparable). "Having the network constantly spend X every single day" can be interpreted as assuming that an attack takes one day, i.e., $A^* \cdot t(A^*)$ is one day worth of block-compute costs, and then requiring for security that $A^* \cdot t(A^*) \cdot p_{block} > V_{attack}$, i.e., equation (3).

## 3.4 Zero Net Attack Cost Theorem

What we may call the *net* cost of attack can differ from the gross cost of attack, modeled above, for three potential reasons: block rewards, attacker cost frictions, and effects of the attack on the value of Bitcoin itself.

First, the attacker earns block rewards from their attack. That is, after the attacker's alternative chain replaces the honest chain, the attacker earns the block rewards associated with the blocks in the new longest chain. These block rewards in effect subsidize the attack. An $A$ attacker who attacks for $t$ time units performs $At \cdot N^*$ units of trust support. If the difficulty stays constant at $D' = D^* = N^*$, then this corresponds to $At$ block rewards in expectation. If the difficulty on the attacker chain adjusts upwards, i.e., $D' > D^*$, then the attacker will earn $At \cdot \frac{N^*}{D'} < At$ block rewards.

Second, the attacker may face frictions relative to the cost of honest mining. For example, if the attacker's compute power is less energy efficient than the honest miners' compute power, or because there are costs of starting and stopping the attack. Let $\kappa \geq 0$ parameterize the attacker's cost inefficiency relative to honest mining, such that their total cost of attack is $(1 + \kappa)At \cdot N^*c$.

Third, the attack may harm the value of Bitcoin. This reduces the value of the attacker's block rewards and reduces the value of the Bitcoin the attacker is left with after double spending. If we let $\Delta_{attack} \geq 0$ parameterize this decline, this reduces the value of the attacker's block rewards by $\Delta_{attack} At \cdot N^*c$ and reduces the benefit of a double spending attack originally worth $V_{attack}$ by $\Delta_{attack} V_{attack}$. If the capital equipment is specific to the attacked cryptocurrency, then the attack would reduce the value of the capital equipment as well; we will return to this issue in Section 5.

In the ideal case for the attacker with respect to these three sources of cost difference, we have the following remarkable conclusion:

**Proposition 3.** *(Zero Net Attack Cost) If the attacker does not face any cost frictions relative to the costs of honest participants ($\kappa = 0$), the attack concludes without any difficulty adjustment ($D' = D^*$), and the attack does not cause the value of Bitcoin to fall ($\Delta_{attack} = 0$), then the net cost of attack is zero.*

*Proof.* The attackers' trust-support cost of attack is $(1+\kappa)At \cdot N^*c$. The net value of the attacker's block rewards is $At \cdot \frac{N^*}{D'} p_{block}(1 - \Delta_{attack})$. The reduction in the value of the Bitcoin the attacker is left with after double spending is $\Delta_{attack} V_{attack}$. If $\kappa = \Delta_{attack} = 0$ and $D' = N^*$, then substituting $p_{block} = N^*c$ from equation (1) yields a net cost of 0. $\square$

The intuition behind this result is that the attacker is fully compensated for their computational costs for the same reason that honest miners are fully compensated for their costs under honest play. In effect, Nakamoto (2008) consensus treats the attacker "as if" they are an honest participant, because the longest chain is the true state.

Moroz et al. (2020), Auer (2019), Tabarrok (2019) and Jacob Leshno (in a communication with the author) derive similar results to Proposition 3 building off of Budish (2018).[19] Bonneau (2016)'s analysis of "bribery" attacks deserves early credit for the intuition that the net cost of attacking Bitcoin might be very small because of the block rewards subsidy, as does a 2017 blog post of Buterin (2017) who stated the idea in a footnote. Recent work of Gans and Halaburda (2023) generalizes the zero net attack cost result and, by incorporating Bitcoin transaction fees into their model, shows that it is possible for an inside attacker to have a slighty negative net attack cost.

To be clear, zero attack frictions seems unrealistic. But, zero friction is often useful as a benchmark case, and the result does reinforce that Nakamoto trust is economically implausible when taken literally.

---

[19] The analysis in the June 2018 draft artificially constrained the attacker to earn at most $t$ block rewards. The June 2018 draft also did not have explicit cost frictions. Rather, the assumption that an attacker earns at most $t$ block rewards is like an implicit cost friction, related to starting and stopping the attack, of $(A-1)t \cdot N^*c$. As a result, the June 2018 draft had slightly different simulated net costs than here, and that draft did not have Proposition 3.

## 3.5 One-Shot Game Analysis

The analysis above uses a price-theoretic zero-profit equilibrium concept for honest mining, and contains several details that are specific to aspects of Nakamoto (2008). As a complement to that approach consider the following stylized one-shot game, which yields a Nash equilibrium solution and abstracts from some complexities of the above.

There are $I$ players. Each player $i$ chooses a quantity $x_i$ of trust support (work, stake, etc.) and a "posture" $a_i \in \{Honest, Attack\}$. Costs are $c$ per unit of trust support. Define $N = \sum_{i=1}^{I} x_i$.

Payoffs are as follows. If there is a player $i$ with $x_i > \frac{N}{2}$ and $a_i = Attack$, this player gets a payoff of $V_{attack}$, gross of their costs. All other players get zero. Else, each player gets a payoff of $\frac{x_i}{N}p$.

Our question is: under what conditions does there exist a Nash equilibrium, denoted $\{(a_i^*, x_i^*)\}_{i=I}^{I}$, in which all players choose $a_i^* = Honest$? Call such a profile, if one exists, an honest equilibrium.

**Lemma 1.** *If there is an honest equilibrium, then $N^*c \leq p$.*

*Proof.* Towards a contradiction, assume there is an honest equilibrium with $N^*c > p$. Choose any player $i$ with $x_i^* > 0$. Player $i$'s net payoff is $\frac{x_i^*}{N^*}p - x_i^*c < x_i^*c - x_i^*c = 0$. So the player has a profitable deviation by choosing $x_i' = 0$ instead. Contradiction. $\square$

In words, this lemma tells us that the amount spent on trust support $N^*c$ will be no greater than the compensation paid for this trust support $p$, analogously to equation (1) above.

**Proposition 4.** *A necessary condition for an honest equilibrium is $p \geq \frac{V_{attack}}{1+\frac{1}{I}}$.*

*Proof.* Conjecture an honest equilibrium. Player $i$'s payoff in honest equilibrium is $\frac{x_i^*}{N^*}p - x_i^*c$. Consider a deviation by $i$ in which they attack by choosing $a_i' = Attack$ and $N_{j\neq i}^* = \sum_{j\neq i} x_i^* + \epsilon$ for some $\epsilon > 0$. For this to be worse for player $i$ requires $V_{attack} - {N^*}_{j\neq i}c - \epsilon \leq \frac{x_i^*}{N^*}p - x_i^*c$. Note that if $x_i^* = 0$ this simplifies within an epsilon to $N^*c \geq V_{attack}$, analogously to (2) above. Rearranging, using the Lemma, and noting that $\min(x_i^*) \leq \frac{1}{I}N^*$ yields $V_{attack} \leq p(1 + \frac{1}{I})$. $\square$

As the number of players $I$ grows large, the necessary condition for honest play becomes

$$p \geq V_{attack},$$

analogously to (3) above. An interpretation of the timing of this game is that $p$ and $c$ now represent, respectively, blockchain compensation and trust-support-costs for an amount of time commensurate with the duration of an attack, i.e., the analog of $A^* \cdot t(A^*)$ in (3). The analysis tells us that the cost of running the blockchain, for an attack-duration amount of time, must exceed the value of attacking it.

# 4 Analysis of Double-Spending Attacks

The canonical attack Nakamoto (2008) worries about is called a "double spend." In a double spend, an attacker sends cryptocurrency to another party (presumably in exchange for goods or assets), and then uses their computational majority to effectively delete the transaction from the blockchain, allowing them to spend their cryptocurrency again later (hence "double").

This section analyzes double-spending attacks under the assumption that the attack does not cause a post-attack decline in the value of the cryptocurrency, i.e., $\Delta_{attack} = 0$. The case where the attack does cause such a decline will be considered in the next section. The analysis will use the gross cost of attack (i.e., equations (2)-(3)), under various assumptions about escrow periods. This is equivalent to using the net cost of attack considered in Section 3.4 under the assumption that attack frictions cancel out block rewards, i.e., $\kappa = 1$.

## 4.1 Mechanics of a Double-Spending Attack

**What a majority attacker can and cannot do.** Before discussing double-spending attacks it is useful to clarify what, technologically, a majority attacker can and cannot do. Because a majority attacker can find lucky hashes faster in expectation than the honest minority, the attacker can create an alternative longest chain of transactions, and replace the honest chain with their alternative chain at a strategically opportune moment. This allows the attacker to control what transactions get added to the blockchain, and allows the attacker, within computational limits, to remove recent transactions from the blockchain — by creating an alternative chain starting from the recent past. The attacker even earns the block rewards for each period of their alternative chain after they make it the new longest public chain.

What the attacker *cannot* do is create new transactions that spend other participants' Bitcoins. Creating new transactions that spend other participants' coins would require not just a majority of computational power, but enough computational power to break modern cryptography: creating a transaction that spends another participant's coins requires learning their private key. A majority attacker cannot simply "steal all the Bitcoins" (Bitcoin Wiki, 2020*b*).

**Description of double spending.** Figure 3 illustrates a double-spending attack. The attacker engages in the following actions in sequence:

(i) The attacker spends Bitcoins. That is, the attacker signs one or more transactions in which they send Bitcoins to other parties in exchange for other goods or assets.

(ii) The attacker allows those transactions to be added to the blockchain. That is, the transactions are added to the longest chain as parts of mined blocks in the usual way as described in Section 2.2.

(iii) The attacker works in secret to create an alternative longest chain. In this alternative chain, the Bitcoins that were sent to other parties in (i) are instead sent to other addresses controlled by the attacker.

(iv) The attacker waits for any escrow periods to elapse, so they receive the goods or assets they transacted for in (i). In Bitcoin a common escrow period is 6 blocks, or about 1 hour.

(v) The attacker then releases their alternative longest chain. The attacker now has the goods or assets they received in (iv) but also has the Bitcoins which they have sent to themselves in the chain in (iii).

## 4.2 Analysis Framework

Equation (3) tells us that the possibility of a double-spending attack places economic constraints on Nakamoto's anonymous, decentralized trust. To understand these constraints we need to analyze the benefits of a double-spending attack (the $V_{attack}$ term) and the expected cost of a double-spending attack in block-compute-cost units (the $A^* \cdot t(A^*)$ term).

Before proceeding with the analysis, I would like to reiterate the "if-then" nature of this paper's argument. This paper is trying to take seriously the "if" possibility in which cryptocurrencies and Nakamoto's anonymous, decentralized trust become a more important and useful part of the global economic and financial system. Some responses to the first draft of this paper's double-spending analysis were about why double-spending attacks would be hard to execute at the scale imagined in this section at *present* — not in the hypothesized future in which cryptocurrencies and Nakamoto trust are much more integrated with the global economy and financial system.

### 4.2.1 Benefits of Double Spending: $V_{attack}$

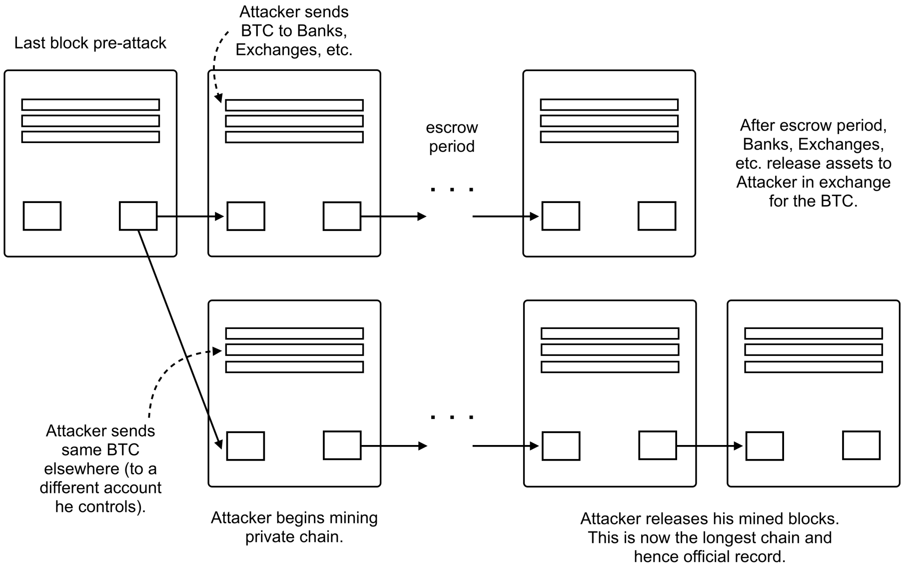
A majority attacker will not use their majority to double spend for a cappucino at Starbucks. They will use their majority to conduct transactions that are as large as possible given the current uses of the Nakamoto blockchain. Furthermore, they might engage in many such transactions using multiple addresses.

$V_{attack}$, therefore, should be understood as a statistic on the amount of transaction volume that large *honest* users of Bitcoin can conduct in a short period of time. The more value that honest users can transact using Bitcoin, the more value an attacker can double spend.[20]

---

[20] This point likely seems obvious, but it was missed in past academic literature on double-spending attacks. The computer science literature did not explicitly model the economic benefits of attack, and therefore missed how they would scale with Bitcoin's usefulness (Rosenfeld, 2014; Eyal and Sirer, 2014; Bonneau, 2016). Within economics, a model of Chiu and Koeppl (2022) assumes that an attack involves just a single transaction and holds this transaction size fixed. The authors conclude that the system becomes more secure as its economic value grows

Figure 3: Illustration of Double-Spending Attack



*Notes:* See the text for description.

Therefore, I will consider a wide range of values for $V_{attack}$. I use \$1,000 as the low-end of this range, representing Bitcoin's early days when even buying a pizza was remarkable. I use \$100 billion as the high-end of this range. While arbitrary, this seems a reasonable order of magnitude for a large-scale attack on the global financial system. This figure also represents about 10% of Bitcoin's peak market capitalization.

### 4.2.2 Cost of Attack in Block-Compute-Cost Units: $A^* \cdot t(A^*)$

It is possible to obtain a closed-form expression for the expected duration $t$ of a double-spending attack.[21] Let $A$ denote the attacker's majority and $e$ denote the escrow period. For simplicity, assume that if the attacker engages in multiple transactions, they are all added to the honest chain at the same time. Motivated by the description of the mining process above, assume that honest miners mine new blocks as a Poisson process with arrival rate 1 and the attacker mines new blocks as a Poisson process with arrival rate $A$. As a reminder, we interpret one unit of time as the amount of time it takes in expectation to mine a single block in honest equilibrium.

In the Appendix I show the following:

---

relative to this fixed transaction size. This is like noting that it is less attractive to engage in a double-spending attack for a cappucino in 2023 than it was in 2009.

[21] Prior work by Grunspan and Perez-Marco (2017) provides a closed-form expression for the probability that the attacker is ahead of the honest miners after the honest miners have mined $z$ blocks.

**Proposition 5.** *The expected duration $t$ of the double-spending attack, as a function of the attacker majority $A$ and escrow period $e$, is given by:*

$$t(A,e) = (1+e) + \left[ \sum_{i=0}^{1+e} \left( \frac{i+1}{A-1} \right) \cdot \frac{(1+2e-i)!}{(1+e-i)!e!} \left( \frac{A}{1+A} \right)^{1+e-i} \left( \frac{1}{1+A} \right)^{1+e} \right]. \tag{4}$$

*As the attacker majority grows large ($A \to \infty$), $t(A,e)$ converges to $1+e$. In the limit as $A \to_+ 1$, we have $t(A,e) \to \infty$.*

*Proof.* See Appendix B. □

Expression (4) can be understood as follows. In the attacker's best case, their attack takes $1+e$ time. That is, as soon as their assets are released from escrow, the attacker releases their alternative longest chain. This best case occurs if the attacker mines $1+e+1$ blocks before the honest miners mine $1+e$ blocks. Suppose, on the other hand, that the attacker is behind the honest chain by $i \geq 0$ blocks at the time the honest miners mine their $1+e$ block. Given the Poisson arrival processes, it will take the attacker $\frac{i+1}{A-1}$ of time in expectation to strictly surpass the honest chain. The last part of the expression gives the probability that the attacker's deficit is $i$ blocks, as a function of the escrow period $e$ and attacker majority $A$.

Table 1 provides example calculations of duration $t$ and the gross block-compute-cost term $At$ for a wide variety of escrow periods and attacker majorities. For example, if the escrow period is $e = 6$, which is a fairly common escrow period for Bitcoin, then attacker majorities ranging from $A = 1.2$ to $A = 1.5$ result in attack durations $t$ ranging from 8.77 to 14.37, and gross block-compute-costs $At$ ranging from 13.15 to 17.24. Notably, smaller majorities lead to significantly longer attack durations and costs. If $A = 1.05$, which corresponds to a 51.2% attacker majority, the duration $t$ is 45.06 blocks and the cost term $At$ is 47.31.

If the escrow period is significantly longer than common practice, say $e = 100$ blocks (roughly 16 hours), then attack durations range from 101.0 to 105.1 for attackers with majorities ranging from $A = 1.2$ to $A = 1.5$, and gross block-compute costs $At$ range from 126.2 to 151.5 for attackers with majorities in this range. Notice that as the escrow period grows longer, the average attack duration $t$ gets proportionally closer to the escrow period. The intuition is simple law-of-large numbers.

Also note that, even at very long escrow periods, the gross-cost-minimizing attacker majority is larger than 51%. For escrow periods ranging between $e = 6$ to $e = 1000$, the cost-minimizing attacker majority ranges from about $A = 1.5$ (60%) to about $A = 1.1$ (52%). It is true that a 51% majority is enough to ensure statistically that the attack will eventually succeed, but a cost-minimizing attacker will choose a somewhat larger majority. This is speculative, but it seems possible that the widespread use of the phrase "51% attack" generated a false sense of security

Table 1: Expected Duration and Gross Cost of Attack

A. Expected Duration of Attack ($t$)

| | $e = 0$ | $e = 1$ | $e = 6$ | $e = 10$ | $e = 100$ | $e = 1000$ |
|---|---|---|---|---|---|---|
| $A = 1.05$ | 25.51 | 29.77 | 45.06 | 54.44 | 181.32 | 1,067.82 |
| $A = 1.1$ | 13.02 | 15.42 | 24.48 | 30.35 | 125.81 | 1,004.04 |
| $A = 1.2$ | 6.79 | 8.28 | 14.37 | 18.65 | 105.13 | 1,001.0 |
| $A = 1.25$ | 5.54 | 6.86 | 12.41 | 16.44 | 102.79 | 1,001.0 |
| $A = 1.33$ | 4.34 | 5.49 | 10.57 | 14.40 | 101.47 | 1,001.0 |
| $A = 1.5$ | 3.08 | 4.07 | 8.77 | 12.49 | 101.03 | 1,001.0 |
| $A = 2$ | 1.89 | 2.78 | 7.39 | 11.23 | 101.0 | 1,001.0 |
| $A = 5$ | 1.12 | 2.06 | 7.00 | 11.00 | 101.0 | 1,001.0 |

B. Gross Block-Compute Costs ($At$)

| | $e = 0$ | $e = 1$ | $e = 6$ | $e = 10$ | $e = 100$ | $e = 1000$ |
|---|---|---|---|---|---|---|
| $A = 1.05$ | 26.78 | 31.26 | 47.31 | 57.17 | 190.38 | 1,121.22 |
| $A = 1.1$ | 14.32 | 16.96 | 26.92 | 33.39 | 138.39 | 1104.35 |
| $A = 1.2$ | 8.14 | 9.93 | 17.24 | 22.38 | 126.15 | 1,201.20 |
| $A = 1.25$ | 6.93 | 8.57 | 15.51 | 20.55 | 128.49 | 1,251.25 |
| $A = 1.33$ | 5.78 | 7.31 | 14.06 | 19.15 | 134.96 | 1,331.33 |
| $A = 1.5$ | 4.62 | 6.11 | 13.15 | 18.73 | 151.54 | 1,501.5 |
| $A = 2$ | 3.78 | 5.56 | 14.78 | 22.45 | 202.0 | 2,002.0 |
| $A = 5$ | 5.59 | 10.29 | 35.01 | 55.00 | 505.0 | 5,005.0 |

*Notes:* Expected duration $t$, as a function of attacker majority $A$ and escrow period $e$, is computed using formula (4) in the text and double-checked using a computational simulation.

about how long a successful attack would take, and hence how expensive attacks would be on a gross-cost basis. Indeed, as shown in Proposition 5, the attack duration goes to infinity as the attacker majority converges to 50% from above.

Appendix B provides numerical analysis of the cost-minimizing attacker majority $A^*$, and hence minimum block-compute-costs $A^* \cdot t(A^*)$, as a function of the escrow period $e$.

### 4.2.3 Cases

Since $A^* \cdot t(A^*)$ depends on the escrow period, it will be helpful in our analysis to consider several cases:

- A base case in which the escrow period is the standard $e = 6$ blocks (60 minutes). In this case, the cost-minimizing attacker majority is $A^* = 1.53$ (or 60%), an average attack takes about $t = 8.6$ time (86 minutes), and the attacker's average block-compute-costs are $A^* \cdot t(A^*) = 13.14$ (See Appendix Table 4 for all calculations).

- An expensive attack case in which the escrow period is a full day, or $e = 144$ blocks. In this case, the cost-minimizing attacker majority is $A^* = 1.16$ (or 54%), and the attacker's average block-compute costs $A^* \cdot t(A^*)$ are 176.
- A very expensive attack case in which the escrow period is a full week, or $e = 1008$ blocks. In this case, the cost-minimizing attacker majority is $A^* = 1.07$ (or 52%), and the attacker's average block-compute-costs are $A^* \cdot t(A^*) = 1100$.

Please note that while the expensive and very-expensive cases are defined in terms of longer escrow periods, they can equivalently be interpreted in terms of assumptions about higher attacker frictions. For example, if the escrow period is the standard $e = 6$, the expensive case can be interpreted as attacker frictions $\kappa$ increasing attack costs by a factor of $\approx 10$, and the very expensive case can be interpreted as attacker frictions increasing attack costs by a factor of $\approx 100$.

## 4.3 Results

**Base Case** Results for the base case are presented in Table 2. To keep the Nakamoto blockchain secure in the base case requires a per-block cost that is about 7.6% of the value secured against a double-spending attack. This follows directly from equation (3), rewritten as $\frac{p_{block}}{V_{attack}} \geq \frac{1}{A^* \cdot t(A^*)}$. Per transaction, assuming 2000 transactions per block, the cost is 0.004% of the value secured.

These costs likely sound economically plausible. But consider how they scale with time and with the amount of value secured. A cost of 7.6% per block amounts to over 1,000% of the value secured per day, and about 400,000% of the value secured per year. For example, to secure the system against a \$1 billion attack requires \$4 trillion of annual security expense. To secure the system against a \$100 billion attack requires \$400 trillion of annual security expense—or about 4 times global GDP.

The per-transaction fee of 0.004% likely sounds very small, but this is a percentage of the value secured against attack, not the size of the transaction. For example, if an attack could be worth \$1bn, then each transaction must pay 0.004% of \$1bn which is \$40,000 of security costs. The intuition is that every transaction has to implicitly pay for the costs of the large standing army — as if a transaction for a cappucino has the security required by Fort Knox.[22]

---

[22] This observation explains the motivation for what are known as "level two" blockchain applications which batch small transactions off-chain to economize on fees. For example, the local coffee shop might only reconcile cappucino purchases on the blockchain once per year. The conceptual tension with such ideas, at least to date, is that they rely on off-chain sources of trust, which begs the question of the use of Nakamoto trust in the first place. See further discussion in the Conclusion and Appendix A.5.

Table 2: Cost Per (3) to Secure Against Attack: Base Case Analysis

| | Per-Block | Per-Day | Per-Year | Per-Transaction |
|---|---|---|---|---|
| Security Costs as % of Value Secured | 7.61% | 1,096% | 400,129% | 0.004% |
| To Secure: | | | | |
| \$1 thousand | \$76.1 dollars | \$11.0 thousand | \$4.0 million | 3.8 cents |
| \$1 million | \$76.1 thousand | \$11.0 million | \$4.0 billion | \$38.1 dollars |
| \$1 billion | \$76.1 million | \$11.0 billion | \$4.0 trillion | \$38.1 thousand |
| \$100 billion | \$7.6 billion | \$1.1 trillion | \$400.1 trillion | \$3.8 million |

*Notes:* See equation (3) and the text of Section 4.3 for description. The Base Case scenario assumes an escrow period of $e = 6$ blocks (one hour).

Table 3: Cost Per (3) to Secure Against Attack: Sensitivity Analysis

A. Security Costs as % of Value Secured

| Attack Scenarios | Per-Block | Per-Day | Per-Year | Per-Transaction |
|---|---|---|---|---|
| Base Case | 7.61% | 1,096% | 400,129% | 0.004% |
| Expensive | 0.57% | 82.0% | 29,939% | 0.0003% |
| Very Expensive | 0.09% | 13.1% | 4,777.9% | 0.00005% |

B. Cost to Secure Against \$1 Billion Attack

| Attack Scenarios | Per-Block | Per-Day | Per-Year | Per-Transaction |
|---|---|---|---|---|
| Base Case | \$76 million | \$11 billion | \$4 trillion | \$38 thousand |
| Expensive | \$6 million | \$820 million | \$299 billion | \$3 thousand |
| Very Expensive | \$1 million | \$131 million | \$48 billion | \$455 dollars |

*Notes:* See equation (3) and the text of Section 4.3 for description. The Base Case scenario assumes an escrow period of $e = 6$ blocks (one hour), the Expensive scenario assumes an escrow period of $e = 144$ blocks (one day), and the Very Expensive scenario assumes an escrow period of $e = 1008$ blocks (one week).

**Sensitivity Analysis** Table 3 provides the results of the sensitivity analysis. The expensive and very expensive cases improve the picture by one or two orders of magnitude, but the costs are still extremely high. In the expensive case, to secure against a \$1 billion attack requires per-transaction costs of \$3k and annual security costs of \$300 billion. In the very-expensive case, to secure against a \$1 billion attack requires per-transaction costs of \$450 and annual security costs of about \$50 billion. Even in the very-expensive case, with a one-week escrow period, to secure the system against a \$100 billion attack requires a per-year security cost of about \$5 trillion, which is more than 5% of global GDP.

## 4.4 Discussion

The double-spending analysis is consistent with the modest early use cases of Bitcoin, in which Bitcoin was primarily used by hobbyists and for small-scale black market activity (e.g., online gambling, Silk Road). In these early days, the amount that could be gained in a double-spending attack was not very high, because there were not high-value transaction opportunities. If a double-spending attack could gain at most $1,000, then the implicit cost per transaction in the base case necessary to secure the trust is just $0.04.

The double-spending analysis is also consistent with larger-scale black-market uses of cryptocurrencies, especially as black-market users may be most willing to pay the high implicit costs. For example, if a double-spending attack could gain at most $10 million, then the implicit cost per transaction in the base case needs to be about $380. This is modest relative to the costs of transporting large amounts of cash (Rogoff, 2017).

Where the analysis suggests greater skepticism is the use of cryptocurrencies and Nakamoto trust as a major component of the mainstream global financial system. If cryptocurrencies and Nakamoto trust were to become more integrated with the mainstream global financial system, then it would be possible to move amounts of value that are ordinary in the scheme of global finance, and hence it would be possible to double spend for amounts of value that are ordinary in the scheme of global finance. The analysis suggests that this scenario is unrealistic because of the way the trust model scales. To secure the system against attacks of $1 billion — which is less than 0.2% of daily trading volume in the U.S. Treasury market alone — requires a per-transaction security cost of $38,000, and an annual security cost of $4 trillion. To secure against attacks of $100 billion requires an annual security cost of four times global GDP. While market power and fees in traditional finance are clearly an important economic issue (Greenwood and Scharfstein, 2013; Philippon, 2015), and Huberman, Leshno and Moallemi (2021) are careful to remind us to compare the costs of the Nakamoto trust model against the costs of market power in traditional finance, it is clear from these calculations that Nakamoto trust is absurdly expensive relative to traditional trust. We will return to this comparison between Nakamoto trust and traditional trust in Section 6.

A conceptual insight that is reinforced by the double-spending analysis is that blockchain security should not be thought of as a 0-1 variable but as more like a very high percentage tax. Taxes and equilibrium reasoning (i.e., that security costs will have to go up in equilibrium if the blockchain becomes more economically useful) are natural to economists but are somewhat foreign to computer science. It is true that for any fixed dollar attack amount, there exists a long enough escrow period that keeps the system secure against such an attack, but this misses the point.

# 5 Analysis of Sabotage Attacks

A way out of the flow-stock argument considered theoretically in Section 3 and quantitatively in Section 4 is if the attack causes a significant decline in the value of the cryptocurrency, i.e., if $\Delta_{attack}$ is large. In this case, while the attacker still has their falsely-obtained goods or assets, the cryptocurrency the attacker is left with to double spend is now worth significantly less than before. Moreover, the decline in the value of the cryptocurrency would cause a decline in the value of any specialized capital the attacker needs to conduct the attack — this could be specialized hardware in the case of a proof-of-work cryptocurrency or stake in the case of a proof-of-stake cryptocurrency. The Bitcoin Wiki classifies the majority attack into its "Probably Not a Problem" category for these reasons (Bitcoin Wiki, 2022).

In this section I will show that this argument is logically correct. However, it raises the possibility of an attacker motivated by harming the cryptocurrency per se. I derive a "pick your poison" result in Section 5.1, discuss sabotage attacks in Section 5.2, discuss the combination of sabotage and specialized capital in Section 5.3 and consider collapse scenarios in Section 5.4.

## 5.1 Pick Your Poison

Assume that the double-spending attack analyzed in Section 4 causes a proportional decline in the value of the cryptocurrency of $\Delta_{attack}$. For this section, maintain the assumption from Section 3.1 that the attacker's cost of trust support is $c$ per unit of trust support per unit time. In Section 5.3 I will explicitly model the stock cost of capital, $C$, and assume that the attack harms the value of the attacker's capital stock too.

The $\Delta_{attack}$ parameter modifies the attacker's incentive compatibility condition in two ways. First, the attacker double spends for $V_{attack}$ of value (e.g., traditional financial assets from a traditional financial institution), but to realize this benefit has to hold cryptocurrency worth this amount, which declines in value $\Delta_{attack}V_{attack}$. Hence the net benefit of the attack is $(1 - \Delta_{attack})V_{attack}$. Second, the attacker's net cost of attack has to be adjusted for the decline in the value of the block rewards they earn. An $A$ attacker who attacks for $t$ time still earns $At$ block rewards in expectation, but each block reward declines in value by $\Delta_{attack}$. The net incentive compatibility condition thus becomes:

$$(\kappa + \Delta_{attack})At \cdot N^{*}c > (1 - \Delta_{attack})V_{attack}.$$

Substituting in equation (1) yields the following modification of equation (3):

$$p_{block} > \frac{(1-\Delta_{attack})}{At(\kappa+\Delta_{attack})}V_{attack} \tag{5}$$

The larger is $\Delta_{attack}$, the smaller is the per-block cost necessary to deter the double-spending attack. This is easiest to see by considering the extreme case of $\Delta_{attack} = 1$, i.e., if the attack causes a total collapse in the value of the cryptocurrency. In this case, the attacker loses exactly as much cryptocurrency value as they gain from double spending, so the numerator on the right-hand-side of (5) is zero — in effect, there is no chance to "double spend" at all. More generally, we have the following simple result:

**Proposition 6.** *("Pick your Poison") For any potential value of a double-spending attack* $V_{attack} > 0$*, and any level of block reward* $p_{block} > 0$*, the Nakamoto blockchain is secure against the double-spending attack if the post-attack decline in the cryptocurrency's value,* $\Delta_{attack}$*, is sufficiently high.*

*Proof.* Follows directly from (5), noting that the numerator of the right-hand-side goes to zero as $\Delta_{attack}$ goes to one. ☐

Proposition 6 may at first sound reassuring about the Nakamoto blockchain's security against double spending, but it raises the possibility of an attacker motivated by the harm to the cryptocurrency's value per se. $\Delta_{attack}$ is a "pick your poison" parameter: If $\Delta_{attack}$ is small, then the system is vulnerable to the double-spending attack analyzed in Section 4, and the implicit tax on economic activity using the blockchain has to be high. If $\Delta_{attack}$ is high, then the system is indeed secure against double-spending attacks — but this concedes that an attack would effectively cause a collapse in the trust, which in turn raises the possibility of an attacker movitated by this harm per se.[23]

Put differently, if $\Delta_{attack}$ is large, then a short time period of access to a large amount of trust-support capital can sabotage the whole novel economic system.

## 5.2 Sabotage

A well-known early paper on double-spending attacks, Rosenfeld (2014), observes that such an attack might in effect be a sabotage that causes collapse of the system:

---

[23] You can't have your cake and eat it too. If your view is that Bitcoin's value would fall in the immediate aftermath of the double-spending attack, but then would recover and this is all predictable, then the attacker can just hold on to their Bitcoins until the value recovers, and their cost of attack becomes what it was as originally analyzed in Section 4. A more elaborate version of this argument involves the Bitcoin community coodinating on a hard fork after the attack, to both nullify the attack and enable a recovery in the value of Bitcoin. This "community response" is a valid argument but begs several questions. See Appendix A.2 for discussion.

> “In this section we will assume $q < p$ [i.e., that the attacker does not have a majority]. Otherwise, all bets are off with the current Bitcoin protocol ... The honest miners, who no longer receive any rewards, would quit due to lack of incentive; this will make it even easier for the attacker to maintain his dominance. This will cause either the collapse of Bitcoin or a move to a modified protocol. As such, *this attack is best seen as an attempt to destroy Bitcoin*, motivated not by the desire to obtain Bitcoin value, but rather wishing to maintain entrenched economical systems or obtain speculative profits from holding a short position.” (Emphasis Added)

What is the value of a sabotage attack on a significant cryptocurrency such as Bitcoin or Ethereum? It is hard to say of course, but easy to imagine that the magnitudes are already large, and would be larger still if cryptocurrencies become more significantly integrated into the global financial system. Open interest in CME Bitcoin futures as of Sept 2023 is about 15,000 contracts, each tracking 5 Bitcoins, worth about $2 billion at current prices. According to data from The Block, open interest in Bitcoin futures aggregated across the major crypto exchanges has exceeded $20 billion (and is about $10 billion as of Sept 2023), and open interest in Ethereum futures has been as high as $10 billion (about $5 billion as of Sept 2023).[24] These figures give a sense of magnitudes for what could be made from a short-selling attack.

The market capitalization of cryptocurrencies gives another sense of magnitudes for the amount of economic harm an attacker could cause. Bitcoin’s market capitalization has been as high as about $1 trillion and Ethereum’s as high as about $500 billion. Across all crypto assets tracked by CoinMarketCap, market capitalization peaked at about $3 trillion in Nov 2021 and ranged from about $1.2 trillion to $2.2 trillion in 2022. Paypal co-founder Peter Thiel (2022) recently predicted that Bitcoin will be worth more than $100 trillion.

Last, Ethereum founder Vitalik Buterin described a future in which it is “just considered normal for there to be *trillion dollar assets* that are managed on Ethereum.” (Klein, 2022, emphasis added). If indeed assets of that magnitude are managed on Ethereum or other blockchains, without implicit or explicit protections from rule of law, then the value and risk of sabotage would be large.

## 5.3 Specialized Capital: Sabotage Attacks Cost a Stock not a Flow

Nakamoto (2008) envisioned that blockchain mining would be performed by ordinary computers: “one-CPU-one-vote.” Since 2013, however, Bitcoin mining has been dominated by computational

---

[24] CME open interest data is available via its website. I found open interest data from crypto exchanges at https://www.theblock.co/data/crypto-markets/futures/. I believe this to be a credible source but am less confident in it than I am the CME figures. For what it’s worth, when I wrote the June 2018 draft of this paper, CME + CBOE open interest was about $160 million, and crypto exchange futures did not, to my knowledge, exist at the time. That is, futures market open interest has grown by two orders of magnitude in the past few years.

equipment that is extremely specialized to Bitcoin mining. These machines have a large number of specialized chips called ASICs (application specific integrated circuits) which have the SHA-256 hash function programmed directly into their hardware — making them extremely fast at Bitcoin mining, and useless for any task that does not involve computing a large number of SHA-256 hashes.[25] The capital used for trust support for proof-of-stake blockchains, such as Ethereum's, is also intrinsically specialized to the blockchain. As emphasized in the introduction, if the capital used to maintain Nakamoto's anonymous, decentralized trust is non-repurposable, and the attack causes a collapse of the trust, then the attacker cost model needs to be modified. In addition to charging the attacker the flow cost of attack, the attacker must also be charged for the decline in the value of their specialized capital. This makes the attacker's cost more like a stock than a flow, and thus makes the blockchain more secure — but this security rests on the fragile precipice of specific capital and vulnerability to sabotage.

Let $c = rC + \eta$ denote the cost of trust support, where $C$ denotes the capital cost (e.g., ASICs, stake), $r$ denotes the rental cost of capital per unit time, and $\eta$ denotes variable costs per unit time (e.g., electricity). The honest-mining equilibrium (1) tells us that

$$N^*(rC + \eta) = p_{block}. \tag{6}$$

An outside attacker would need at least $N^*C$ worth of capital to conduct the attack, while an inside attacker would need at least $\frac{N^*C}{2}$ of capital.

Consider the extreme case in which the attack causes a total collapse of the economic value of the blockchain, including the specialized equipment; this is the case for which the incentive constraint against attack is least constraining. Given how small the flow costs of attack are, as analyzed in Section 4, ignore these and focus only on the stock cost of the specialized capital. This yields an incentive compatibility constraint for an outside sabotage attack of

$$N^*C > V_{attack} \tag{7}$$

and $\frac{N^*C}{2} > V_{attack}$ for an inside sabotage attack. We can compute $N^*C$ as a function of $p_{block}$. Let $\mu = \frac{rC}{rC+\eta}$ denote the capital share of mining. The honest-mining equilibrium (6) can be rewritten:

$$N^*C = \frac{\mu p_{block}}{r} \tag{8}$$

---

[25] These specialized chips are so much more efficient than non-specialized chips that, remarkably, even if one controlled *all* of the computational power owned by Amazon Web Services, one would have <0.1% of Bitcoin's hash rate. This calculation is based on AWS owning $65 billion of technology capital per its 2021 10-K filing, the calculations below that the Bitcoin capital stock is about $10 billion, and an assumption that specialized ASIC chips are at least 10,000 times more economically efficient at SHA-256 hashing than general-purpose computers.

Hence we can derive a modified version of equation (3):

$$p_{block} > \frac{r}{\mu} V_{attack}. \tag{9}$$

This is several orders of magnitude more secure than before because $r$ is the interest rate per block. Here is an example calculation for Bitcoin. Assume the capital share of mining is $\mu = 0.4$ (De Vries, 2018; Digiconomist, 2022), and the annual discount rate for ASICs is 50% (ASICs depreciate quickly and mining is risky), which implies that the per-unit-time discount rate is $r \approx 0.001\%$. Now compare $\frac{r}{\mu}$ on the right-hand-side of (9) to the $\frac{1}{A^* \cdot t(A^*)}$ factor on the right-hand-side of (3). If we use the base case value of $\frac{1}{A^* \cdot t(A^*)} = 7.6\%$, we have an over 2000-fold improvement in security. If we use these same values for $\mu$ and $r$ and use $p_{block}$ of \$250,000, then (8) implies a capital stock of \$10 billion, which about matches what is implied by current prices for state-of-the-art ASIC machines.[26] This suggests these magnitudes are reasonable.

Thus, if one concedes that a majority attack on Bitcoin would effectively be a sabotage that would cause the entire trust model to collapse, then, given the specialized computational equipment currently used for Bitcoin mining, Bitcoin is significantly more secure than is implied by the analysis in Section 4 of double-spending attacks.[27] The analysis in this section suggests that Bitcoin is currently secure against sabotages worth on the order of \$10 billion if the attacker comes from the outside, and for sabotages worth on the order of \$5 billion from inside attackers.

## 5.4 Collapse Scenarios

This section's analysis gives a candidate answer to what we can colloquially call the "Chicago Lunch Table question": if the analysis of Sections 3-4 is right, why have major cryptocurrencies like Bitcoin and Ethereum not been attacked already? Our answer is the combination of (i) specialized capital; (ii) an attack would not just steal money, but would in effect be a sabotage that causes collapse of the trust; and (iii) at present, the sabotage possibilities do not merit the cost.

Suppose, for the purpose of a speculative discussion, that this answer is correct. That is, the major cryptocurrency blockchains currently do satisfy the IC constraint (7) that is based on a stock cost of attack, $N^*C > V_{attack}$, but do not satisfy the IC constraint (2) that is based on a flow cost of attack, $A^*N^*c \cdot t(A^*) > V_{attack}$. For a sense of magnitudes, at current levels of

---

[26] A Bitmain Antminer S19j XP has a current retail price of \$4,983 (https://www.bitmain.com/, accessed August 31, 2023) and it would take about 2.3 million of these machines to match Bitcoin's current hash rate, for a total capital cost of about \$11.5 billion at retail prices. I do not have any information on how retail prices relate to the prices paid by large-scale miners.

[27] A blog post of Joseph Bonneau (2014) is the earliest written argument I am aware of that ASICs might make Bitcoin more secure.

$p_{block}$, these assumptions place the value of a successful attack on Bitcoin at less than \$10 billion (a rough estimate for $N^*C$) but greater than anywhere from about \$4 million (base case) to about \$275 million (very expensive case). This seems plausible given the values of 51% attacks, cryptocurrency thefts and cryptocurrency financial institution collapses that have been observed to date (see Appendix C).

This paper's analysis framework then suggests three possible scenarios that could precipitate a successful attack.

### 5.4.1 Collapse Scenario I: Cheap-Enough Specialized Capital

Suppose there are previous-generation chips that are not economically efficient for mining, but are powerful enough for the purpose of an attack and exist in large quantity. Mathematically, if $c = rC + \eta$ denotes the cost per unit of the most efficient specialized chips, then chips with variable costs $\eta' > c$ would be cost prohibitive for mining even if the capital is free. However, such chips could be used by an attacker if they exist in sufficient quantity: if there are at least $N^*$ computational units of such chips, an attacker could attack with low capital expenditure and a flow cost of $N^*\eta'$.

A variation on this theme occurs if an attacker can rent large amounts of specialized capital for a short period of time. The April 2022 attack on Beanstalk Farms for \$182 million involved the attacker borrowing enough tokens in that project to take temporary majority control, using what is known as a "flash loan." The attacker's cost was a rental cost of capital not the stock value of the tokens the attacker borrowed.

### 5.4.2 Collapse Scenario II: Sufficient Fall in Honest Mining Rewards

Suppose there is a large decline in Bitcoin's price for reasons unrelated to this paper's analysis. Since at present most of the block reward $p_{block}$ consists of newly issued Bitcoins, such a fall would directly cause a fall in honest miner compensation. This in turn could lead to a glut of specialized chips relative to the amount needed to maintain equilibrium in the honest mining market, (1), enabling an attacker to attack at a flow cost with low capital expenditure.

Additionally, the number of Bitcoins issued per block reward halves every four years. The next such halving will occur around March 2024, to 3.125 Bitcoins. In 2032, the reward will halve to less than 1 Bitcoin and by 2044 the reward will be less than 0.1 Bitcoin. Unless the dollar value per Bitcoin grows signficantly, or transaction fees increase significantly, these halvings will cause significant drops over time in $p_{block}$. This could also cause a glut of specialized chips, facilitating an attack at a flow cost.

### 5.4.3 Collapse Scenario III: Increase in Economic Importance (Relative to Cost)

The first two scenarios identify conditions under which the cost of attack changes from a stock to a flow. The other logical possibility is that a cryptocurrency grows in economic importance, relative to its compensation for trust support, to the point where the stock-cost incentive constraint (7) itself no longer holds. That is, $V_{attack} > N^*C$.

Speculatively, this seems most likely to occur if Bitcoin or Ethereum were to become more meaningfully integrated into the global financial system.

# 6 Comparison of Nakamoto Trust and Traditional Trust

In this section we return to the contrast discussed in the introduction between Nakamoto trust and traditional trust supported by rule of law and complementary sources, such as reputations and relationships. The essential difference is economies of scale.

## 6.1 Beckerian Deterrence as an Economy of Scale

For concreteness, consider a financial transaction between two parties of size $V$, but where one of the parties has an opportunity to cheat and steal the other party's assets. Specifically, Party 1 chooses an action from the set $\{Engage,\ Don't\ Engage\}$; Party 2 chooses an action from the set $\{Honest,\ Cheat\}$; if the players choose $Engage$ and $Honest$ then both parties get a payoff of $b > 0$, representing the net benefit of transacting; but if Party 1 chooses $Engage$ and Party 2 chooses $Cheat$, then Party 2 gets a payoff of $V$ and Party 1 gets a payoff of $-V$, representing that Party 2 has stolen Party 1's assets. If Party 1 chooses $Don't\ Engage$ then both parties get a payoff of zero. Clearly, in the game as described so far, the only equilibrium is for Party 1 to choose $Don't\ Engage$, so the parties will forego the benefit of transacting.

Now add a legal system with the power to enforce contracts. Specifically, if Party 2 plays $Cheat$, then Party 1 can pay a cost $c_l$ to adjudicate the transaction in court, and the court can perfectly observe whether Party 2 played $Cheat$ or $Honest$. If the court observes that Party 2 played $Cheat$ it can compel Party 2 to return Party 1's assets and punish Party 2 with a large fine $f$. In this scenario, Party 1's payoff is $-c_l$, the cost of bringing the matter to court, and Party 2's payoff is $-f$, the cost of the fine.

Clearly, the legal system makes it an equilibrium for the parties to transact honestly; the credible threat of a large fine deters Player 2 from cheating. And, since the players will transact honestly on the equilibrium path, the court need not even involve itself with most transactions in the first place. This is the point emphasized in the introduction about the economies of scale in traditional trust that is implicit in Hayek (1960) and Becker (1968). The key insight is: *A society that pays a fixed cost of operating a court system can facilitate honest transactions that have zero marginal cost of security because of the deterrence effect. This is a scale economy for traditional trust.*

We can translate this conceptual point about scale economies for traditional trust into the language of our earlier analysis. Consider the stripped-down model of Section 3.5 augmented in two ways. First, add a parameter $V_{honest}$ that represents the average volume transacted per period by honest users of the system if trust is secured. So, under Nakamoto trust, if the per-period payment for security $p$ exceeds the value of attacking the system $V_{attack}$, then honest participants transact $V_{honest}$ volume. Second, model traditional trust as costing a fixed cost $F$ plus a variable cost per unit transacted of $c$, such that society's cost of traditional trust is $F + cV_{honest}$ per period. In Figure 1, the $c$ can be interpreted as the cost of the security guards outside the bank and the $F$ can be interpreted as society's cost of police and courts.

The two trust models' cost per unit volume are thus:

$$Traditional\ Trust : \frac{F}{V_{honest}} + c \tag{10}$$

$$Nakamoto\ Trust : \frac{V_{attack}}{V_{honest}}$$

Traditional trust has scale economies to the extent that fixed costs $F$ that support trust can scale over a large quantity of transaction volume $V_{honest}$. Beckerian deterrence of crime is a leading example. Similar scale economies of trust arise in the private sector from fixed-cost investments in brands, reputation, relationships or collateral. Often such investments work in conjunction with societal fixed-cost investments in rule of law. For example, a firm's brand, reputation or relationships can serve as a credible commitment to provide high quality on those dimensions of quality that are not contractible (Nelson, 1974; Fudenberg, Levine and Maskin, 1994; Tadelis, 1999; Baker, Gibbons and Murphy, 2002; Levin, 2003), while laws or contracts can cover the dimensions that are contractible.

Collateral is a particularly important example to discuss in our context. Imagine that a bank intermediates the transaction above between Party 1 and Party 2 and has general-purpose collateral on its balance sheet that exceeds the value of the transaction. The bank can be trusted not to abscond with the parties' assets, without any appeal to reputation or brand, if a court can compel it to compensate the parties out of its general-purpose collateral if it cheats. Moreover, the cost of general-purpose collateral as a source of trust support is low because collateral earns a market rate of return. Under the assumptions of the Modigliani and Miller (1958) theorem the cost of collateral as a source of trust support is *zero.* Estimates from the empirical literature on the magnitudes of violations of the Modigliani-Miller theorem find that the cost of collateral is not literally zero, but is less than 1% per year of the collateral amount, which likely translates to less than 0.01% of transaction volume (see Budish and Sunderam, 2023). That is, $\frac{F}{V_{honest}} < 0.0001$ for collateral.

Nakamoto trust, in contrast, only enjoys economies of scale with transaction volume if the scope for attacking the system $V_{attack}$ does not grow with the system's usefulness for honest participants $V_{honest}$, which seems unlikely without support from rule of law. In the stylized financial transaction above, the size of the attack opportunity equals the size of the honest transaction, i.e., $\frac{V_{attack}}{V_{honest}} = 1$. Indeed, the ratio $\frac{V_{attack}}{V_{honest}}$ could easily exceed 1, as a majority attacker will engage in large transactions whereas $V_{honest}$ measures the size of average transactions.

## 6.2 Sense of Magnitudes for Finance

Total annual spending on police, prisons and courts, at the state, local and federal level, is about \$300 billion per year in the United States (Urban Institute, 2023). Real value added in the U.S. financial industry is about \$800 billion per year.[28] So we could use \$1 trillion per year as a conservative upper bound for the cost of trust in the U.S. financial sector, since the former figure includes spending that is unrelated to finance and the latter figure includes spending that is unrelated to trust.[29] We can use \$1 quadrillion per year as a conservative lower bound for transaction volume in the U.S. financial sector (Budish and Sunderam, 2023). We can thus upper bound the cost of trust support $\frac{F}{V_{honest}} + c$ in traditional finance by 0.1% of transaction volume. Clearly this is just a rough ballpark. Many fees in traditional finance, especially for large transactions, are on the order of 0.01% or less of transaction volume.[30]

---

[28] This figure is taken from the U.S. Bureau of Economic Analysis "Real Value Added by Industry" data, lines 56-57 ("Federal Reserve banks, credit intermediation, and related activities" and "Securities, commodity contracts, and investments.") Real value added measures payments to both capital and labor. See Philippon (2015) on why it is a useful measure for the cost of the financial sector.

[29] Gennaioli, Shleifer and Vishny (2015) distinguish between financial sector trust in the sense of security from expropriation or theft and trust in the sense of confidence to take risks. They argue that high fees and market power in the financial sector, especially as relates to investment management (see Greenwood and Scharfstein, 2013), can often be intrepreted as demand for the latter kind of trust, confidence to take risks. The former kind of trust, security from theft, is the aspect I have in mind here as relevant for the comparison to Nakamoto trust.

[30] For example, Hu, Pan and Wang (2021) report that the median fee charged in the treasury repo market is 2 basis points (0.02%) on an annual basis which translates to about 0.00005% per day. Interactive Brokers' fees for large foreign exchange transactions are about 0.001%. Budish, Lee and Shim (forthcoming) find that the average exchange trading fee in the U.S. stock market is \$0.0001 per-share-per-side, or about 0.0001% on a \$100 share of stock. Fees are higher for retail transactions but still often relatively small. Visa's annual operating expenses are less than 0.1% of their annual transaction volume and their revenue is about 0.25% of volume (Visa Annual Report, 2022). Asset management fees for hundreds of Vanguard index funds are less than 0.10% (https://investor.vanguard.com/investment-products/list/all). There are of course numerous other fees in finance that are much higher, but these tend not to be fees for trust (in the sense of security from theft, see previous

# 7 Conclusion

The anonymous, decentralized trust enabled by the Nakamoto (2008) blockchain, while ingenious, is *expensive*. Equation (3) says that for the trust to be meaningful requires that the flow cost of running the blockchain is large relative to the one-shot value of attacking it. In the double-spending attack considered in Section 4, the implication is that the transaction costs of the blockchain must be large in relation to the highest-value economic uses of the blockchain, which can be interpreted as a very large implicit tax — e.g., from \$500 to \$63,000 per transaction if the system is to be secured against \$1 billion attacks, all growing linearly with the value of attack. It would take *all of global GDP* to secure the system against a \$25 billion attack. The argument that an attack is more expensive than this flow cost, considered in Section 5, requires one to concede both (i) that the security of the blockchain relies on its use of specialized capital (contra to the Nakamoto (2008) ideal of "one-CPU-one-vote"), and (ii) that the blockchain is vulnerable to a sabotage attack that causes its novel form of trust to collapse. Risk of collapse is not a reassuring foundation for a novel economic system. Overall, the results place serious economic constraints on the use of the Nakamoto (2008) blockchain.

It bears emphasis that the paper's analysis is consistent with the continued use of cryptocurrencies and the Nakamoto blockchain for black-market purposes, and more generally in use cases where users are willing to pay the high implicit costs of anonymous, decentralized trust. Rather, this paper suggests skepticism and caution about large-scale uses of cryptocurrencies and the Nakamoto blockchain by traditional global businesses or the traditional global financial system. Such entities have access to cheaper forms of trust.

This paper's analysis is also consistent with the usefulness of the blockchain data structure *without* Nakamoto (2008)'s novel form of trust. This is often called distributed ledger technology or a permissioned blockchain (see fn. 2 and Section 2.5.1). Indeed, what this paper highlights is that it is exactly the aspect of Bitcoin and Nakamoto (2008) that is so innovative relative to these kinds of distributed databases — the anonymous, decentralized trust that emerges from proof-of-work — that is the source of its economic limits. As one specific example, it is completely consistent with this paper's analysis that Central Bank Digital Currencies (CBDCs) could be of high economic value. CBDCs take some technical inspiration from cryptocurrencies but are anchored in traditional trust from rule of law and the reputation of central banks, and thus do not face the scaling problem of Nakamoto trust highlighted in this paper.

---

fn.) but fees that reflect market power over consumers (Campbell, 2006, Greenwood and Scharfstein, 2013) or consumers' willingness to pay for financial advice (Gennaioli, Shleifer and Vishny, 2015). A famous paper of Philippon (2015) estimates the cost of the financial sector as a percentage of the value of real intermediation (as opposed to transaction volume) and finds that this is about 1.5-2%.

At a broader level, this paper builds on the view tracing all the way back to Adam Smith that government and laws are essential ingredients for the market system. A central point this paper has tried to emphasize is a fairly simple one implicit in Hayek (1960) and Becker (1968), though I have not seen it stated explicitly in this language, which is that traditional trust supported by rule of law enjoys economies of scale. Society pays the fixed cost of the apparatus of rule of law, or firms pay the fixed cost of building a brand or reputation or holding collateral (each of which works in conjunction with laws), and these fixed cost assets can provide trust over a large number of economic activities at low or zero marginal cost.

**Directions for Future Research.** I want to close with two directions for future research given the message of this paper. First, and most directly, is there a "solution" to this paper's critique of Bitcoin and Nakamoto trust? Informally, is there a way to generate trust in a public dataset (or, specifically, a cryptocurrency) that has some of the anonymity and decentralization aspects of Nakamoto while being signficantly less economically constrained by the arguments in this paper?[31] Appendix A describes several of the responses this paper has received since it first circulated in 2018. The most promising responses combine blockchain-based trust with traditional trust in some way. For example, Ethereum's new protocol algorithmically mimics the combination of collateral plus rule of law to punish small attackers but must rely on some external source of trust support to punish large attackers. If large sums of money were in dispute, it seems obvious that this external source of trust support would involve rule of law. Another interesting response is to concede that blockchain trust is intrinsically very expensive, per this paper's argument, but to only use it for occasional large transactions with long escrow periods ("Layer 1"), while most transactions are conducted off chain ("Layer 2"), supported by external sources of trust. An open conceptual question about these responses is what the permissionless consensus part adds given the external sources of trust support.

The second direction for future research is a broader conceptual question: How should economists model trust that comes from a combination of technology and rule of law? More generally, how should economists understand trust when it comes from multiple sources in the same transaction that work in complement with each other?[32] This is often the case in practice, with trust arising from some combination of rule of law, reputations, relationships, brands, collateral, norms,

---

[31] Recent efforts to formalize this question include Leshno, Pass and Shi (2023) and Lewis-Pye, Roughgarden and Budish (2023). See Appendices A.1 and A.6 for further discussion.

[32] One simple preliminary exploration of this question is in Section 4.1 of Budish and Sunderam (2023) who conceptualize trust as getting to cooperate-cooperate in a prisoner's dilemma (as discussed in La Porta et al., 1997), and model (i) technology as eliminating some actions from the possibility set, (ii) law as changing the payoffs to some actions via punishment, and (iii) reputation as the differential incentive to cooperate if play is repeated versus one-shot (as in the traditional folk theorem arguments of Aumann, 1959, Fudenberg, Levine and Maskin, 1994 and others).

technology, etc., often implicitly and without drawing notice. Consider the completely ordinary transaction of buying a cup of coffee at the local coffee shop. The consumer trusts the coffee shop to provide quality coffee because of reputational incentives, and perhaps implicitly food-safety laws. The coffee shop trusts the consumer's payment if cash because counterfeiting is technologically complex and illegal, and if electronic because of traditional cryptography and because the financial intermediary has reputational, relational, and legal reason to follow through. The employee trusts their employer to follow through with promised compensation because of laws and the implicit relational contract. Both the customer and the employee trust the other not to rob them because of laws and social norms. All this trust for a cup of coffee!

As one appreciates how many different sources of trust work together for even the most ordinary of economic transactions, it is hard not to regard the traditional market system, and old-fashioned rule of law, with a sense of wonder. It is exciting to imagine how new technological innovations, including new uses of cryptography, might continue to enhance this system — I trust this paper has convinced the reader that blockchain-based trust will not be a replacement for it.

# References

**Auer, Raphael.** 2019. “Beyond the Doomsday Economics of “Proof-of-Work” in Cryptocurrencies.” *BIS Working Paper No. 765.*

**Aumann, R.J.** 1959. “Acceptable Points in General Cooperative n-Person Games.” In *Contributions to the Theory of Games IV, Annals of Mathematics Study 40.* 287–324. Princeton University Press.

**Bailey, Norman T.J.** 1957. “Some Further Results in the Non-Equilibrium Theory of a Simple Queue.” *Journal of the Royal Statistical Society: (Statistical Methodology)*, 19(2): 326–333.

**Baker, George, Robert Gibbons, and Kevin J. Murphy.** 2002. “Relational Contracts and the Theory of the Firm.” *The Quarterly Journal of Economics*, 117(1): 39–84.

**Bakos, Yannis, and Hanna Halaburda.** 2021. “Tradeoffs in Permissioned vs Permissionless Blockchains: Trust and Performance.” *NYU Stern School of Business Working Paper.* Available at SSRN: `https://ssrn.com/abstract=3789425`.

**Bayer, Dave, Stuart Haber, and W. Scott Stornetta.** 1993. “Improving the Efficiency and Reliability of Digital Time-Stamping.” In *Sequences II: Methods in Communication, Security and Computer Science.* 329–334. Springer.

**Becker, Gary S.** 1968. “Crime and Punishment: An Economic Approach.” *Journal of Political Economy*, 76(2): 169–217.

**Biais, Bruno, Christophe Bisiere, Matthieu Bouvard, and Catherine Casamatta.** 2019. “The Blockchain Folk Theorem.” *The Review of Financial Studies*, 32(5): 1662–1715.

**Bitcoin Magazine.** 2022. “Peter Thiel - Bitcoin Keynote - Bitcoin 2022 Conference.” Last modified April 7, 2022. Retrieved May 1, 2022 from `https://www.youtube.com/watch?v=ko6K82pXcPA`.

**Bitcoin Wiki.** 2020*a*. “Weaknesses → Might Be a Problem → Energy Consumption.” Last modified June 27, 2020. Retrieved April 22, 2022, from `https://en.bitcoin.it/wiki/Weaknesses#Energy_Consumption`.

**Bitcoin Wiki.** 2020*b*. “Weaknesses → Probably Not a Problem → Attacker Has A Lot of Computing Power.” Last modified June 27, 2020. Retrieved April 22, 2022, from `https://en.bitcoin.it/wiki/Weaknesses#Attacker_has_a_lot_of_computing_power`.

**Bitcoin Wiki.** 2022. "Irreversible Transactions → Attack Vectors → Majority Attack." Last modified April 8, 2022. Retrieved April 22, 2022, from `https://en.bitcoin.it/wiki/Irreversible_Transactions#Majority_attack`.

**Böhme, Rainer, Nicolas Christin, Benjamin Edelman, and Tyler Moore.** 2015. "Bitcoin: Economics, Technology, and Governance." *Journal of Economic Perspectives*, 29(2): 213–238.

**Bonneau, Joseph.** 2014. "Why ASICs May Be Good for Bitcoin." Last modified December 12, 2014. Retrieved April 22, 2022 from `https://freedom-to-tinker.com/2014/12/12/why-asics-may-be-good-for-bitcoin/`.

**Bonneau, Joseph.** 2016. "Why Buy When You Can Rent? Bribery Attacks on Bitcoin Consensus." In *International Conference on Financial Cryptography and Data Security.* 19–26. Springer.

**Budish, Eric.** 2018. "The Economic Limits of Bitcoin and the Blockchain." NBER Working Paper No. 24717.

**Budish, Eric, and Adi Sunderam.** 2023. "Blockchain Technology and Stablecoins in Traditional Finance." In *Sveriges Riksbank 7th Annual Macroprudential Conference.*

**Budish, Eric, Robin S. Lee, and John J. Shim.** forthcoming. "A Theory of Stock Exchange Competition and Innovation: Will the Market Fix the Market?" *Journal of Political Economy.*

**Buterin, Vitalik.** 2014*a*. "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform." (White Paper).

**Buterin, Vitalik.** 2014*b*. "Slasher: A Punitive Proof-of-Stake Algorithm." Last modified January 15, 2014. Retrieved May 5, 2022 from `https://blog.ethereum.org/2014/01/15/slasher-a-punitive-proof-of-stake-algorithm/`.

**Buterin, Vitalik.** 2016. "A Proof of Stake Design Philosophy." *Medium*, Last Modified December 30, 2016. Retrieved May 1, 2022 from `https://medium.com/@VitalikButerin/a-proof-of-stake-design-philosophy-506585978d51`.

**Buterin, Vitalik.** 2017. "Minimal Slashing Conditions." Last Modified March 2, 2017. Retrieved Nov 7, 2023 from `https://medium.com/@VitalikButerin/minimal-slashing-conditions-20f0b500fc6c`.

**Buterin, Vitalik.** 2020. "Combining GHOST and Casper." *arXiv preprint arXiv:2003.03052.*

**Buterin, Vitalik.** 2022. "What in the Ethereum Application Ecosystem Excites Me." Last Modified December 5, 2022. Retrieved Sep 26, 2023 from `https://vitalik.eth.limo/general/2022/12/05/excited.html`.

**Buterin, Vitalik, and Virgil Griffith.** 2019. "Casper the Friendly Finality Gadget." *arXiv preprint arXiv:1710.09437.*

**Campbell, John Y.** 2006. "Household Finance." *The Journal of Finance*, 61(4): 1553–1604.

**Carlsten, Miles, Harry Kalodner, S. Matthew Weinberg, and Arvind Narayanan.** 2016. "On the Instability of Bitcoin Without the Block Reward." In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security.* 154–167.

**Chiu, Jonathan, and Thorsten V. Koeppl.** 2022. "The Economics of Cryptocurrency: Bitcoin and Beyond." *Canadian Journal of Economics*, 55(4): 1762–1798.

**Cochrane, John H.** 2013. "Finance: Function Matters, Not Size." *Journal of Economic Perspectives*, 27(2): 29–50.

**Cong, Lin William, Zhiguo He, and Jiasun Li.** 2021. "Decentralized Mining in Centralized Pools." *The Review of Financial Studies*, 34(3): 1191–1235.

**Cox, Jeff.** 2021. "Yellen Sounds Warning About 'Extremely Inefficient' Bitcoin." *CNBC*. Last modified February 22, 2021. Retrieved May 1, 2022 from `https://www.cnbc.com/2021/02/22/yellen-sounds-warning-about-extremely-inefficient-bitcoin.html`.

**De Vries, Alex.** 2018. "Bitcoin's Growing Energy Problem." *Joule*, 2(5): 801–805.

**Digiconomist.** 2022. "Bitcoin Energy Consumption Index." Retrieved May 19, 2022 from `https://digiconomist.net/bitcoin-energy-consumption`.

**Dolev, D., and H. R. Strong.** 1983. "Authenticated Algorithms for Byzantine Agreement." *SIAM Journal on Computing*, 12(4): 656–666.

**Dwork, Cynthia, Nancy Lynch, and Larry Stockmeyer.** 1988. "Consensus in the Presence of Partial Synchrony." *Journal of the ACM*, 35(2): 288–323.

**Easley, David, Maureen O'Hara, and Soumya Basu.** 2019. "From Mining to Markets: The Evolution of Bitcoin Transaction Fees." *Journal of Financial Economics*, 134(1): 91–109.

**Eyal, Ittay, and Emin Gun Sirer.** 2014. “Majority is not Enough: Bitcoin Mining is Vulnerable.” In *Proceedings of the 18th International Conference on Financial Cryptography and Data Security (FC)*. 436–454. Springer.

**Fischer, M., N. Lynch, and M. Paterson.** 1985. “Impossibility of Distributed Consensus with One Faulty Process.” *Journal of the ACM*, 32(2): 374–382.

**Foley, Sean, Jonathan R. Karlsen, and Tālis J. Putniņš.** 2019. “Sex, Drugs, and Bitcoin: How Much Illegal Activity is Financed through Cryptocurrencies?” *The Review of Financial Studies*, 32(5): 1798–1853.

**Friedman, Milton.** 1962. *Capitalism and Freedom.* The University of Chicago Press.

**Fudenberg, Drew, David Levine, and Eric Maskin.** 1994. “The Folk Theorem with Imperfect Public Information.” *Econometrica*, 62(5).

**Gans, Joshua S., and Hanna Halaburda.** 2023. “‘Zero Cost’ Majority Attacks on Permissionless Blockchains.” Available at `https://ssrn.com/abstract=4505460`.

**Gennaioli, Nicola, Andrei Shleifer, and Robert Vishny.** 2015. “Money Doctors.” *The Journal of Finance*, 70(1): 91–114.

**Gensler, Gary.** 2021. “Remarks Before the Aspen Security Forum.” Last modified August 3, 2021. Retrieved May 1, 2022 from `https://www.sec.gov/news/public-statement/gensler-aspen-security-forum-2021-08-03`.

**Goldman Sachs.** 2018. “Blockchain - The New Technology of Trust.” Retrieved April 11, 2018, from `http://www.goldmansachs.com/our-thinking/pages/blockchain/`.

**Greenwood, Robin, and David Scharfstein.** 2013. “The Growth of Finance.” *Journal of Economic Perspectives*, 27(2): 3–28.

**Grunspan, Cyril, and Ricardo Perez-Marco.** 2017. “Double Spend Races.” *arXiv preprint arXiv:1702.02867.*

**Guiso, Luigi, Paola Sapienza, and Luigi Zingales.** 2006. “Does Culture Affect Economic Outcomes?” *Journal of Economic Perspectives*, 20(2): 23–48.

**Haber, Stuart, and W. Scott Stornetta.** 1991. “How to Time-Stamp a Digital Document.” *Journal of Cryptography*, 3(2): 99–111.

**Halaburda, Hanna, Guillaume Haeringer, Joshua S. Gans, and Neil Gandal.** 2022. "The Microeconomics of Cryptocurrencies." *Journal of Economic Literature*, 60(3): 971–1013.

**Hart, Oliver.** 1995. *Firms, Contracts, and Financial Structure.* Clarendon Press.

**Hayek, Friedrich A.** 1960. *The Constitution of Liberty.* The University of Chicago Press.

**Holmstrom, Bengt, and Paul Milgrom.** 1994. "The Firm as an Incentive System." *The American Economic Review*, 972–991.

**Huberman, Gur, Jacob D. Leshno, and Ciamac Moallemi.** 2021. "Monopoly without a Monopolist: An Economic Analysis of the Bitcoin Payment System." *The Review of Economic Studies*, 88(6): 3011–3040.

**Hu, Grace Xing, Jun Pan, and Jiang Wang.** 2021. "Tri-Party Repo Pricing." *Journal of Financial and Quantitative Analysis*, 56(1): 337–371.

**Kandori, Michihiro.** 1992. "Social Norms and Community Enforcement." *The Review of Economic Studies*, 59(1): 63–80.

**Klein, Ezra.** 2022. "Ethereum's Founder on What Crypto Can – and Can't – Do." Audio podcast episode. Available at `https://www.nytimes.com/2022/09/30/podcasts/transcript-ezra-klein-interviews-vitalik-buterin.html`.

**Kreps, David M., Paul Milgrom, John Roberts, and Robert Wilson.** 1982. "Rational Cooperation in the Finitely Repeated Prisoners' Dilemma." *Journal of Economic Theory*, 27(2): 245–252.

**Kroll, Joshua A., Ian C. Davey, and Edward W. Felten.** 2013. "The Economics of Bitcoin Mining or, Bitcoin in the Presence of Adversaries." In *12th Workshop on the Economics of Information Security.*

**Krugman, Paul.** 1998. "Baby-Sitting the Economy." *Slate.* Last modified Aug 14, 1998. Retrieved May 1, 2022 from `https://slate.com/business/1998/08/baby-sitting-the-economy.html`.

**Lamport, Leslie, Robert Shostak, and Marshall Pease.** 1982. "The Byzantine Generals Problem." *ACM Transactions on Programming Languages and Systems*, 4(3): 382–401.

**La Porta, Rafael, Florencio Lopez-de-Silanes, Andrei Shleifer, and Robert W. Vishny.** 1997. "Trust in Large Organizations." *American Economic Review Papers and Proceedings*, 87(2): 333–338.

**La Porta, Rafael, Florencio Lopez-de-Silanes, Andrei Shleifer, and Robert W. Vishny.** 1998. “Law and Finance.” *Journal of Political Economy*, 106(6): 1113–1155.

**Leshno, Jacob, Rafael Pass, and Elaine Shi.** 2023. “Can open decentralized ledgers be economically secure?” *Cryptology ePrint Archive Working Paper 2023/1516*, Available at `https://eprint.iacr.org/2023/1516`.

**Levine, Matt.** 2017. “Bank Blockchains and an Alibaba Box.” *Bloomberg View*, Last modified January 10, 2017. Retrieved May 24, 2022 from `https://www.bloomberg.com/view/articles/2017-01-10/bank-blockchains-and-an-alibaba-box`.

**Levin, Jonathan.** 2003. “Relational Incentive Contracts.” *American Economic Review*, 93(3): 835–857.

**Lewis-Pye, Andrew, and Tim Roughgarden.** 2023. “Permissionless Consensus.” *arXiv preprint arXiv:2304.14701.*

**Lewis-Pye, Andrew, Tim Roughgarden, and Eric Budish.** 2023. “Economically Secure Permissionless Consensus.” Manuscript in Preparation.

**Ma, June, Joshua S. Gans, and Rabee Tourky.** 2018. “Market Structure in Bitcoin Mining.” NBER Working Paper No. 24242.

**Makarov, Igor, and Antoinette Schoar.** 2021. “Blockchain Analysis of the Bitcoin Market.” NBER Working Paper No. 29396.

**Modigliani, Franco, and Merton H. Miller.** 1958. “The Cost of Capital, Corporation Finance and the Theory of Investment.” *The American Economic Review*, 48(3): 261–297.

**Moroz, Daniel J., Daniel J. Aronoff, Neha Narula, and David C. Parkes.** 2020. “Double-Spend Counterattacks: Threat of Retaliation in Proof-of-Work Systems.” *arXiv preprint arXiv:2002.10736.*

**Nakamoto, Satoshi.** 2008. “Bitcoin: A Peer-to-Peer Electronic Cash System.” (White Paper).

**Narayanan, Arvind, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder.** 2016. *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction.* Princeton, NJ:Princeton University Press.

**Nelson, Phillip.** 1974. “Advertising as Information.” *Journal of Political Economy*, 82(4): 729–754.

**Pease, M., R. Shostak, and L. Lamport.** 1980. “Reaching Agreement in the Presence of Faults.” *Journal of the ACM*, 27(2): 228–234.

**Philippon, Thomas.** 2015. “Has the US Finance Industry Become Less Efficient? On the Theory and Measurement of Financial Intermediation.” *American Economic Review*, 105(4): 1408–38.

**Popper, Nathaniel.** 2015. *Digital Gold: Bitcoin and the Inside Story of the Misfits and Millionaires Trying to Reinvent Money.* Harper Paperbacks.

**Prat, Julien, and Benjamin Walter.** 2021. “An Equilibrium Model of the Market for Bitcoin Mining.” *Journal of Political Economy*, 129(8): 2415–2452.

**Rogoff, Kenneth.** 2017. *The Curse of Cash.* Princeton University Press.

**Rosenfeld, Meni.** 2014. “Analysis of Hashrate-Based Double-Spending.” *arXiv preprint arXiv:1402.2009.*

**Roughgarden, Tim.** 2023. “COMS 6998-006: Foundations of Blockchains.” Retrieved Sep 26, 2023 from `https://timroughgarden.github.io/fob21/`.

**Sapienza, Paola, and Luigi Zingales.** 2012. “A Trust Crisis.” *International Review of Finance*, 12(2): 123–131.

**Schelling, Thomas C.** 1956. “An Essay on Bargaining.” *American Economic Review*, 46(3): 281–306.

**Schelling, Thomas C.** 1960. *The Strategy of Conflict.* Harvard University Press.

**Shleifer, Andrei, and Robert W. Vishny.** 1997. “A Survey of Corporate Governance.” *The Journal of Finance*, 52(2): 737–783.

**Smith, Adam.** 1776. *The Wealth of Nations.* Penguin Classics; First Edition (March 25, 1982).

**Tabarrok, Alex.** 2019. “Bitcoin is Less Secure than Most People Think.” Last Modified January 7, 2019. Retrieved Sep 26, 2023 from `https://marginalrevolution.com/marginalrevolution/2019/01/bitcoin-much-less-secure-people-think.html`.

**Tadelis, Steven.** 1999. “What’s in a Name? Reputation as a Tradeable Asset.” *American Economic Review*, 89(3): 548–563.

**Tas, Ertem Nusret, David Tse, Fangyu Gai, Sreeram Kannan, Mohammad Ali Maddah-Ali, and Fisher Yu.** 2023. “Bitcoin-Enhanced Proof-of-Stake Security: Possibilities and Impossibilities.” *arXiv preprint arXiv:2207.08392.*

**Urban Institute.** 2023. “Criminal Justice Expenditures: Police, Corrections, and Courts.” Retrieved Sep 26, 2023 from `https://www.urban.org/policy-centers/cross-center-initiatives/state-and-local-finance-initiative/state-and-local-backgrounders/criminal-justice-police-corrections-courts-expenditures`.

**Visa.** 2021. “Visa Annual Report.” Last modified Nov 18, 2021. Retrieved May 1, 2022 from `https://s29.q4cdn.com/385744025/files/doc_downloads/Visa-Inc_-Fiscal-2021-Annual-Report.pdf`.

**Visa.** 2022. “Visa Inc. Fiscal 2022 Annual Report.” Retrieved Sep 22, 2023 from `https://s29.q4cdn.com/385744025/files/doc_downloads/2022/Visa-Inc-Fiscal-2022-Annual-Report.pdf`.

**Zingales, Luigi.** 2015. “Does Finance Benefit Society?” *The Journal of Finance*, 70(4): 1327–1363.

# Appendix

## A Discussion of Responses to this Paper's Argument

This paper first circulated in shorter form in June 2018. I received a lot of comments and counter-arguments in response to the paper's main line of argument.

I have tried to handle the central line of counter-argument throughout the main text of this updated draft. This is the point made by Huberman, Leshno and Moallemi (2021) and many practitioners that we should compare Nakamoto's costs to the costs of market power in traditional finance, which are also high.[33] I hope the present draft of the text makes more clear the conditional nature of the paper's argument: if Nakamoto trust becomes more economically useful, then it will have to get even more expensive, linearly, or it will be vulnerable to attack. I hope as well that the more explicit computational simulations, for varying levels of $V_{attack}$ all the way up to \$100 billion, make clear that the way Nakamoto's security cost model scales is importantly different from how costs scale for traditional finance protected by rule-of-law.

In this appendix I discuss several of the other most common comments and counter-arguments I have received about this paper since it was first circulated.

### A.1 Ethereum Proof-of-Stake with Slashing as an Attempt to Mimic Collateral Plus Rule of Law

This sub-section expands on the discussion of proof-of-stake in the main text in Section 2.5.3.

In its simplest form, proof-of-stake is vulnerable to the exact same critique as proof-of-work. Just conceptualize $c$ as the per-block opportunity cost of locking up stake for validation. Equations (1)-(3) go through virtually unchanged.

However, the use of stakes rather than computational work opens up possibilities for punishing attackers that do not exist in Nakamoto (2008). This is for two reasons. First, stakes are locked on chain, somewhat analogously to collateral, whereas the computers used for proof-of-work blockchains exist off chain in the physical world. Second, the use of stakes opens up the use of a different paradigm for permissionless consensus called Byzantine Fault Tolerance (BFT). Very roughly speaking, all locked-up stake explicitly signs all transactions, and for a transaction to be confirmed, 2/3 of locked-up stake must sign it. Therefore, in the event of a double-spending attack in which two blocks sending the same funds to alternative destinations are confirmed, at least 1/3

---

[33] See Philippon (2015) and Greenwood and Scharfstein (2013) on high costs of traditional finance, and see Cochrane (2013) for a counterpoint.

of the total stake (i.e., 2/3 + 2/3 - 1) must have signed conflicting transactions. This creates algorithmically observable proof that an attacker has misbehaved, which can be the grounds for the algorithmic confiscation of the attacker's capital — analogously to the use of rule of law. This is a key reason why Ethereum, the second largest cryptocurrency after Bitcoin, adopted proof-of-stake starting in September 2022 (Buterin, 2014*b*; Buterin, 2016; Buterin, 2020; Buterin and Griffith, 2019).

This approach would be a compelling response to the issues raised in this paper if it works, because it would make the cost of a double-spending attack a stock not a flow without needing the whole system of trust to collapse. Mathematically, if we denote $C$ the value of a unit of stake, $c$ the opportunity cost of stake per unit time, and $N^*$ the equilibrium level of honest stake (i.e., $N^*c = p_{block}$ as in (1)), then an attacker from the outside will have at least $\frac{N^*C}{2}$ of capital confiscated and an attacker from the inside will have at least $\frac{N^*C}{3}$ of capital confiscated. At recent levels of $N^*C$ for Ethereum, this is about \$10-\$15 billion. So a double-spending attack against Ethereum is similarly expensive to the sabotage attacks against Bitcoin studied in Section 5, but *without the premise that an attack would collapse the system.* This is clearly an important improvement if it works.

Unfortunately, as previewed in the main text in Section 2.5.3, recent research suggests that this approach may not work — at least on its own without external support from rule of law. The subtle issue is how do you ensure that the attacker's stake is confiscated before they can withdraw and spend it — keeping in mind that the attacker is large, so can control the blockchain including potentially its adjudication of the attacker's own crime. Tas et al. (2023) show an impossibility result for what they call "slashable safety", which requires that the double-spend attacker's stake is slashed before the attacker can withdraw it from the system. Their result holds for any strictly-positive amount of slashing. Lewis-Pye, Roughgarden and Budish (2023) obtain a positive result under two strong assumptions: (i) a network reliability assumption that says that it is possible to define a finite amount of time that is long enough to ensure that any network outage repairs by then, and (ii) that the attacker's majority is no more than 5/9 of the total stake.[34] If the attacker is too large then the Tas et al. (2023) impossibility kicks in and the attacker can prevent the protocol's version of a legal system from punishing them. An interpretation of these results is that (under strong network reliability assumptions) the Ethereum blockchain can mimic the combination of collateral plus rule of law to deter small attackers, but must rely either on

---

[34] The intuition for where the 5/9 bound comes from is as follows. As described above, at least 1/3 of the total stake must have signed conflicting transactions to double spend. So an attacker with 5/9 would be left with 5/9 - 1/3 = 2/9 of the total stake post-slashing, while honest participants would have the other 4/9. So, post-slashing, the attacker would have weakly less than 1/3 of the post-slashing stake, which is the critical threshold for the usual positive results for BFT consensus to come into play (see citations in Section 2.3).

a collapse argument or traditional rule of law external to the blockchain to punish large-enough attackers.

A separate issue is that the use of the BFT consensus paradigm creates a vulnerability to what are known as "liveness" attacks, as distinct from "safety" attacks like double spending. Tradeoffs between safety and liveness properties are commonplace in the literature on distributed consensus protocols. The issue is that since in BFT consensus 2/3 of all stake must sign a transaction for it to be finalized, an attacker can halt consensus for a long period of time by simply refusing to sign transactions. Importantly, in the case of a liveness attack, unlike a double-spending attack, the attacker has not observably done anything wrong like sign conflicting transactions. They could just be having a network outage. For this reason, confiscating stake for being silent is controversial and Ethereum has chosen to do so only very slowly. Using Ethereum's slashing formula as of its Fall 2022 switch to proof-of-stake, I compute that if the total amount of stake on Ethereum is $25 billion, an attacker could silence Ethereum for an hour for $13,000, a day for $7 million, or a full week for $388 million. If Ethereum became a significant component of the global financial system, these costs of shutting it down seem trivial to a motivated saboteur.

I conclude that Ethereum's new approach to permissionless consensus does achieve a security improvement relative to Nakamoto (2008) and this improvement can be understood through the lens of this paper. Under plausible assumptions about network latency, the combination of proof-of-stake and slashing can make double-spending attacks by a sub-5/9 attacker economically expensive — i.e., cost a stock not a flow — without reliance on a collapse argument. However, given the vulnerability to larger attackers and to liveness attacks, it remains difficult to see how Ethereum or others like it can be secure on their own without rule of law as an external source of support. As noted in the conclusion, a very interesting question for future research is how conceptually to think about combinations of trust from technology and trust from rule of law. What do combinations of the two make economically possible that is not feasible with either alone?

## A.2 Community

A majority attack on Bitcoin or any other major cryptocurrency would be widely noticed. A line of argument I heard frequently in response to the June 2018 draft is that the Bitcoin community would organize a response to the attack. For example, the community could organize a "hard fork" off of the state of the blockchain just prior to the attack, which would include all transactions perceived to be valid, void any perceived-as-invalid transactions, possibly confiscate or void the attacker's other Bitcoin holdings if these are traceable, and possibly change the hash function or find some other way to ignore or circumvent the attacker's majority of compute power.[35]

The community response argument seems valid as an argument that attacks might be more expensive or difficult to execute than is modeled here, but it raises two important issues.

First, and most obviously, the argument contradicts the notion of anonymous, decentralized trust. It relies on a specific set of trusted individuals in the Bitcoin community.

Second, consider the community response argument from the perspective of a traditional financial institution. In the event of a large-scale attack that involves billions of dollars, the traditional financial institution would, in this telling, be left in the hands of the Bitcoin community. At present, reliance on a tight-knit community of those most invested in Bitcoin (whether financially, intellectually, etc.), may sound reassuring — those with the most to lose would rally together to save it. But now imagine the hypothetical future in which Bitcoin becomes a more integral part of the global financial system, and imagine there is a fight over whether an entity like a Goldman Sachs is entitled to billions of dollars worth of Bitcoin that it believes was stolen — but the longest chain says otherwise. Will the "vampire squid" be made whole by the "Bitcoin community?" Quite possibly, but one can hopefully see the potential weakness of relying on an amorphous community as a source of trust for the global economic and financial system.

## A.3 Rule of Law

A related line of argument I have heard frequently is that, in the event of a large-scale attack specifically on a financial institution such as a bank or exchange, rule of law would step in. For example, the financial institutions depicted as the victims of a double-spend in Figure 3, once they realize they no longer have the Bitcoins paid to them because of the attack, would obtain help from rule-of-law tracing down the attacker and recovering the stolen funds.

This response, too, seems internally valid while contradicting the idea of anonymous, decentralized trust. It also seems particularly guilty of wanting to "have your cake and eat it too." In this view, cryptocurrencies are mostly based on anonymous, decentralized trust — hence evading most forms of scrutiny by regulators and law enforcement — but, if there is a large attack, then rule of law will come to the rescue.

## A.4 Counterattacks

Moroz et al. (2020) extend the analysis in Budish (2018) to enable the victim of a double-spending

---

[35] The phrase "hard fork" means that in addition to coordinating on a particular fork of a blockchain if there are multiple — in this case, the attacker's chain, which is the longest, and the chain the community is urging be coordinated on in response — the code used by miners is updated as well. This could include hard-coded state information such as the new chain or information about voided Bitcoins held by the attacker, code updates such as a new hash function, etc.

attack to attack back. They consider a game in which there is an Attacker and a Defender. If the Attacker double spends against the Defender for $v$ dollars, the Defender can then retaliate, themselves organizing a 51% or more majority, to attack back so that the original honest chain becomes the longest chain again. This allows the Defender to recover their property.

For example, suppose the escrow period is 6, denote the initial double-spend transaction as taking place in block 1, and suppose the attacker chain replaces the honest chain as soon as the escrow period elapses, as in Figure 3. Notationally, suppose the honest chain consists of blocks {1, 2, ..., 7} at the time the honest chain is replaced, and the attacker chain that replaces it is {1', 2', ..., 7', 8'}. If the Defender can quickly organize a majority of their own, then they can build off of the {1, 2, ..., 7} chain, and eventually surpass the attacker chain, recovering their property. For example, maybe the honest chain reaches block 10 before the Attacker chain reaches block 10', so then {1, 2, ..., 10} is the new longest chain and the Defender has their property back from the correct transaction in block 1.

This argument is game theoretically valid, and indeed there are theoretical subtleties to the argument that the reader can appreciate for themselves in the paper. That said, it relies on every large-scale participant in the Bitcoin system being able and willing to conduct a 51% attack on a moment's notice. This is kind of like requiring every major financial institution to have not just security guards, but access to a standing army.

## A.5 Modification to Nakamoto I: Increase Throughput

Bitcoin processes about 2000 transactions per block, which is about 288,000 per day or 105 million per year. In contrast, Visa processes about 165 billion transactions per year (Visa, 2021).

The reader will notice that the logic in equations (1)-(3) does not depend directly on the number of transactions in a block. If the number of transactions in a Bitcoin block were to increase by 1000x (to roughly Visa's level), then the required $p_{block}$ to keep Bitcoin secure against a given scale of attack $V_{attack}$, per equation (3) would not change. Thus, the required cost *per transaction* to keep Bitcoin secure against a given scale of attack would decline by a factor of 1000.

In this scenario of a 1000x throughput increase, Bitcoin's security costs per transaction are still large, but less astonishingly so. In the base case, to secure Bitcoin against a $1 billion attack would require costs per transaction of $31 instead of $31,000. To secure against a $100 billion attack would require costs per transaction of $3,100 instead of $3.1 million.

A subtlety is that as the number of transactions per block grows, so too might the scope for attack. That is, $V_{attack}$ might grow as well.

Still, this seems a promising response to the logic of this paper. A particularly interesting variation on this idea is the paradigm called "Level 2." In this paradigm, the Bitcoin blockchain

("Level 1") would be used for relatively large transactions, but smaller transactions would be conducted off-chain, possibly supported with traditional forms of trust, with just occasional netting on the main Bitcoin blockchain. In this paradigm, as well, the large transactions on chain could also have a long escrow period, making attacks more expensive.[36]

## A.6 Modification to Nakamoto II: Tweak Longest-Chain Convention

The discussion above in A.2 expressed skepticism about the "community" response to the logic of this paper. However, what about modifying the longest-chain convention to try to encode what the community would *want* to do in the event of an attack.

The modification to the longest-chain convention could take advantage of two specific features of double-spending attacks:

1. The Attacker has to sign transactions both to the victim of the double-spending attack — call this the Bank — and to another account they control — call this the Cousin account. The fact that there are multiple-signed transactions for the same funds is an initial proof that something suspicious has happened.
2. The Attacker has to make the signed transaction to the Bank public significantly before — in "real-world clock time" — the signed transaction to their Cousin account.

The difficulty with just using facts #1 and #2 to void the transaction to the Cousin is alluded to with the phrase "real-world clock time." Part of what the Nakamoto (2008) blockchain innovation accomplishes is a sequencing of data that does not rely on an external, trusted, time-stamping device.

Relatedly, the difficulty with just using fact #1 and having the policy "if there are multiple correctly signed transactions sending the same funds, destroy the funds" is that the victim of the double-spending attack, the Bank, will by now have sent real-world financial assets to the Attacker — and this transaction, in the real world (off the blockchain), cannot be voided no matter how we modify the blockchain protocol. A different way to put the concern is that such a policy would allow any party that sends funds on the blockchain in exchange for goods or financial assets off the blockchain, to then void the counterparty's received funds after the fact. This seems a recipe for sabotage of the traditional financial sector.

The open question, then, is whether the protocol can be modified so that in the event of fact #1, multiple signed transactions, there is some way to appeal to fact #2, grounded in the sequencing of events in real-world clock time, not adjudicated by the longest-chain convention's determination of the sequence of events.

---

[36] I thank Neha Narula for several helpful conversations about this approach.

One pursuit along these lines is Leshno, Pass and Shi (2023). Their approach, which they call "Stubborn Nakamoto", is fully secure against double-spending attacks but, instead, has to permanently halt in response to observing conflicting transactions. In consensus terminology, it trades a security problem for a liveness problem. In conjunction with a source of external trust support, such as rule of law, to restart the system in case of such an outage, this could work. The open conceptual question then becomes what the permissionless consensus part adds given the source of external trust support (i.e., the same question asked in the Conclusion and A.1).

# B Double-Spending Attack Technical Appendix

## B.1 Proof of Proposition 5 (Closed-Form Expression for Duration of Double-Spending Attack)

Let $s = 0$ denote the time of the last block prior to the attack. As a reminder, time is normalized so that one unit of time is the amount it takes on average for honest miners to mine one block, e.g., 10 minutes for Bitcoin.

The attacker spends Bitcoins in exchange for other goods or assets in the honest miners' first block after time 0. In parallel, the attacker mines an alternative chain starting from the last block prior to the attack.

Honest miners mine blocks as a Poisson process with rate 1, and the attacker mines at rate $A > 1$. Both the honest miners' and the attacker's chains are time-independent Poisson processes, with:

$$\begin{aligned} B_H(s) &\coloneqq \text{Number of blocks on honest chain at time } s, \\ B_A(s) &\coloneqq \text{Number of blocks on attacker chain at time } s. \end{aligned}$$

The attack is completed when both (i) the honest chain has mined at least $1+e$ blocks, therefore passing the attacker transaction's escrow period, and (ii) the attacker chain has mined strictly more blocks than the honest chain. Therefore, the expected duration of the double-spending attack, as a function of the attacker majority $A$ and escrow period $e$, is given by the stopping time formula:

$$t(A, e) = E[\inf\{s : B_H(s) \geq 1 + e,\, B_A(s) > B_H(s)\}].$$

It will be useful to define a random variable that denotes the time at which the honest chain completes the escrow period. Call this $S_H^{1+e}$:

$$S_H^{1+e} := \inf\{s : B_H(s) \geq 1+e\}.$$

Similarly, it will be useful to define the difference in length between the honest chain and the attacker chain at the random time at which the honest chain completes the escrow period. Call this $D^{1+e}$:

$$\begin{aligned} D^{1+e} &:= B_H(S_H^{1+e}) - B_A(S_H^{1+e}) \\ &= (1+e) - B_A(S_H^{1+e}). \end{aligned}$$

If the realization of $D^{1+e} < 0$, the attacker chain is strictly longer than the honest chain at the conclusion of the escrow period, and the attacker immediately completes the double-spending attack. The total duration of attack is simply the time elapsed in completing the escrow period.

Else, if the realization of $D^{1+e} \geq 0$, the attacker faces a deficit and must continue the attack after the conclusion of the escrow period. In this case, the total duration of attack is the length of the escrow period plus the time it takes for the attacker to overcome the deficit. Note, if the attacker deficit is $i$ blocks, to overcome the deficit the attacker must mine $i+1$ more blocks than the honest miners, as the attacker chain must be strictly longer than the honest chain to complete the attack.

Hence, we can partition $t(A,e)$ based on the sign of $D^{1+e}$ for a tractable expression for $t(A,e)$:

$$\begin{aligned} t(A,e) = {} & E[S_H^{1+e}|D^{1+e} < 0)] \times P(D^{1+e} < 0) \\ & + \sum_{i=0}^{1+e} \Big( E[S_H^{1+e}|D^{1+e} = i] + E[\text{Time for attacker to overcome deficit} = i] \Big) \times P(D^{1+e} = i) \\ = {} & E[S_H^{1+e}] + \sum_{i=0}^{1+e} E[\text{Time for attacker to overcome deficit} = i] \times P(D^{1+e} = i). \end{aligned}$$

The second equality follows from the law of total probability, $\sum_{k=-\infty}^{1+e} E[S_H^{1+e}|D^{1+e} = k] \times P(D^{1+e} = k) = E[S_H^{1+e}]$. Now, there are three terms left to simplify: $E[S_H^{1+e}]$, $E[\text{Time for attacker to overcome deficit} = i]$, and $P(D^{1+e} = i)$.

Consider the first term, $E[S_H^{1+e}]$. A well-known property of Poisson processes is that arrivals are distributed according to the Gamma distribution, $S_H^{1+e} \sim \mathit{Gamma}(1+e, 1)$. This Gamma distribution has a simple expression for its mean:

$$E[S_H^{1+e}] = 1 + e.$$

Now consider the second term, $E[\text{Time for attacker to overcome deficit} = i]$. Via the Markov property, we know this random variable does not depend on *when* the honest chain finishes the escrow period, only the deficit itself. So, consider the stochastic process:

$$\begin{aligned} D_{i+1}(s) &:= \overline{B_H}(s) - \overline{B_A}(s) \\ &= \text{Difference between (auxiliary)} \\ &\quad\ \text{honest and attacker chains at } s. \\ \overline{B_H}(0) &= i+1 \\ \overline{B_A}(0) &= 0 \end{aligned}$$

That is, start two auxiliary honest and attacker chains at $s = 0$, but initialize the difference between the length of the two chains to be $i + 1$, as the attacker must overcome a deficit of $i$. The stochastic movement of this difference process can be thought of as an $M/M/1$ queue, where 'arrivals' are blocks on the honest chain, and 'departures' are blocks on the attacker's chain. We want the time it takes the difference process $D_{i+1}(s)$ to reach 0 – i.e., how long it takes the attacker to overcome the deficit $i$. In the queueing literature, this is known as the "first passage time" of a queue, $\text{FPT}(i + 1) := \inf\{s : D_{i+1}(s) = 0\}$. The mean of the first passage time of the $M/M/1$ queue is $E[\text{FPT}(i + 1)] = \frac{i+1}{A-1}$ (equation 41 in Bailey, 1957). Hence,

$$E[\text{Time for attacker to overcome deficit} = i] = \frac{i+1}{A-1}.$$

Finally, consider the term $P(D^{1+e} = i)$. Recall $D^{1+e}$ is the difference between the honest and attacker's chains' length at the time the honest chain completes the escrow period. Hence, we can write:

$$\begin{aligned} \{D^{1+e} = i\} &= \left\{B_H(S_H^{1+e}) - B_A(S_H^{1+e}) = i\right\} \\ &= \left\{(1+e) - B_A(S_H^{1+e}) = i\right\} \\ &= \left\{B_A(S_H^{1+e}) = 1 + e - i\right\}. \end{aligned}$$

Thus, we want to find $P\left(B_A(S_H^{1+e}) = 1 + e - i\right)$. To proceed, we first find the probability $P(B_A(r) = k)$ for any realization $r$ of the random escrow length $S_H^{1+e}$ and any possible value of the attacker chain length $k$ as of the time the honest chain completes the escrow period. Then, we will integrate over all possible realizations of $r$ according to the probability distribution of $S_H^{1+e}$. The attacker's chain is $Poisson(A)$ and $S_H^{1+e}$ is distributed $Gamma(1+e,1)$, so that:

$$\begin{aligned} P(B_A(S_H^{1+e})=k) &= \int_0^\infty P(B_A(r)=k \mid S_H^{1+e}=r)\cdot P(S_H^{1+e}=r)\,dr \\ &= \int_0^\infty \frac{(Ar)^k\cdot\exp(-Ar)}{k!}\cdot\frac{r^e\cdot\exp(-\mathrm{r})}{\Gamma(1+e)}\,dr \\ &= \frac{A^k}{k!\,e!}\cdot\frac{\Gamma(k+e)}{(1+A)^{k+e}}\int_0^\infty r\cdot\frac{(1+A)^{k+e}\cdot r^{k+e-1}\cdot\exp(-(1+A)r)}{\Gamma(k+e)}\,dr \\ &= \frac{(k+e)!}{k!\,e!}\left(\frac{A}{1+A}\right)^k\left(\frac{1}{1+A}\right)^{1+e}. \end{aligned}$$

The second equality exploits the independence of $B_A(s)$ and $S_H^{1+e}$ (inherited from the independence of $B_A$ and $B_H$) and substitutes the expressions for the respective Poisson and Gamma densities. The third equality moves terms out of the integral and multiplies and divides by $\frac{\Gamma(k+e)}{(1+A)^{k+e}}$, so that the integrand is exactly the first moment of $Gamma(k+e,1+A)$. The expression for the mean is well known: $\frac{k+e}{1+A}$. The fourth equality substitutes this expression and simplifies. Hence, plugging in $k=1+e-i$, the probability of an attacker deficit $i$ at the time the honest chain completes the escrow period is:

$$P(D^{1+e}=i)=\frac{(1+2e-i)!}{(1+e-i)!\,e!}\left(\frac{A}{1+A}\right)^{1+e-i}\left(\frac{1}{1+A}\right)^{1+e}.$$

Substituting these three expressions into that of $t(A,e)$, we have

$$t(A,e)=(1+e)+\left[\sum_{i=0}^{1+e}\left(\frac{i+1}{A-1}\right)\cdot\frac{(1+2e-i)!}{(1+e-i)!\,e!}\left(\frac{A}{1+A}\right)^{1+e-i}\left(\frac{1}{1+A}\right)^{1+e}\right]$$

obtaining expression (4) in the text as required.

To complete the proof let us consider the limits as $A\to_+ 1$ and $A\to\infty$. Define $f(A,e)$ as the bracketed expression above,

$$f(A,e)\equiv\left[\sum_{i=0}^{1+e}\left(\frac{i+1}{A-1}\right)\cdot\frac{(1+2e-i)!}{(1+e-i)!\,e!}\left(\frac{A}{1+A}\right)^{1+e-i}\left(\frac{1}{1+A}\right)^{1+e}\right],$$

such that $t(A,e)$ takes the form:

$$t(A,e)\equiv(1+e)+f(A,e).$$

First, consider the limit $\lim_{A\to\infty} t(A,e)$. Observe that each term in $f(A,e)$ either goes to 0 or is bounded by a constant. The first and fourth terms go to 0 in the limit: $0\le\lim_{A\to\infty}\left(\frac{i+1}{A-1}\right)\le$

$\lim_{A\to\infty}\left(\frac{2+e}{A-1}\right)=0$ and $\lim_{A\to\infty}\left(\frac{1}{1+A}\right)^{1+e}=0$. The second and third terms are bounded by a constant: $\frac{(1+2e-i)!}{(1+e-i)!\,e!}$ is constant in $A$ and $\lim_{A\to\infty}\left(\frac{A}{1+A}\right)^{1+e-i}\leq 1$. Hence, the product of these terms is 0 in the limit, so $\lim_{A\to\infty} t(A,e)=(1+e)+0=1+e$ as desired.

Second, consider the limit $\lim_{A\to_{+}1} t(A,e)$. The first term in $f(A,e)$ goes to $\infty$ in the limit while all other terms are strictly positive and bounded below. Formally, for the first term, $\lim_{A\to_{+}1}\left(\frac{i+1}{A-1}\right)\geq\lim_{A\to_{+}1}\left(\frac{1}{A-1}\right)=\infty$. For the other terms: $\frac{(1+2e-i)!}{(1+e-i)!\,e!}>0$ is constant in $A$; $\lim_{A\to_{+}1}\left(\frac{A}{1+A}\right)^{1+e-i}=\left(\frac{1}{2}\right)^{1+e-i}>0$; and $\lim_{A\to_{+}1}\left(\frac{1}{1+A}\right)^{1+e}=\left(\frac{1}{2}\right)^{1+e}>0$. Hence, the product of these terms goes to infinity in the limit, so $\lim_{A\to_{+}1} t(A,e)=\infty$ as desired.

## B.2 Numerical Analysis of Cost-Minimizing Attacker Majority

The gross cost of attack, for an attacker with majority $A>1$ and an attack that takes $t$ time in expectation, is defined as $At\cdot N^{*}c$. Proposition 5 provides an explicit formula for $t(A,e)$, the expected duration of a double-spending attack as a function of the attacker majority $A$ and the escrow period $e$.

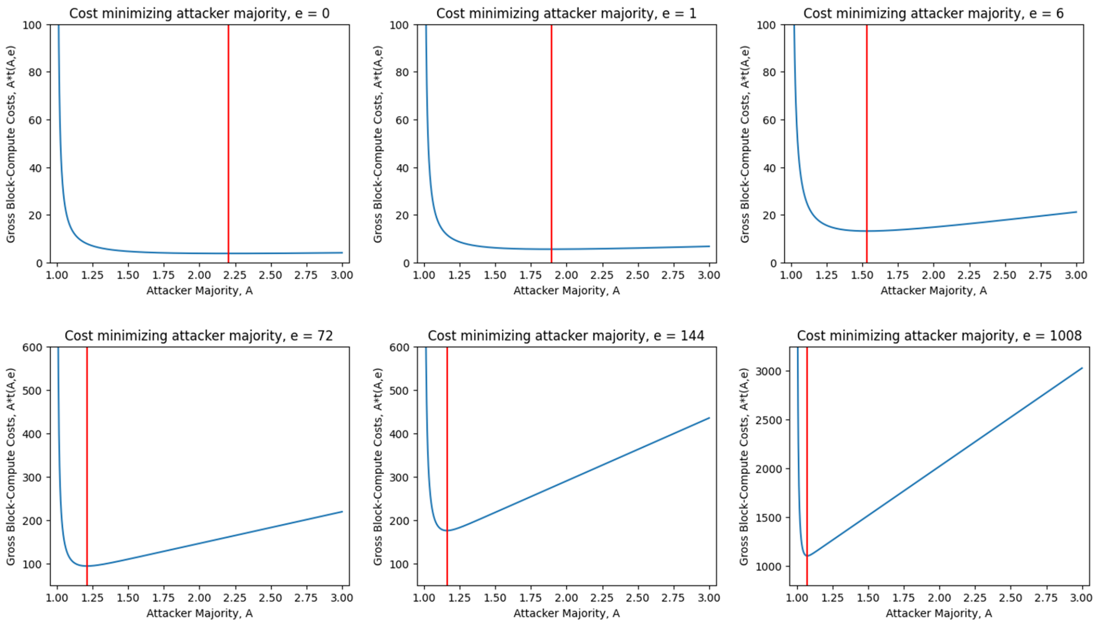
In this section of the Appendix, we use this definition and formula to numerically study the cost-minimizing attacker majority $A$ as a function of the escrow period $e$.

Formally, the gross-cost-minimization problem is given by:

$$\begin{aligned} A^{*}(e) &:= \arg\min_{A} A\cdot t(e,A)\\ &= \arg\min_{A} A\cdot(1+e)+A\cdot\left[\sum_{i=0}^{1+e}\left(\frac{i+1}{A-1}\right)\cdot\frac{(1+2e-i)!}{(1+e-i)!\,e!}\left(\frac{A}{1+A}\right)^{1+e-i}\left(\frac{1}{1+A}\right)^{1+e}\right]. \end{aligned}$$

While this minimization problem is not analytically tractable, it is straightforward to solve numerically. Figure 4 plots the cost of attack for a variety of escrow periods, as well as the cost-minimizing $A^{*}(e)$.

Figure 4: Attacker Gross Cost Minimization



*Notes:* The gross cost of attack as a function of majority $A$ is in blue, plotted as $A \cdot t(A, e)$. As discussed in the main text, this quantity needs to be multiplied by equilibrium per-block compute costs $N^* c$ to obtain gross costs in dollars. The gross-cost-minimizing attacker majority $A^*(e)$ is denoted in red, and is obtained via scipy.optimize.minimize_scalar, a numerical solver in Python.

Intuitively, an attacker majority that is too large will mine more blocks than is necessary for the attack to succeed, whereas an attacker majority that is too close to $A \approx 1$ will, as shown in Proposition 5, have an attack duration that converges to infinity, and hence also be more expensive than is optimal. Because the double-spending attack must be at least as long as the escrow length, the cost-minimizing choice of $A^*(e)$ decreases as the escrow length increases. The longer the escrow length is, the more sure a large majority is to mine more blocks than is necessary, by simple law-of-large numbers reasoning.

Table 4 provides the cost-minimizing majority $A^*(e)$, the duration of attack at this attacker majority $t(A^*(e), e)$, and the total gross cost of attack at this attacker majority $A^*(e) \cdot t(A^*(e), e)$ for a variety of escrow periods.

Table 4: Optimal Attacker Majority, Duration and Compute Costs

| | $e = 0$ | $e = 1$ | $e = 6$ | $e = 72$ | $e = 144$ | $e = 1008$ |
|---|---|---|---|---|---|---|
| $A^*(e)$ | 2.21 | 1.89 | 1.53 | 1.21 | 1.16 | 1.07 |
| $t(A^*(e), e)$ | 1.70 | 2.92 | 8.57 | 77.53 | 151.2 | 1,024.4 |
| $A^*(e) \cdot t(A^*(e), e)$ | 3.74 | 5.53 | 13.14 | 93.88 | 175.6 | 1,100.1 |

*Notes:* $A^*(e)$ is solved numerically as described in the text. The expected duration of attack then follows from Proposition 5. Gross compute costs are in units of equilibrium per-block compute costs $N^*c$.

As before, the duration of attack is given in block units of the honest miner, and the total gross cost of attack is given in compute units ($N^*c$). Note that even very large escrow periods induce a cost-minimizing majority larger than 51% — for example, an escrow period of 1000 blocks induces an optimal attacker majority of $A = 1.07$, or 51.7%.

# C Selected 51% Attacks, Crypto Thefts and Crypto Collapses to Date

To date, there has not been a majority attack on the largest cryptocurrencies such as Bitcoin or Ethereum. There have been several majority attacks on smaller cryptocurrencies, including forks off of Bitcoin (Bitcoin Gold, Bitcoin SV, Bitcoin Cash ABC) and Ethereum (Ethereum Classic). A list of such attacks is provided as Appendix Table C.1. Of the attacks for which the amount stolen was reported, the largest such attack to date was for $18.6 million against Bitcoin Gold in May 2018. This amount represents about 74% of average daily transaction volume in the week prior to the attack. The longest such attack to date was against Ethereum Classic, reportedly 7,000 blocks or about a full day's worth of blocks at typical mining speeds for Ethereum Classic. For all of these attacks of forks off of Bitcoin and Ethereum, there was speculation in the crypto press that the attacker's motive was at least partly to sabotage the coin as opposed to stealing funds, but the details are fairly thinly reported.

There have been many attacks on cryptocurrency financial entities that are based on exploiting flawed code, compromising private keys, manipulating prices of thinly traded tokens, or taking temporary control of a project's governance. These are compiled as Appendix Table C.2. Many of these attacks have been for in excess of $100 million with several in excess of $500 million. The April 2022 attack on Beanstalk Farms for $182 million is interesting in the context of this paper because the attack vector involved the attacker borrowing funds (using what is known as a flash loan) to take temporary majority control of the token, which the attacker then used to vote for a resolution that drained the project's funds into accounts controlled by the attacker. The cost of borrowing majority control is a flow, and the attacker drained the project of its full stock of tokens.

This paper's model analyzes collapse risk as a source of security, albeit a precarious one. While none of the major cryptocurrencies themselves have collapsed to date, many cryptocurrency projects and financial entities have indeed collapsed. A list is compiled as Appendix Table C.3. These collapses may lend plausibility to this paper's hypothesis, explored in Section 5, that collapse risk explains why the major cryptocurrencies have not yet been attacked given the analysis of Sections 3-4.

Interested readers may also wish to consult Molly White's lists of collapses, scams and hacks available at https://web3isgoinggreat.com/charts/top.

Table C.1: 51% Attacks of Bitcoin and Ethereum Forks

| Name of Coin | Date of Attack | Amount Stolen | Length of Largest Reorganization | Market Cap at Time of Attack | Market Cap at Peak |
|---|---|---|---|---|---|
| Bitcoin SV | 8/3/2021 | Unknown | Unknown | $2.7 billion | $8.3 billion |
| | 6/24/2021 | Unknown | Unknown | $2.4 billion | |
| Ethereum Classic | 8/29/2020 | Unknown | 7,000 Blocks | $760 million | $15.6 billion |
| | 8/6/2020 | $1.7 million | 4,200 Blocks | $840 million | |
| | 8/1/2020 | $5.6 million | 3,700 Blocks | $860 million | |
| | 1/5/2019 | $1.1 million | 140 Blocks | $560 million | |
| Bitcoin Gold | 1/23/2020 | $72 thousand | 16 Blocks | $190 million | $7.6 billion |
| | 5/16/2018 | $18.6 million | 22 Blocks | $1.0 billion | |

Sources: Bloomberg, CCN, Coinbase, CoinDesk, Cointelegraph, GitHub, Twitter, and https://dci.mit.edu/51-attacks. For a list of all articles consulted with URLs please see the author's website or the paper's online appendix. For interpreting length of longest reorganization, note that for Bitcoin SV and Bitcoin Gold there are typically 6 blocks per hour. For Ethereum Classic there are typically 275 blocks per hour. Amount Stolen is based on press reports of the dollar value stolen where available or based on press reports of the amount of cryptocurrency stolen converted into dollars using price data from CoinMarketCap. Market cap data is from CoinMarketCap. The market cap at the time of attack is based on the average of the 7 days prior to the date of attack.

Table C.2: Attacks of Crypto Financial Entities

| Name | Type of Business | Date of Attack | Amount Stolen | Attack Vector |
|---|---|---|---|---|
| Poloniex | Centralized Exchange | November 2023 | $120 Million | Unknown |
| Mixin Network | Decentralized Exchange and Lending Protocol | September 2023 | $200 Million | Compromised Cloud Database |
| Multichain | DeFi Bridge | July 2023 | $230 Million | Compromised Private Keys or Rug Pull |
| Euler Finance | DeFi Lending Protocol | March 2023 | $197 Million | Flash Loan Attack + Flawed Code |
| FTX | Centralized Exchange | November 2022 | $477 Million | Compromised Private Keys |
| | | March-April 2021 | $600 Million | Price Manipulation |
| Mango Markets | Decentralized Exchange | October 2022 | $100 Million | Price Manipulation |
| BNB Chain | DeFi Bridge | October 2022 | $568 Million | Flawed Code |
| Wintermute | DeFi Market Maker | September 2022 | $160 Million | Compromised Hot Wallet |
| Nomad | DeFi Bridge | August 2022 | $190 Million | Flawed Code |
| Horizon Bridge | DeFi Bridge | June 2022 | $100 Million | Compromised Private Keys + Governance Control |
| Beanstalk Farms | DeFi Stablecoin | April 2022 | $182 Million | Flash Loan Attack + Governance Control |
| Ronin Network | DeFi Bridge | March 2022 | $625 Million | Compromised Private Keys + Governance Control |
| Wormhole | DeFi Bridge | February 2022 | $320 Million | Flawed Code |
| Qubit Finance | DeFi Lending Protocol | January 2022 | $80 Million | Flawed Code |
| BitMart | Centralized Exchange | December 2021 | $150 Million | Compromised Private Keys |
| C.R.E.A.M. Finance | DeFi Lending Protocol | October 2021 | $130 Million | Flash Loan Attack + Price Manipulation |
| PolyNetwork | DeFi Bridge | August 2021 | $600 Million | Flawed Code |
| KuCoin | Centralized Exchange | September 2020 | $281 Million | Compromised Private Keys |
| BitGrail | Centralized Exchange | February 2018 | $170 Million | Unknown |
| Coincheck | Centralized Exchange | January 2018 | $530 Million | Unknown |
| The DAO | Decentralized Venture Capital | June 2016 | $55 Million | Flawed Code |
| Mt. Gox | Centralized Exchange | 2011-2014 | $480 Million | Compromised Private Keys |

Sources: BitMart, Bloomberg, Chainalysis, Coinbase, CoinDesk, CoinMarketCap, Cointelegraph, Elliptic Inc, Forbes, *Going Infinite* by Michael Lewis, Kraken Blog, Mango Markets, Medium Blog, Mixin Network, PolyNetwork, Reuters, The Verge, Twitter, Unchained Crypto, and WSJ. For a list of all articles consulted with URLs please see the author's website or the paper's online appendix. Amount Stolen is based on press reports of the dollar value stolen where available or based on press reports of the amount of cryptocurrency stolen converted into dollars using price data from CoinMarketCap.

Table C.3: Collapses of Crypto Financial Entities

| Name | Type of Business | Date of Collapse | Entity Size |
|---|---|---|---|
| Genesis | Lending Firm | January 2023 | \$5.1 billion |
| BlockFi | Lending Firm | November 2022 | \$3.9 billion |
| FTX | Centralized Exchange | November 2022 | \$8.9 billion - \$32 billion |
| Three Arrows Capital | Hedge Fund | June 2022 | \$3.4 billion - \$10 billion |
| Voyager | Lending Firm | July 2022 | \$5.8 billion |
| Celsius | Lending Firm | July 2022 | \$5.5 billion - \$19.1 billion |
| Terra + Luna | Blockchain + Stablecoin | May 2022 | \$40 billion |
| Africrypt | Investment Firm | April 2021 | \$3.6 billion |
| Thodex | Centralized Exchange | April 2021 | \$2 billion |
| Mt. Gox | Centralized Exchange | February 2014 | \$480 million |

Sources: BlockFi Blog, Bloomberg, Chainalysis, CoinDesk, Cointelegraph, Forbes, PR Newswire, Reuters, WSJ, and bankruptcy filings. For Celsius: the \$19.1 billion figure is Celsius's balance sheet size as of August 13, 2021 based on an investment memorandum provided to the author and reported in the Wall Street Journal in June 2022; the \$5.5 billion figure is Celsius's balance sheet size when it filed for bankruptcy. For FTX: the \$8.9 billion figure is FTX's customer assets at the time of its bankrupty filing; the \$32 billion is its peak market valuation. For Three Arrows Capital: the \$10 billion figure is based on reports of the hedge funds size as of March 2022 and the \$3.4 billion figure is based on reports of claims from creditors. For Mt. Gox: the \$480 million figure is the reported value of customer Bitcoins reported lost by the exchange when it filed for bankruptcy. For a list of all articles consulted with URLs please see the author's website or the paper's online appendix.