Roughchain DRAFT v0.0.1

Deprecated by Roughchain Whitepaper Draft v0.0.2 👈

Preface For Crypto Skeptics

The cryptocurrency world is full of shifting promises, most of which have yet to be fulfilled. One such promise is financial inclusion. For example, in Argentina—a country in a constant struggle[1], yet home to pioneers in cybersecurity, and cryptocurrency startups[2] and their use—it has been a 'Web2' company, MercadoLibre, through its fintech arm MercadoPago, that has made real progress in expanding financial inclusion[3].

If we play hardball, we could argue that the 'Web3' ecosystem has strayed far from the cypherpunk idealism of decentralized payments that Bitcoin's implementation inaugurated, arriving instead at a space where both small and large-scale frauds are common, along with actors who, while not always breaking the law, erode trust through ethically questionable behavior—ironically undermining the very decentralized trust the system was meant to foster.

With real, multidimensional experience in fields like cybersecurity and having witnessed the rise and fall of countless players—often driven by bad actors—in this space, how is it even possible that I'm here talking about a new Layer 1 blockchain called Roughchain? I've never been in direct contact with so many people who ended up in jail in my life, and they deserved it—it wasn't a political issue but a matter of justice within this ecosystem. Additionally, interacting with people from the Bitcoin Core team, Ethereum cofounders, and other well-known technologies has given me real insights into how different the public perception can be from the private, closer realities.

The problem is that, despite my 10+ years in this business, I come from a background in cybersecurity, research, and development. I kept thinking about foundational areas such as state-machine replication, Byzantine fault tolerance, cryptography, P2P networks, economics[4], etc. It is what it is. I had a non-revolutionary idea more than five years ago that I discarded externally, but it kept flowing in the background, always assuming those who knew much more than me would dismiss it. Yet in 2024, as I re-examine the 'Web3' sector from a macro perspective, I find myself discussing the idea and gathering feedback from a small group of friends. It is time to open the discussion and let Roughchain go if we come to a dead end.

---

[1] Simon Kuznet mentioned four types of countries: developed, underdeveloped, Japan, and Argentina. What more could be said about Argentina and Japan to be classified as such?
[2] Oldest Blockchain Companies
[3] https://fintech.industryexaminer.com/mercado-pago-fintech-innovations/
[4] https://publications.ut-capitole.fr/id/eprint/14924/1/Laffont_14924.pdf

Preface

Does a blockchain, whether public or private, hold the same value for the Internet as other well-known decentralized protocols, such as TCP/IP (Internet), BGP (routing)[5], DNS (domain name servers), SMTP (email), HTTP, and P2P? If the logical answer is "no," then continuing to read about Roughchain is futile, and we can stop here.

However, I believe the answer is "yes." Rather than relying on the usual financial, economic, or cypherpunk motives, I will approach this from a different foundation. This is your warning: I will take an original software engineering perspective. This is not a stretch, but rather a method to discuss blockchain with more precise and logical arguments.

Sadly, blockchain has become an overly charged keyword nowadays. While Satoshi Nakamoto did mention a "chain of blocks" in the Bitcoin paper, it was never framed as the grandiose concept we associate with blockchain today.

But, of course, language evolves. And that's why we're here—continuing the conversation with a fresh perspective on the topic. Obviously, the meaning of words can shift over time, which is exactly why it makes sense to dive into this discussion about Roughchain.

Interoperability[6][7]

The balance between decentralization, centralization, and interoperability has driven progress long before the advent of computers. In this section, we will focus specifically on digital interoperability between entities: applications and systems governed and operated by different organizations, authorities, or participants. It's crucial to avoid immediately connecting interoperability to financial operations linked to blockchains, as this can detract from the core issue at hand: true digital interoperability.

Nick Szabo[8] first introduced the term "smart contracts" in 1995[9] and 1996[10].

Here's where things get interesting.

Traditionally, when two or more entities aim to interoperate, they meet and agree on a specific protocol to achieve their goals. Then, they build software to implement the protocol. If new goals emerge, they must revisit and modify the protocol, restarting the entire process of specification and implementation. This approach is obviously time-consuming and resource-intensive, moving specifications slowly toward working software, and supporting its operation.

This is where smart contracts—though not exactly as Nick Szabo envisioned—make complete sense. They allow entities to specify a protocol in a way that resembles a legal

---

[5] White House asks agencies to step up internet routing security efforts
[6] https://en.wikipedia.org/wiki/Interoperability
[7] https://aws.amazon.com/what-is/interoperability/
[8] https://en.wikipedia.org/wiki/Nick_Szabo
[9] https://ia801806.us.archive.org/24/items/extropy-16/Extropy-16.pdf
[10] Smart Contracts: Building Blocks for Digital Markets

contract, reducing the need for manual software development. By automating the process from specification to execution, and operations, smart contracts save time and resources, making them a powerful tool for interoperability. We are assuming here that a smart contract is an independent concept, not tied exclusively to blockchains. In fact, it’s possible to envision two independent entities using smart contracts to iterate over their technology based on legal agreements quickly and efficiently.

A simple example is airline points. For airlines to make their points compatible and exchangeable, they must agree on a common protocol and navigate each organization's bureaucracy, operations, and software development—a process that can take years, often ending in failure and wasted capital. A smart contract, by specifying the agreement between different organizations, reduces the risks, time, and resources typically required to create such relationships.

This opportunity remains wide open.

As someone who has been interested in cryptography, I’ve had the privilege of collecting signatures from several key figures in the field.

Nick Szabo (signatures at the end from the book)

Long before blockchains and decentralized technologies came to the forefront, foundational work in cryptography was already shaping the digital landscape. One such pioneer was Leonard Adleman, co-creator of the RSA encryption algorithm—a crucial building block for secure, decentralized communication. His work laid the groundwork for much of what we now consider essential in blockchain security and trustless systems.

(Image of Leonard Adleman’s autograph)
Autograph of Leonard Adleman, co-creator of RSA encryption, whose contributions to cryptography helped pave the way for modern blockchain security.

Additionally, my companies were featured in one of the earliest Algorand blocks, the L1 conceived by Silvio Micali, further validating our longstanding dedication to blockchain long before its current widespread recognition. This early involvement underscores my genuine and enduring interest in the decentralized technologies we discuss today.

Introduction

Blockchains are built upon the principles of Byzantine Fault Tolerance (BFT), which ensures that systems can function correctly even when some actors behave maliciously or unpredictably. While many blockchain implementations loosely resemble BFT research, it’s critical to start with a solid understanding of BFT before deviating from these foundations.

A classic BFT implementation is inherently slow and doesn’t scale to meet the high transaction volumes required, for example, for daily payment systems, such as those processed by credit or debit card companies. Unfortunately, very few actors in the blockchain space fully recognize this as a "black-box" problem, and even fewer understand

the fundamental limitations of BFT itself. This ignorance is crucial: if a blockchain technology claims to exceed the state-of-the-art BFT limits, it directly implies that it is compromising other dimensions, such as security or decentralization, to achieve that scalability.

We will not enter into the depths of the blockchain trilemma (2017)[11][12], Zooko's triangle (2001)[13], or CAP theorem (1998)[14] discussions here but will highlight the collusion problem which is central to consensus algorithms even if the specific topic is not often cited rigorously in the general blockchain space as in papers such as “Collusion, efficiency, and dominant strategies”[15]. Consensus is often used as a very simplified umbrella term for several simultaneous factors. If we return to basic BFT research it does not cover the collusion in a political way (e.g. could you bribe a node or a miner?).

Roughchain

A core issue in blockchain technologies is how to order transactions effectively and address the double-spend problem, which is fundamental to Byzantine Fault Tolerance (BFT). Another critical, yet less discussed, issue is the potential for collusion among participants responsible for creating blocks. This game-theoretical problem poses risks to decentralization and fairness in blockchain technologies.

The Roughchain architecture is relatively simple, consisting of just two high-level components:

1. A set of public servers that perform signed timestamping of transactions with precise time accuracy. These servers aim to be "collusion-free", highly resistant to collusion by optionally using multiple servers per transaction, and hierarchies.
2. A P2P gossip network that accepts only transactions passing through a filter. These transactions are then propagated using a gossip protocol while the ledger is built on top of it.

We view this as an elegant architecture where the P2P network functions as a ledger, clearly separating concerns and addressing distinct problems. The "firewalled" behavior of the P2P nodes minimizes the number of transactions entering the network, reducing the load on memory pools. Meanwhile, the timestamping signing services handle a separate, simpler task, ensuring accurate transaction timestamps without overburdening the system.

Signing Servers

Who can sign transactions before they enter the P2P Gossip Network to protect its integrity? To safeguard the network, we propose a governed list of signers. Since signing transactions is an almost instantaneous process, we can enhance security by requiring multiple signatures. Additionally, we could mandate that opposing parties sign the same transaction

[11] https://vitalik.eth.limo/general/2017/12/31/sharding_faq.html

[12] https://algorandtechnologies.com/news/silvio-micali-lex-fridman-algorand-and-the-blockchain-trilemma

[13] https://en.wikipedia.org/wiki/Zooko%27s_triangle

[14] https://en.wikipedia.org/wiki/CAP_theorem

[15] https://www.sciencedirect.com/science/article/pii/S0899825616300136

to further minimize the risk of collusion, ensuring that no single group has too much influence over the network's transaction flow.

Boosting Roughchain with a Hack!

…
…
…