

# Online Appendix for “Trust at Scale: The Economic Limits of Cryptocurrencies and Blockchains”

Eric Budish

*Quarterly Journal of Economics*

## Contents

**Appendix A. Discussion of Responses to this Paper’s Argument**

**Appendix B. Double-Spending Attack Technical Appendix**

**Appendix C. Selected 51% Attacks, Crypto Thefts and Crypto Collapses to Date**

## **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 et al. (2021) and many practitioners that we should compare Nakamoto’s costs to the costs of market power in traditional finance, which are also high.<sup>1</sup> I hope the present draft of the text makes more clear the conditional nature of

---

<sup>1</sup>See Philippon (2015) and Greenwood and Scharfstein (2013) on high costs of traditional finance, and see Cochrane (2013) for a counterpoint.

the paper’s argument: if Nakamoto trust becomes more economically useful, then it will also have to get 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, as well as the analysis in Section 6, 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.

I have also tried to address the main technical innovation that may seem to address my argument in the main text, which is the idea of proof-of-stake with slashing. This approach is economically attractive because it mimics the traditional trust model of collateral plus rule-of-law, so the cost of attack is not the flow cost of trust-support but the stock value of the attacker’s collateral. This is not quite as cost-effective as the use of collateral in the traditional financial system backed by rule-of-law, which costs zero under the assumptions of the Modigliani-Miller theorem, but it is a 2500x improvement over the costs of Bitcoin’s security model as shown in Section 5. Unfortunately, as discussed in the main text, impossibility theorems of Tas et al. (2023) and Budish, Lewis-Pye, and Roughgarden (2024) show that proof-of-stake with slashing faces technical limitations if the attacker is large enough or if the communications network is not sufficiently reliable. The rough intuition is that a large enough attacker can thwart the protocol’s legal system. Thus, the present paper’s model is a lens through which we can understand the economic goals of Ethereum’s adoption of proof-of-stake with slashing, while the related computer science research helps us understand the conditions under which it works and its technical limitations.

In the remainder of 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 Community Response

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.<sup>2</sup>

The community response argument seems valid as an argument that attacks might be more

---

<sup>2</sup>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.

expensive or difficult to execute than is modeled in Sections 3-4, but it raises four 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, a hard fork harms honest holders of specialized capital too.

Third, after a hard fork, the blockchain goes from equilibrium constraint (10) to equilibrium constraint (3), so it is vulnerable to repeated attack. This point is made in Buterin’s 2016 blog post quoted in the main text.

Fourth, 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.2 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. 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.3 Counterattacks

Moroz et al. (2020) extend the analysis in Budish (2018) to enable the victim of a double-spending 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  $\{1, 2, \dots, 7\}$  at the time the honest chain is replaced, and the attacker chain that replaces it is  $\{1', 2', \dots, 7', 8'\}$ . If the Defender can quickly organize a majority of their own, then they can build off of the  $\{1, 2, \dots, 7\}$  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  $\{1, 2, \dots, 10\}$  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.

#### A.4 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 \$25 instead of \$25,000. To secure against a \$100 billion attack would require costs per transaction of \$2,500 instead of \$2.5 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 "Layer 2." In this paradigm, the Bitcoin blockchain ("Layer 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.<sup>3</sup>

## A.5 Modification to Nakamoto II: Tweak Longest-Chain Rule

The discussion above in A.1 expressed skepticism about the “community” response to the logic of this paper. However, what about modifying the longest-chain rule to try to encode what the community would *want* to do in the event of an attack.

The modification to the longest-chain rule 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 rule’s determination of the sequence of events.

One pursuit along these lines is Leshno et al. (2023). Their approach, which they call “Stubborn Nakamoto,” is fully secure against double-spending attacks but, instead, has to permanently

---

<sup>3</sup>I thank Neha Narula for several helpful conversations about this approach.

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).

## B Double-Spending Attack Technical Appendix

### B.1 Proof of Proposition 3 (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:

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

The attack is completed when both (i) the honest chain has mined at least  $k+e$  blocks, therefore passing the attacker transactions' escrow periods, 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  $A$ , escrow period  $e$ , and number of blocks in which the attacker places transactions  $k$ , is given by the stopping time formula:

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

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

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

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  $D^{k+e}$ :

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

If the realization of  $D^{k+e} < 0$ , 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  $D^{k+e} \geq 0$ , 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  $t(A, e, k)$  based on the sign of  $D^{k+e}$  for a tractable expression for  $t(A, e, k)$ :

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

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

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

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

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  $P(D^{k+e} = i)$ . Recall  $D^{k+e}$  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:

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

Thus, we want to find  $P(B_A(S_H^{k+e}) = k + e - i)$ . To proceed, we first find the probability  $P(B_A(r) = l)$  for any realization  $r$  of the random escrow length  $S_H^{k+e}$  and any possible value of the attacker chain length  $l$  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  $S_H^{k+e}$ .

The attacker's chain is  $Poisson(A)$  and  $S_H^{k+e}$  is distributed  $Gamma(k+e, 1)$ , so that:

$$\begin{aligned}
P(B_A(S_H^{k+e}) = l) &= \int_0^\infty P(B_A(r) = l \mid S_H^{k+e} = r) \cdot P(S_H^{k+e} = r) \, dr \\
&= \int_0^\infty \frac{(Ar)^l \cdot \exp(-Ar)}{l!} \cdot \frac{r^{k+e-1} \cdot \exp(-r)}{\Gamma(k+e)} \, dr \\
&= \frac{A^l}{l! (k+e-1)!} \cdot \frac{\Gamma(l+k+e-1)}{(1+A)^{l+k+e-1}} \int_0^\infty r \cdot \frac{(1+A)^{l+k+e-1} \cdot r^{l+k+e-2} \cdot \exp(-(1+A)r)}{\Gamma(l+k+e-1)} \, dr \\
&= \frac{(l+k+e-1)!}{l! (k+e-1)!} \left(\frac{A}{1+A}\right)^l \left(\frac{1}{1+A}\right)^{k+e}.
\end{aligned}$$

The second equality exploits the independence of  $B_A(s)$  and  $S_H^{k+e}$  (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  $\frac{\Gamma(l+k+e-1)}{(1+A)^{l+k+e-1}}$ , so that the integrand is exactly the first moment of  $Gamma(l+k+e-1, 1+A)$ . The expression for the mean is well known:  $\frac{l+k+e-1}{1+A}$ . The fourth equality substitutes this expression and simplifies. Hence, plugging in  $l = k+e-i$ , the probability of an attacker deficit  $i$  at the time the honest chain completes the escrow period is:

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

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

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

obtaining expression (6) in the text as required.

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

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

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

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

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

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

Second, consider the limit  $\lim_{A \rightarrow +1} t(A, e, k)$ . The first term in  $f(A, e, k)$  goes to  $\infty$  in the limit while all other terms are strictly positive and bounded below. Formally, for the first term,  $\lim_{A \rightarrow +1} \left( \frac{i+1}{A-1} \right) \geq \lim_{A \rightarrow +1} \left( \frac{1}{A-1} \right) = \infty$ . For the other terms:  $\frac{(2(k+e)-1-i)!}{(k+e-i)!(k+e-1)!} > 0$  is constant in  $A$ ;  $\lim_{A \rightarrow +1} \left( \frac{A}{1+A} \right)^{k+e-i} = \left( \frac{1}{2} \right)^{k+e-i} > 0$ ; and  $\lim_{A \rightarrow +1} \left( \frac{1}{1+A} \right)^{k+e} = \left( \frac{1}{2} \right)^{k+e} > 0$ . Hence, the product of these terms goes to infinity in the limit, so  $\lim_{A \rightarrow +1} t(A, e, k) = \infty$  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 3 provides an explicit formula for  $t(A, e, k)$ , the expected duration of a double-spending attack as a function of the attacker majority  $A$ , the escrow period  $e$ , and the number of blocks in which the attacker places transactions  $k$ .

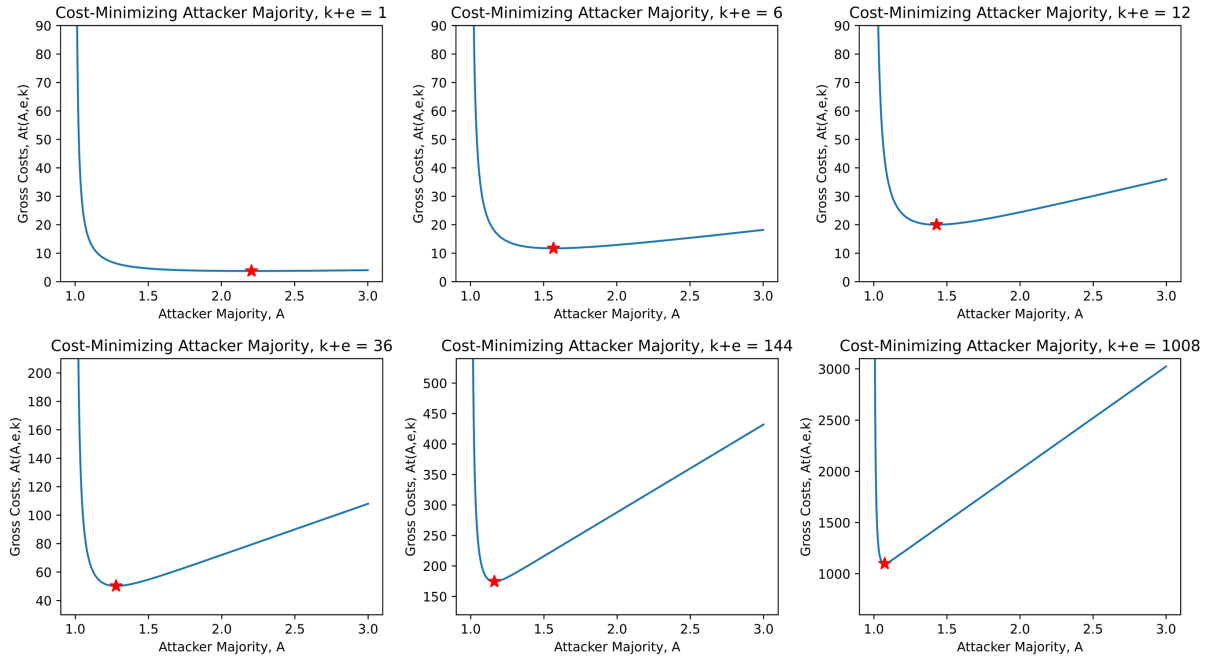
In this section of the Appendix, we use this definition and formula to numerically study the cost-minimizing attacker majority  $A$  as a function of  $k+e$ .

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

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

While this minimization problem is not analytically tractable, it is straightforward to solve numerically. Figure B.1 plots the cost of attack for a variety of values of  $k+e$ , as well as the cost-minimizing  $A^*(k+e)$ .

Figure B.1: Attacker Gross Cost Minimization



*Notes:* The gross cost of attack as a function of majority  $A$  is in blue, plotted as  $A \cdot t(A, e, k)$ . As discussed in the main text, this quantity needs to be multiplied by equilibrium per-block trust-support costs  $N^*c$  to obtain gross costs in dollars. The point representing the gross-cost-minimizing attacker majority  $A^*(k + e)$  is marked by a red star, 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 3, 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 number of blocks double spend plus the escrow length, the cost-minimizing choice of  $A^*(k + e)$  decreases as  $k + e$  increases. The larger is  $k + e$ , the more sure a large majority is to mine more blocks than is necessary, by simple law-of-large numbers reasoning.

Table B.1 provides the cost-minimizing majority  $A^*(k + e)$ , the duration of attack at this attacker majority  $t(A^*(k + e), e, k)$ , and the total gross cost of attack at this attacker majority  $A^*(k + e) \cdot t(A^*(k + e), e, k)$  for a variety of values of  $k + e$ .

Table B.1: Optimal Attacker Majority, Duration and Gross Costs

	# Blocks of Double Spending + Escrow Period ( $k + e$ )					
	1	6	12	36	144	1,008
$A^*(k + e)$	2.21	1.57	1.43	1.28	1.16	1.07
$t(A^*(k + e), e, k)$	1.70	7.47	13.99	39.28	150.21	1,023.37
$A^*(k + e) \cdot t(A^*(k + e), e, k)$	3.74	11.70	20.00	50.21	174.44	1,099.02

*Notes:*  $A^*(k + e)$  is solved numerically as described in the text. The expected duration of attack then follows from Proposition 3. Gross costs are in units of equilibrium per-block trust-support costs  $N^*c$ .

As before, the gross cost of attack is given in units of per-block trust-support costs  $N^*c$ . Note that even very large escrow periods induce a cost-minimizing majority larger than 51%—for example, the case  $k + e = 1008$  blocks (1 week) 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 thinly reported.

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: Bitcoin Gold Forum, Bloomberg, CCN, Coinbase, CoinDesk, Cointelegraph, Decrypt, 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. Market cap data is from CoinMarketCap. The market cap at the time of attack is based on the day prior to the date of attack.

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.

This paper’s model analyzes collapse risk as a potential source of security (Section 5.2.1), albeit an unattractive 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.

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 bankruptcy 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.

## References

- Bailey, Norman T.J.** 1957. “Some Further Results in the Non-Equilibrium Theory of a Simple Queue.” *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 19(2): 326–333.
- Budish, Eric.** 2018. “The Economic Limits of Bitcoin and the Blockchain.” NBER Working Paper No. 24717.
- Budish, Eric, Andrew Lewis-Pye, and Tim Roughgarden.** 2024. “The Economic Limits of Permissionless Consensus.” In *Proceedings of the 25th ACM Conference on Economics and Computation (EC '24)*. Forthcoming. Available at <https://arxiv.org/abs/2405.09173>.
- Cochrane, John H.** 2013. “Finance: Function Matters, Not Size.” *Journal of Economic Perspectives*, 27(2): 29–50.
- Greenwood, Robin, and David Scharfstein.** 2013. “The Growth of Finance.” *Journal of Economic Perspectives*, 27(2): 3–28.
- 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.
- 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>.
- 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).
- Philippon, Thomas.** 2015. “Has the US Finance Industry Become Less Efficient? On the Theory and Measurement of Financial Intermediation.” *American Economic Review*, 105(4): 1408–1438.
- 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*.

**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](https://s29.q4cdn.com/385744025/files/doc_downloads/Visa-Inc_-Fiscal-2021-Annual-Report.pdf).