ECONOMIC FINDING • JULY 2022

The Economic Limits of Bitcoin and Anonymous, Decentralized Trust on the Blockchain

Based on BFI Working Paper 2022-83, "The Economic Limits of Bitcoin and Anonymous, Decentralized Trust on the Blockchain," by Eric Budish, Chicago Booth

Bitcoin's new form of decentralized trust suffers from a pick-your-poison conundrum with two possible outcomes: Either this trust is extremely expensive relative to its economic usefulness, or it is vulnerable to collapse.

Satoshi Nakamoto, the creator of Bitcoin, invented a new form of trust without the need for the rule of laws, reputations, relationships, collateral, or trusted intermediaries that govern mainstream financial systems. Nakamoto did this by combining ideas from computer science and economics to incentivize a large, anonymous, decentralized, freely entering and exiting mass of compute power around the world to collectively monitor and maintain a common data set, and thus enabling trust in this data set. The specific data structure maintained by this large mass of compute power is called a blockchain.

This paper argues that while this new trust is clearly ingenious, it suffers from a pick-your-poison conundrum with two possible outcomes: Either this new form of trust is extremely expensive relative to its economic usefulness, or it is vulnerable to collapse. On the first count—the high cost of this new trust—Budish presents three equations. Very broadly summarized, the first equation says that the dollar amount of compute power devoted to maintaining trust is equal to the dollar value of compensation to miners. For a sense of magnitudes, in 2022 through early June, this compensation has averaged about $250,000 per block of data, or about $40 million per day.

The second equation addresses the key vulnerability to Nakamoto's form of trust—a "majority attack." Nakamoto's method for creating an anonymous, decentralized consensus about the state of a dataset relies on most of the computing power devoted to maintaining the data to behave honestly. In other words, it must not be economically profitable for a potential attacker to acquire a 51% majority (or greater) of the compute power. The *cost* of such an attack must exceed the *benefits* of an attack.

Before describing the third equation, let's pause to consider the terms "stock" and "flow," which economists use when describing variables like, say, a bank balance at a particular point in time (stock), vs. the amount of interest earned over time (flow). In this case, the recurring payments to miners to maintain honest compute power is a flow (as in equation one), while the value of attacking the system at any given time is a stock (equation two). To illustrate, imagine a Main Street bank that must secure the money in the building on any given day. The daily wages of the security guards protecting the bank are a flow, and the money in the bank on any given day is the stock.

The third equation, then, tells us that the flow-like costs of maintaining trust *must exceed* the stock-like value of breaking the trust. The key to understanding this trust is that it is memoryless, which means that Nakamoto's trust is only as secure as the amount of compute power devoted to maintaining it during a given unit of time. Likewise, a big attack at a low-secure moment puts Bitcoin at jeopardy.

One way to understand this idea of memoryless trust is to consider the amount of security that your bank provides for your financial accounts on a given day, let's call it Wednesday. You benefit from all the security features implemented by your bank in the previous days, weeks, months, and years—as well as from laws, regulations, and reputational incentives—and that security stays in place 24/7. You should be no more worried about your accounts on Wednesday as you were on Tuesday, or you will be on Thursday, and so on.

Nakamoto's system of trust has no built-in "memory," but is only as good as the amount of compute power dedicated to maintaining that trust on that Wednesday, and then again on Thursday, and so on. Each day starts anew. If this were the case for your bank, it would mean that its daily security budget would have to be large relative to the whole value of attacking it. Again: the flow-like costs of maintaining trust *must exceed* the stock-like value of breaking the trust. Moreover, the costs for Nakamoto's system of trust scale linearly, so if an attack becomes 1,000 times more attractive, that means 1,000 times more compute power must be spent to secure trust. Or, to return to our Main Street bank example, if there is suddenly 1,000 times more money in the bank, bank management would need 1,000 times more security guards. As Budish bluntly states: "This is a very expensive form of trust!"

Regarding Budish's second poison—the system's vulnerability to collapse—let us first consider the nature of the computers that secure trust in Bitcoin. These are not ordinary computers (as Nakamoto first envisioned), like the ones on our desks and laps, but rather machines with highly specialized chips that are dedicated to Bitcoin mining. They are very good at this task, they operate quickly, and they are essentially useless for any other function. Likewise, if an attack causes collapse of the system, it will render those machines nearly worthless. This recasts the attacker cost model: In addition to charging the attacker the flow cost, the attacker must also be charged for the decline in the value of their specialized capital, which makes the attacker's cost more like a stock (expensive!) than a flow (much cheaper), and thus makes the blockchain more secure.

So, if this attacker cost model is correct in describing why Bitcoin has not yet been majority attacked, then what changes to the environment could cause incentives to flip and lead to a majority attack? Budish's analysis yields three main attack scenarios, with the first two describing instances when the cost of an attack changes from an expensive stock cost to a relatively cheaper flow cost. First, changes could occur in the market for the specific technology used for Bitcoin mining; for example, a chip glut, including for previous generation "good enough" chips, would make attack costs more like a flow than a stock. Second, a large enough fall in the rewards to mining due to a decline in either the value of Bitcoin or the number of Bitcoins awarded to successful miners would lead to mothballing a large amount of specialized mining equipment. If more than 50 percent of capital is mothballed for a sufficiently long period of time, this would raise the vulnerability to attack on two counts: Economically, the opportunity cost of using otherwise-mothballed equipment to attack is very low; and logistically, large amounts of mothballed equipment might make an attack easier to execute. This, again, would make the opportunity cost of attack more like a flow than a stock. And third, Budish describes a scenario with a large increase in the economic usefulness of Bitcoin, (without a commensurate increase in the rewards to miners), thus incentivizing an attack.

Bottom line: the cost of securing the blockchain system against attacks can be interpreted as an implicit tax on using Nakamoto's anonymous, decentralized trust, with the level of the tax in dollar terms scaling linearly with the level of security. Numerical calculations suggest that this tax could be significant and preclude many kinds of transactions from being economically realistic.

READ THE WORKING PAPER

NO. 2022-83 · JUNE 2022

The Economic Limits of Bitcoin and Anonymous, Decentralized Trust on the Blockchain

bfi.uchicago.edu/working-paper/202283

ABOUT OUR SCHOLAR

Eric Budish

Paul G. McDermott Professor of Economics and Entrepreneurship and Centel Foundation/ Robert P. Reuss Faculty Scholar, Chicago Booth

chicagobooth.edu/faculty/directory/b/eric-budish

CHICAGO BOOTH
The University of Chicago Booth School of Business