crypto.com

Exploring Selfish Mining Attacks and Blockchain & Database Dichotomy

April 2022

Research and Insights

Head of Research and Insights
Henry Hon PhD, CFA

Senior Research Analyst
William Wu PhD

Research Intern
Bowen Liu

Published on 18 Apr 2022

RESEARCH DISCLAIMER

This report alone must not be taken as the basis for investment decisions. Users shall assume the entire risk of any use made of it. The information provided is merely complementary and does not constitute an offer, solicitation for the purchase or sale of any financial instruments, inducement, promise, guarantee, warranty, or an official confirmation of any transactions or contract of any kind.

The views expressed herein are based solely on information available publicly, internal data, or information from other reliable sources believed to be true. This report includes projections, forecasts, and other predictive statements that represent Crypto.com's assumptions and expectations in light of currently available information. Such projections and forecasts are made based on industry trends, circumstances, and factors involving risks, variables, and uncertainties. Opinions expressed herein are our current opinions as of the date appearing on the report only.

No representations or warranties have been made to the recipients as to the accuracy or completeness of the information, statements, opinions, or matters (express or implied) arising out of, contained in, or derived from this report or any omission from this document. All liability for any loss or damage of whatsoever kind (whether foreseeable or not) that may arise from any person acting on any information and opinions contained in this report or any information made available in connection with any further enquiries, notwithstanding any negligence, default, or lack of care, is disclaimed.

This report is not meant for public distribution. Reproduction or dissemination, directly or indirectly, of research data and reports of Crypto.com in any form is prohibited except with the written permission of Crypto.com. Persons into whose possession this report may come are required to observe these restrictions.

Contents

Executive Summary

This article introduces the key points of two highlighted research papers on selfish mining attacks and a dichotomy between blockchains and databases.

Selfish Mining Attacks Exacerbated by Elastic Hash Supply

Selfish mining is a deceitful mining strategy on Proof-of-Work (PoW) blockchains in which one miner (or a group) mines a block, withholds it privately, and eventually releases it to surpass the honest miners' public chains to 'steal' the mining rewards. This paper makes the following contributions:

1. An empirical analysis illustrates that there is a statistically significant correlation between the profitability of mining and the total hash rate, confirming that miners indeed respond to changing profitability (i.e., hash supply is elastic).

2. A theoretical analysis demonstrates that selfish mining under such elastic hash supply leads either to the collapse of a chain (i.e., all honest nodes will leave, despite the low chances in practice), or to a stable equilibrium depending on the attacker's initial share.

Blockchains vs. Distributed Databases: Dichotomy and Fusion

Blockchains and distributed databases share many similarities. This paper addressed an important research question about how blockchains compare against traditional distributed databases – by focusing on four dimensions:

1. **Replications** – Transaction-based replication models in blockchains have a negative impact with higher latency, while operation-based replication approaches have plain effects on distributed databases.

2. **Concurrency** – The number of operations per transaction is a key factor affecting concurrency in both blockchains and distributed databases when a large number of transactions pour in.

3. **Storage** – Compared to distributed databases, blockchain systems can introduce additional storage overhead as the full ledger (i.e., historical data) is maintained among all participating nodes.

4. **Sharding** – When increasing the number of shards, the performance of blockchains is inferior in terms of transactions per second (TPS) due to their underlying consensus mechanisms.

This work is seen as pioneering research for future blockchain-database design fusions.

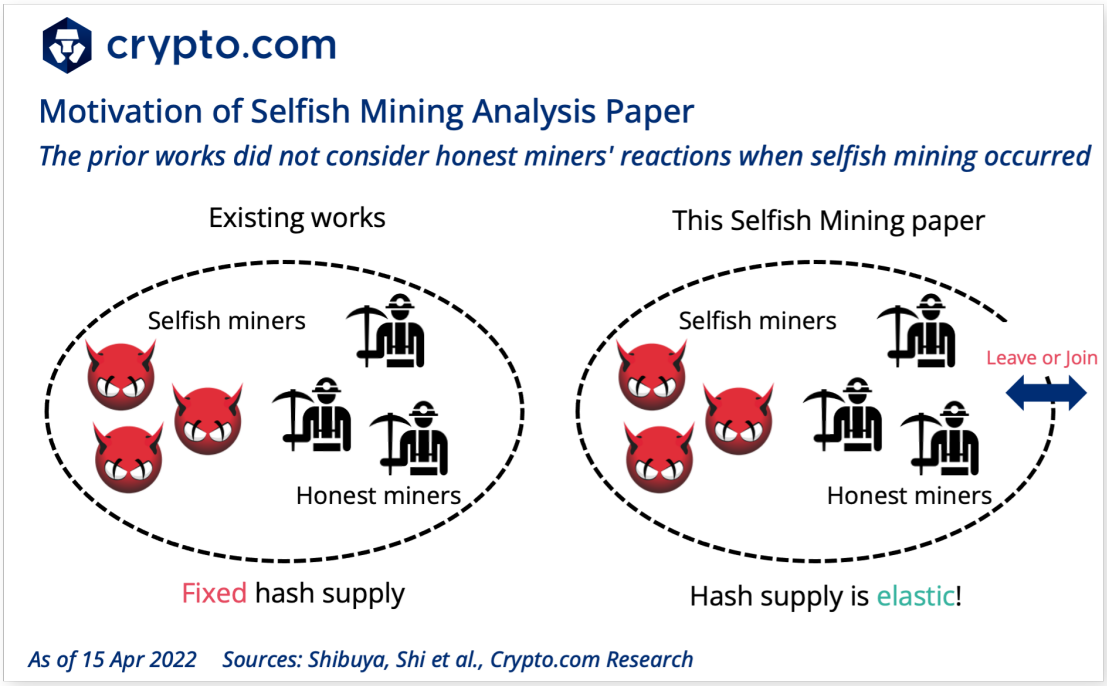
1. Analysis of Selfish Mining Attacks

We introduce the key points of the paper "Selfish Mining Attacks Exacerbated by Elastic Hash Supply" (referred to as "This Selfish Mining paper" in this report) by Yoko Shibuya, Elaine Shi, et al. This work has been published at the flagship International Conference on Financial Cryptography and Data Security (FC) 2021.

1.1 Introduction

In Proof-of-Work style blockchains such as Bitcoin, the existence of selfish mining attacks were first reported by Ittay Eyal and Emin Sirer in 2013.

Theoretically, selfish mining is a deceitful mining strategy where a sole miner (or a group) **first** mines out one or several blocks. **Instead of** releasing these blocks into the public chains to receive the corresponding mining rewards, these selfish miners **withhold** it privately, and **keep watching** the system. When any honest miners propose a valid block, the malicious miners will then **publish the withheld blocks immediately,** which incentivises other miners to append newer blocks into the selfish miner's longer chain so that they're eventually **forked into the main chain.** In this case, the selfish miners will make more profit.



However, the prior works did not consider honest miners' reactions to changes in profitability when the selfish mining attacks occurred. In particular, the fundamental assumptions of these works conclude that the total hash supply in a

chain is **fixed** and does not respond to changes in the profitability of the chain. In practice, the miners are rational and profit-oriented, and may leave or join the system based on their profitability. This implies that the total hash supply in a blockchain is **elastic.**

This Selfish Mining paper aims to move this limitation a step further by modelling and analysing the honest miners' reactions when selfish mining happens. The contributions are as follows:

1. By empirically studying the data from three blockchains (Bitcoin, Ethereum and Ethereum Classic), the authors found a significant correlation between total hash supply and per-hash mining revenue (i.e., the evidence of elastic hash supply with respect to miners' revenue).

2. With elastic hash supply, this paper further analyses the long-term effects of selfish mining on the ecosystem in the equilibrium state.

1.2 Terminology & Assumptions

In this section, we first describe the mechanism of selfish mining strategy and the concept of elastic hash supply. Subsequently, we will briefly introduce the model for experiments considered in this paper.

1.2.1 Selfish Mining

Selfish mining is a strategic mining algorithm that allows a miner or a coalition (e.g., mining pool) to make profits by compromising honest miners' revenues. Theoretically, successful selfish mining consists of following processes, as shown in the diagram below:

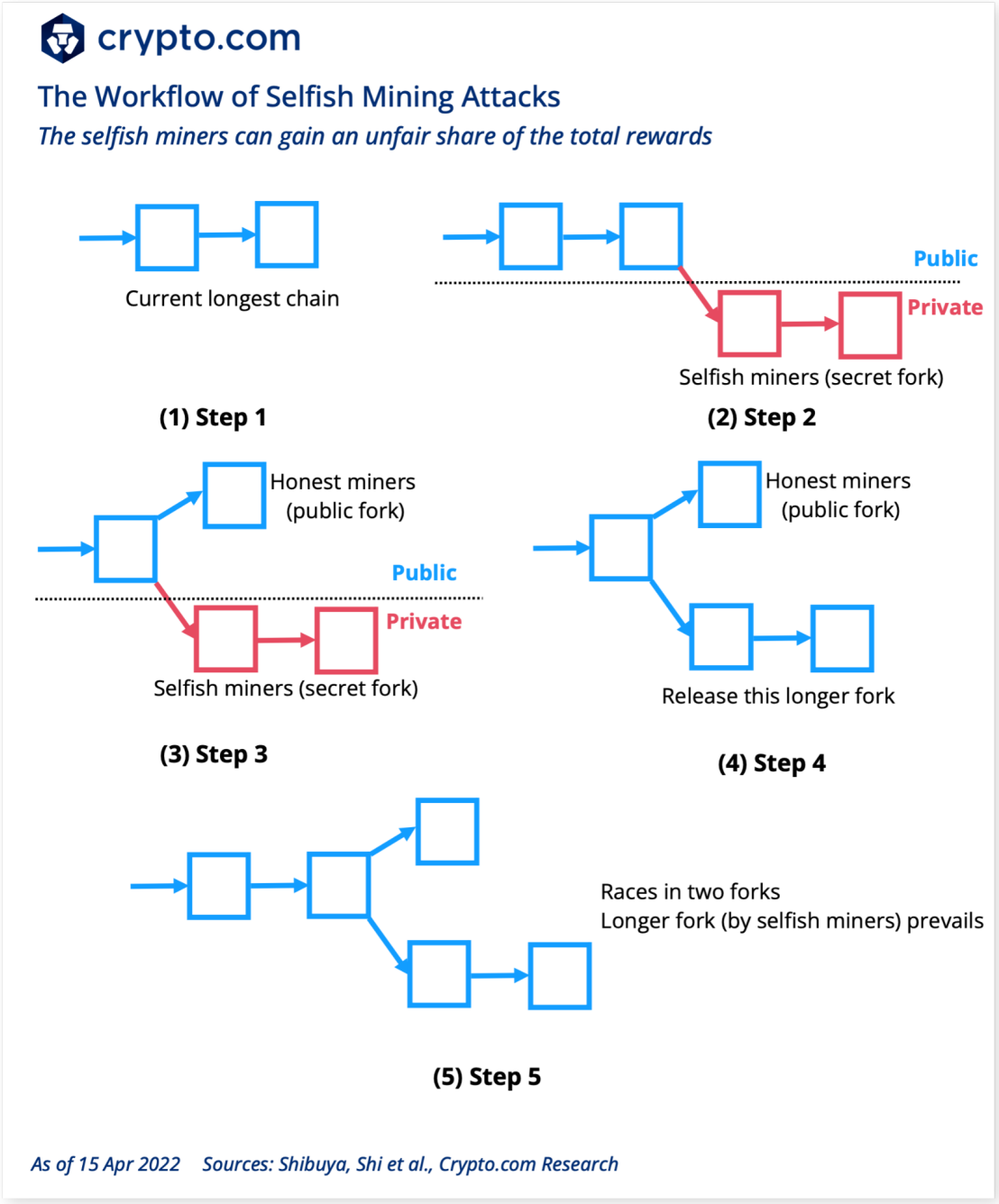
1. Let us first consider the current longest chain. The objective of selfish mining is to extend this longest chain by playing a suitable strategy.

2. When a malicious miner (or a coalition) mines several new blocks **(B*)** off the current longest chain, that miner keeps **B*** secretly rather than publishing it, forming a **secret fork.**

3. Whenever an honest miner mines out a block **(B)** to extend the current longest chain, the selfish miner releases the withheld **secret fork** immediately to become a **public longer fork.**

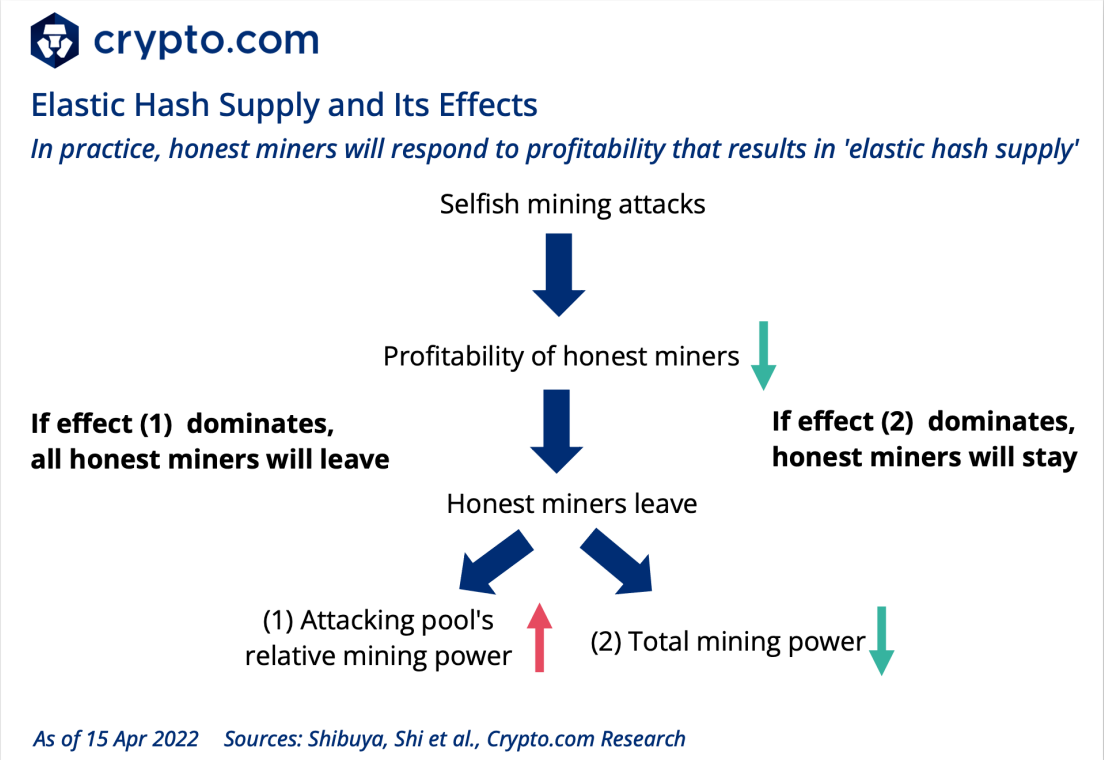
4. Since the fork released by the selfish miner is longer, it can convince other miners to consider it as the main chain. Therefore, every miner will follow the selfish miner's blocks.

5. In this case, the blocks generated by the honest miners are thus pruned, and their creators cannot receive any reward. In other words, through selfish mining, an adversary can erase some fraction of the honest mining power, and therefore the selfish coalition can gain unfair shares of the total rewards.



1.2.2 Elastic Hash Supply

The prior works only assume that the total hash power participating in mining is **fixed.** Such an assumption is unrealistic in real-world mining scenarios because, in practice, since each honest miner is economic-rational, they can freely enter and leave the system based on profitability.



The effects of elastic hash supply are as follows:

1. With a selfish mining attack, because a fraction of the honest mining power is being erased, the erased fraction is essentially not gaining rewards. For honest miners, the immediate effect is that the cost of mining to earn each unit of reward becomes **proportionally higher.**
2. If the honest miners' profitability plunges below zero, they start to leave the system.
3. As honest miners leave, the consequences are twofold.
    - The impact of the attack on the remaining miners is magnified, as a higher fraction of their mining power is now erased, which in turn drives more miners away (see **Effect (1)** in the diagram above).

- Meanwhile, as honest miners leave, the total mining power **decreases.** Hence, the mining difficulty drops, and mining becomes cheaper (see **Effect (2)** above). This somewhat counteracts the decreased profitability for honest miners that stems from being the victims of selfish mining.

By extending the previous **fixed hash supply** assumptions, this paper discovered and adopted the assumption that overall hash power is elastic in a chain. Following this assumption, the authors **statistically demonstrated** that elastic hash supply can result in two consequences:

1. If **Effect (1)** dominates, selfish mining drives costs up for honest miners. Eventually, all honest miners end up leaving the system.
2. Since **Effect (1)** and **Effect (2)** counteract each other, the system can reach a new equilibrium after some, but not all, honest miners have left.

By empirically measuring the data of Bitcoin, Ethereum, and Ethereum Classic, the authors concluded that in either scenario above, the unfairness of selfish mining is significantly **exacerbated** by the elasticity of hash power.

1.3 Evaluation Results

In this section, we will introduce the evaluation results of this paper. The authors studied the elasticity of hash supply with respect to miners' revenue using data from three different blockchains (Bitcoin, Ethereum, and Ethereum Classic) between 2015 and 2020.

Based on their statistical analysis, **this paper dispelled the mist of two hypotheses;** namely, '*Do the honest miners react to the profitability changes?*' and '*Does elastic hash supply affect selfish mining?*'

1.3.1 Do Honest Miners React to Selfish Mining?

This paper used the historical daily hash supply data and miners' per-hash revenue data from Coinbase in the experiments. In terms of evaluation methodologies, this work employed regression analysis to measure the relationship between 1% daily changes in miners' revenues and the changes in total hash supply. **In layman's terms,** regression analysis is a set of statistical processes for estimating the relationship between multiple variables (in this report, these are changes in miners' revenues and hash supply).

crypto.com

Correlation Analysis Between Miners' Revenue and Hash Supply

The honest miners indeed respond to the profitability change (cause elastic hash supply)

Table 1. Correlation results for three currencies in sample period (1 Jan 2017 - 31 Jul 2020)

	Bitcoin			Ethereum			Ethereum Classic		
	Mtd1	Mtd2	Mtd3	Mtd1	Mtd2	Mtd3	Mtd1	Mtd2	Mtd3
Relationship coefficients (%)	17.5	**18.3**	18.1	2.8	3.3	7.9	4.1	4.8	**2.7**
# of Obs	1308	1296	1308	1308	1296	1308	1308	1296	1308

** Relationship coefficients indicate if miners' revenue changes 1%, how much will the total hash supply change*

As of 15 Apr 2022 Sources: Shibuya, Shi et al., Crypto.com Research

As shown above, the authors leveraged three different regression methods on measuring Bitcoin, Ethereum, and Ethereum Classic. The **relationship coefficient** indicates **if miners' revenues increased/decreased by 1% per day, how much would the total hash rate change?** From the table above, with any regression method, the relationship coefficients are positive and statistically significant. **In other words, the total hash supply is elastic with respect to the miners' per-hash revenue.** The range of this coefficient is from 2.7% to 18.3%, illustrating that 1% change in the miners' revenues causes 2.7% ~ 18.3% change in the overall hash supply.

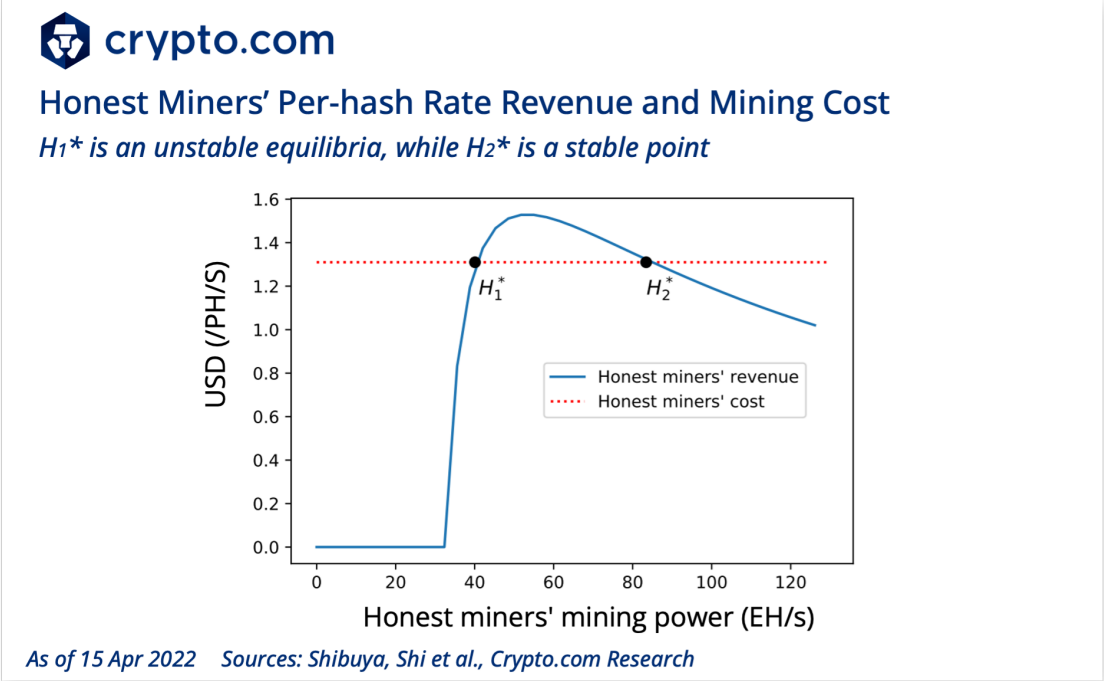
In summary, the honest miners indeed respond to the profitability changes, which results in the elasticity of total hash supply.

1.3.2 Does Elastic Hash Supply Affect Selfish Mining?

The regression analysis in the previous section illustrates that the overall hash rate keeps changing due to the prompt reactions from honest miners in the real world. In this section, we will describe the findings of how the elasticity of hash supply in a system can affect selfish mining.

The figure below describes the honest miners' per-hash revenue and mining cost. Note that PH/s refers to PetaHash per second, and EH/s denotes ExaHash per second. Under the free entry condition, the equilibria correspond to points $\mathbf{H_1^*}$ and $\mathbf{H_2^*}$, where the revenue curve intersects the mining cost (i.e., with zero profit). In this case, equilibrium $\mathbf{H_2^*}$ is stable, while $\mathbf{H_1^*}$ is not.

1. When honest miners' mining power **increases (decreases)** by any small amount from **H_1*, positive (negative)** profit will be generated and more honest miners will **enter (leave)** the system, ending up reaching equilibrium **H_2*** (or an equilibrium H = 0).
2. When mining power **increases (decreases)** from point **H_2*, negative (positive)** profit will be generated and honest miners **leave (enter)** the system. Therefore, **H_2*** is the only stable equilibrium.



In either scenario, the unfairness of selfish mining is significantly exacerbated by the elasticity of hash power.

1.4 Conclusion

The selfish mining literature assumed **fixed** total hash power. In contrast, this work showed that elastic hash supply **indeed existed,** and **significantly exacerbated** the impact of selfish mining.

Consequently, the statistical measurement first indicated that hash supply is elastic with respect to the miners' per-hash revenue (i.e., honest miners will react to in-platform profitability changes). Meanwhile, this paper discovered a threshold such that if the attacker's initial share of the total mining power was above the threshold, all the honest miners would leave and the chain would collapse. Like 51% attack, note that the chance of the collapse of a chain in practice is quite low.

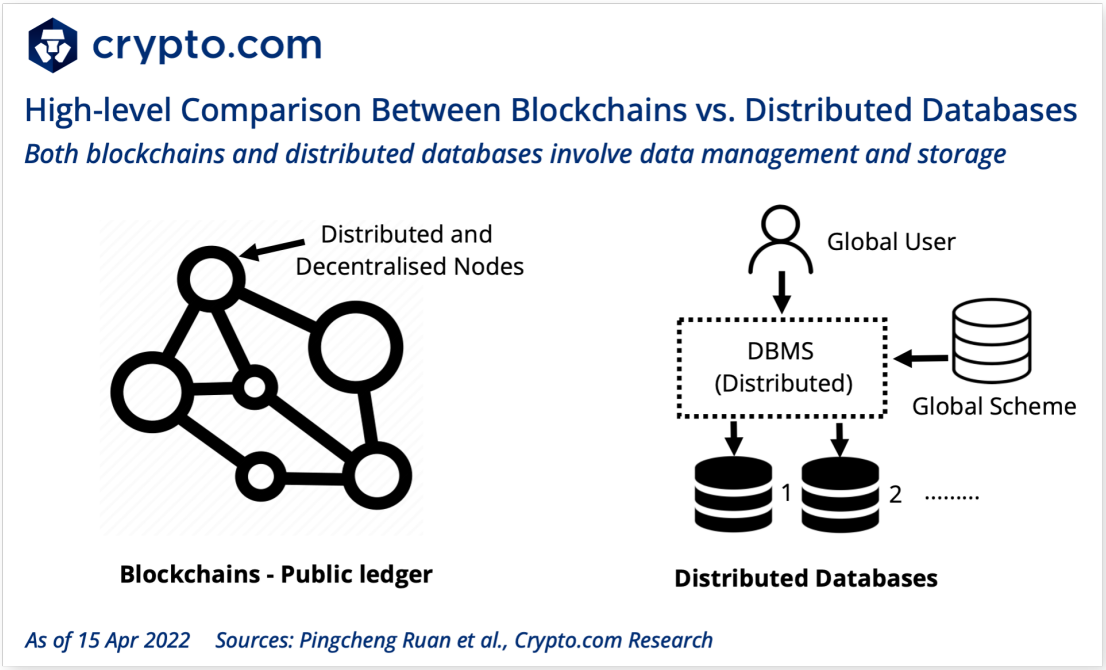
2. Dichotomy of Blockchains and Distributed Databases

We present an overview of the paper "Blockchains vs. Distributed Databases: Dichotomy and Fusion" by Pingcheng Ruan, Beng Chin Ooi, et al. This paper was a joint work by NUS, SUTD, ByteDance (Singapore), and ZJU, BIT (China). This work has been published at a flagship ACM annual conference SIGMOD in 2021.

2.1 Introduction

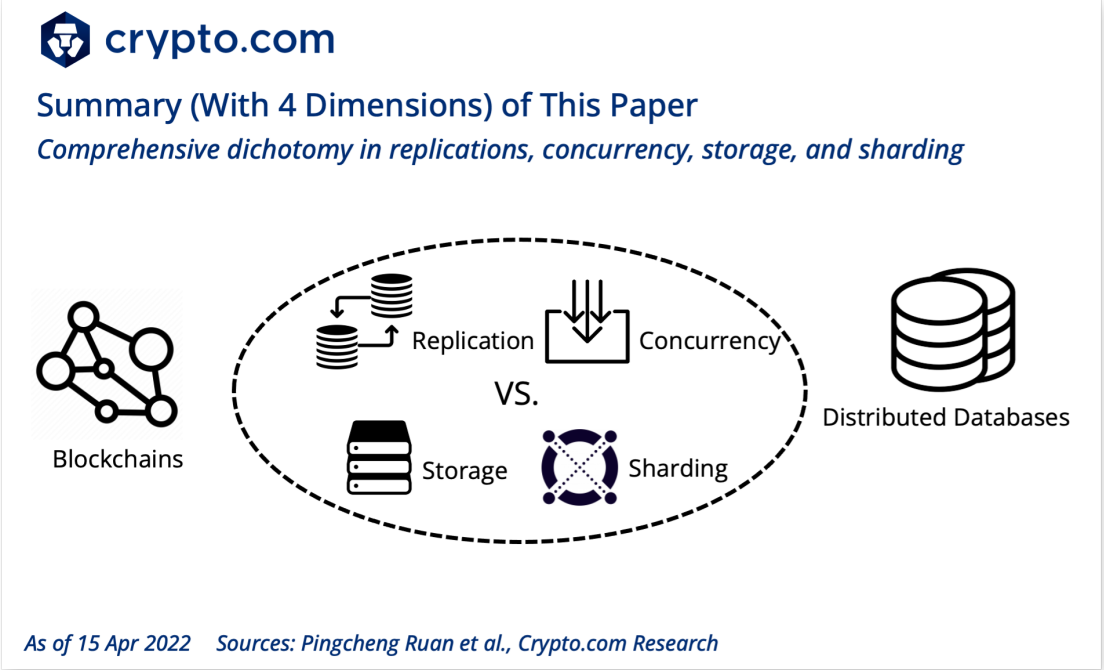
A blockchain is seen as an append-only public ledger that allows transactions (or any other types of data) to be securely stored by mutually untrusted participants over a consensus protocol. **From the perspective of data storage, blockchain innovations have evolved into a new data management system.**

Intuitively, an interesting research question is *'How do new-emerging blockchains compare against traditional distributed databases?'* The existing empirical studies only focus on high-level comparisons, such as security and throughput, which are limited and far from being enough.



This paper provides a comprehensive dichotomy on blockchain systems (i.e., **Quorum and Hyperledger Fabric)** and distributed databases (namely, **TiDB and etcd)** by four dimensions: **replications, concurrency, storage, and sharding.**

The authors describe how these dimensions have impacts on the performance. This work can be regarded as a potential direction for future blockchain-database design fusions.



Below, we will extend the introduction of the methodologies of this research, and its interesting findings.

2.2 Dichotomy Dimensions

As indicated in this paper, the previous works were merely trying to focus on the comparison of high-level features (e.g., security and throughput). How the low-level design choices result in the overall differences are vague and not yet investigated. To address this problem, the authors selected the following four metrics and proposed a twin study on blockchains and distributed databases.

Replications

Replication refers to the approach of storing copies (i.e., so-called replicas) of the data on multiple nodes. The key challenge in such a system is to ensure consistency under the worst cases, such as system failures.

This paper talks about three areas of replication metric:

1. **What to replicate – In practice, distributed databases replicate the ordered log of CRUD operations (i.e., create, retrieve, update, and delete)**

on top of the storage. In contrast, blockchains replicate the entire transaction (i.e., full ledger) so that its execution can be synchronised by each node.

2. **How to keep the replicas consistent – The paper introduced two primary approaches to maintain consistency among replicas, which are adopted by blockchains (sequentially transaction-based approach) and databases (operation-based approach), respectively.**
3. **The reaction under failures – No system is perfect without any failures. In general, most computer systems devise their fault tolerance solutions in which even a small fraction of nodes are compromised, but the services are still able to recover.**

Concurrency

Concurrency refers to the extent to which **transactions are executed at the same time.** There are two choices in most design spaces: transactions are executed either serially or concurrently. Most blockchains (e.g., Bitcoin) adopt serial execution as transaction execution is often not the bottleneck, while distributed databases employ sophisticated concurrency control mechanisms to extract as much concurrency as possible.

Storage

The storage designs in blockchains and distributed databases are different. Concretely, **the storage in distributed databases is saved in the form of write-ahead logs,** which are periodically pruned. In blockchain systems, **the ledger (i.e., chain of blocks) records historical transactions** and the changes made to the global state. This comparison paper discussed the actual performance with different storage record sizes.

Sharding

To improve the scalability limitations, sharding was originally proposed in distributed databases. The data storage is divided into numerous shards, each of which will store a fraction of the full data to improve the performance. Fortunately, sharding has been adopted on blockchain to harness concurrency across shards. This paper compared the impact of sharding on distributed databases and blockchain systems.

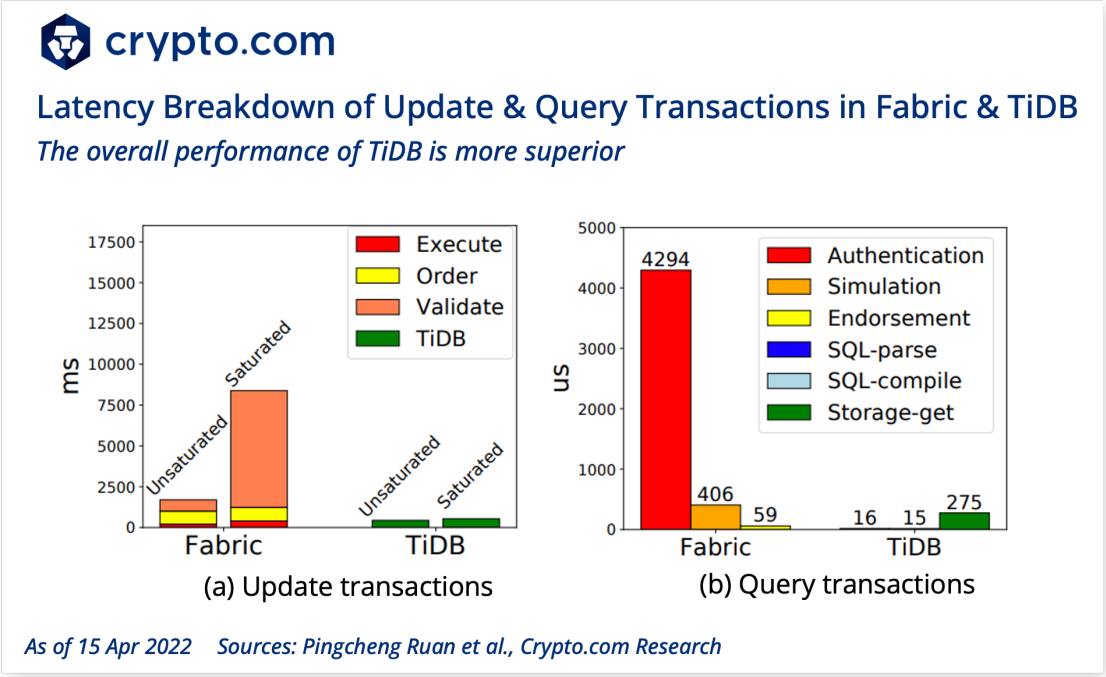
2.3 Results and Analysis

Following the four dimensions above, in this section we will introduce several interesting results. Note that the evaluation benchmarks are based on two blockchains (namely Quorum, Hyperledger Fabric), and two distributed databases (i.e., TiDB and etcd).

2.3.1 Effect of Replication

Subfigure (a) compares the replication latency of **update transactions** when the systems are both unsaturated and saturated in Hyperledger Fabric (Fabric going forward) and TiDB. The particular latency breakdown in Fabric includes the **execute, order, and validate** phases. In contrast, TiDB is based on an operation-based replication mechanism; thus, the operations are **simple but straightforward.**

As depicted below, apart from **higher latency,** Fabric incurred a significant spike in latency when the system was saturated (validation operation was especially seen as a bottleneck). In unsaturated situations, the execute, order, and validate steps take around 500ms, 700ms, and 700ms, respectively. However, distributed database TiDB **did not suffer from** such strict sequentiality under its operation-based replication, nor does it incur security overhead.



The security overhead is the most prominent in query transactions, which do not involve any consensus phase. As in Subfigure (b), authenticating the users becomes the dominance in Fabric, while TiDB does not introduce any complex cryptographic-relevant overhead.

Throughput with varying number of nodes

Protocol	3	7	11	15	19
Fabric	1566	1288	1031	749	528
TiDB	5726	8301	8898	6235	5465

** Throughput in transactions per second (TPS)*

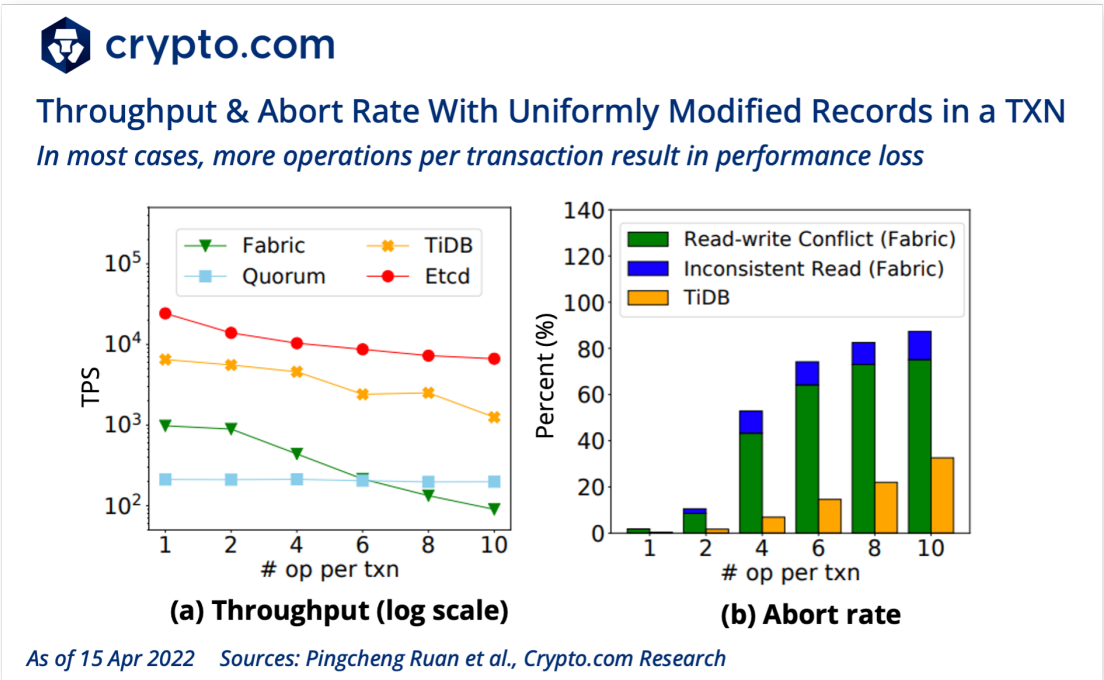
Another experiment was conducted to compare the TPS by increasing the number of participated nodes under full replication mode. The TPS of Fabric dropped around 3x from three (3) to 19 nodes because the **sequentiality in transaction-based replication** requires more signatures and longer validations. Instead, TiDB demonstrated much higher peak performance, but its peak was on 11 nodes. **The authors further concluded that the transaction-based replication model had an obvious impact on blockchains,** while replication approaches had **plain effects on distributed databases.**

The performance of fault tolerance was also presented in this paper. We suggest interested readers refer to the full paper for more details.

2.3.2 Effect of Concurrency

Operation count per transaction is a key factor affecting concurrency when a large number of transactions pour in. This paper performed an experiment by **increasing the update operations of each transaction** to analyse the impact of transaction atomicity on performance.

As shown below in Subfigure (a), the TPS of Fabric, TiDB, and etcd decreased significantly when the operation counts of every single transaction increased from one (1) to 10. **The reasons are twofold: On the one hand,** there are more conflicts when a transaction writes to more records, which leads to a higher abort rate. **On the other hand,** sharding, such as TiDB, is used in this platform. Therefore, more operations of a transaction may span multiple shards. However, **Quorum was unaffected** because it does not entail cross-shard transactions.



Moreover, Subfigure (b) shows the abort rate of TiDB and Fabric by increasing the operation count. In particular, **TiDB and Fabric suffered 26.9% and 87% abort rates, respectively.** The main reasons come from write-write conflicts, inconsistent reads, and the read-write conflicts in concurrency circumstances.

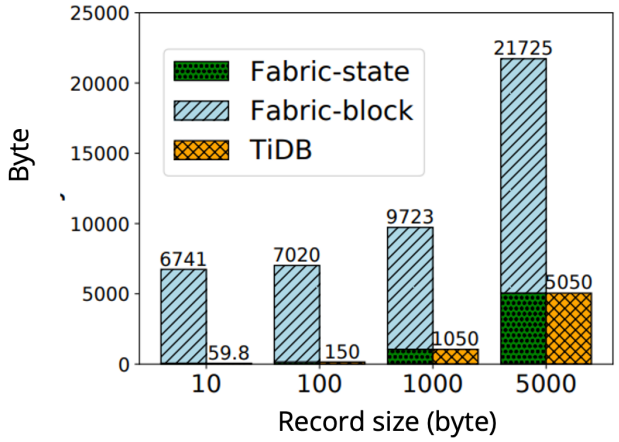
2.3.3 Effect of Storage

The figure below indicates the storage cost for each record by measuring the performance in different record sizes. Since blockchains keep all historical ledger data among all participating nodes, **it's obvious that Fabric brings more intensive storage overhead than TiDB.** In comparison, TiDB has no additional storage overhead since no historical data is maintained. As a result, we could conclude that, **compared to distributed databases, blockchains introduce more significant storage consumptions.**

crypto.com

Storage Breakdown in Fabric and TiDB

Blockchains incur significantly higher storage cost than datatbases



As of 15 Apr 2022 *Sources: Pingcheng Ruan et al., Crypto.com Research*

2.3.4 Effect of Sharding

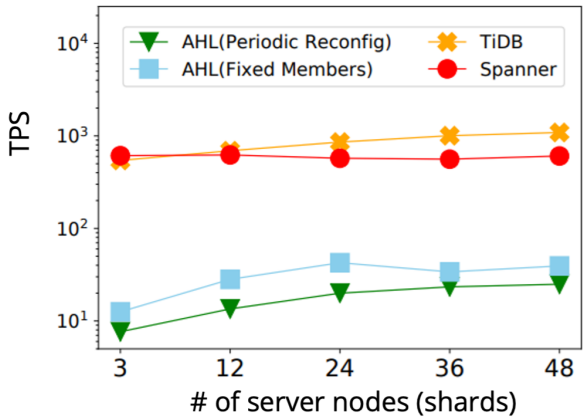
A comparison of the impact of sharding on blockchains and distributed databases is further presented in this paper. In the experiment setup, the authors chose two databases (TiDB and Spanner), and a new version of the Fabric family (i.e., AHL). One of the highlights in AHL protocol is that it periodically reconfigures shards to mitigate adaptive adversaries. **The number of shards in the experiment varies from three (3) to 48 in total.**

As shown in the figure below, **the throughput of both databases is relatively larger than blockchains in performance with the increased number of overall shards.** The reason is straightforward: blockchains have the inherent consensus protocols (e.g., PBFT in Fabric), which incurs higher overheads.

crypto.com

Throughput (varying shards) of TiDB, Spanner, and AHL (a Fabric type)

The gap in performance between Fabric AHL and both databases is relatively large



As of 15 Apr 2022 *Sources: Pingcheng Ruan et al., Crypto.com Research*

2.4 Conclusion

This paper presented a comprehensive dichotomy between blockchains and distributed databases from four design metrics, including replication, concurrency, storage, and sharding. By evaluating the performance based on these dimensions, the results illustrate the effects of different design choices to the overall performance.

This pioneer research work will foster the development and exploration of future blockchain-database fusions in the community.

References

Ruan, Pingcheng, et al. “Blockchains vs. Distributed Databases: Dichotomy and Fusion.” Proceedings of the 2021 International Conference on Management of Data, vol. 9781450383431, no. ACM, 2021, pp. 1504–1517. ACM.

Shibuya, Yoko, et al. “Selfish Mining Attacks Exacerbated by Elastic Hash Supply.” Proceedings of International Conference on Financial Cryptography and Data Security, vol. 12675, no. LNCS, 2021, pp. 269-276. Lecture Notes in Computer Science book series.

e. contact@crypto.com

Published on 18 Apr 2022