Bitcoin

Howard Wu

Bitcoin is a decentralized cryptocurrency that removes the control of transactions from a central authority. The protocol uses a peer-to-peer network and a hash-based proof-of-work to solve the double-spending problem. The Bitcoin blockchain makes publicly available the entire history of transactions and is replicated across all nodes in the network. For the many benefits this architecture achieves, Bitcoin comes with fundamental costs to privacy and scalability. We discuss the limitations of Bitcoin, an alternative architecture for scaling computations in blockchains, and secure multi-party computations on Bitcoin.

1 Privacy

The Bitcoin whitepaper stated that privacy was preserved by keeping public keys anonymous. However, recent work in de-anonymization, transaction linkability, and coin fungibility demonstrates that merely keeping public keys anonymous is not sufficient or practical.

1.0.1 Mixing Services

The goal of a mixing service is to shuffle one's coins with other participants' coins. Similar to traditional financial systems, the objective of mixing is to obfuscate the transactional history of the given assets so that they are untraceable back to the fund source. There exists protocols, such as CoinJoin and TumbleBit, for laundering blockchain tokens. However, in practice, services that operate these protocols impose fees, require waiting periods, have the ability to steal users' funds, and could be operating as a honeypot.

1.0.2 zkSNARKs

Zero-knowledge proofs allow one party, the prover, to convince another party, the verifier, that a given statement is true, without revealing any information beyond the validity of the statement itself. A zkSNARK is a variant of a zero-knowledge proof that enables a prover to succinctly convince any verifier of the validity of a given statement and achieves computational zero-knowledge without requiring interaction between the prover and any verifier.

This proof system was used to construct Zerocash, a protocol that provides a privacy-preserving version of Bitcoin. Instead of maintaining a ledger of all user transactions in public view, Zerocash uses zkSNARKs to construct proofs that attest to the payment details in transaction records. This way, the ledger stores no public information about origin, destination, or amount of the payment. This Zerocash protocol was developed into a decentralized cryptocurrency, Zcash.

2 Scalability

As the Bitcoin blockchain is replicated across all nodes and maintains a complete history of past transactions, Bitcoin faces scalability issues as the number of participants and transactions grows.

2.1 Proof of Work

Proof of Work (PoW) is a protocol used to deter denial-of-service attacks on a network by requiring a requester to demonstrate work done as stated by a provider. A proof of work is a piece of data that is difficult to produce, but easy to verify. In Bitcoin, the participant who submits the proof of work is able to publish the next block in the blockchain and receives a reward for doing this work.

2.1.1 Useless Computation

In order to ensure that every participant has a fair chance of generating a valid proof of work, Bitcoin uses a hashing-based computational puzzle to prove that computational resources were spent. As a consequence, large swaths of computational resources are dedicated to computing proofs of work in the hopes of receiving a financial reward. This useless computation does not provided any additional value outside of the network. As a result, critics remark that Bitcoin consumes more energy resources than many countries do.

2.1.2 ASICs

The mining difficulty is set according to the amount of hashing power in the Bitcoin network. As the number of participants mining for blocks increases, the difficulty for Bitcoin mining increases as well. At the time of writing, it is considered computationally and economically impractical to use CPUs or GPUs to mine Bitcoin. Rather, mining is done on dedicated hardware called ASICs (Application Specific Integrated Circuits). However, the necessity for ASICs inherently counteracts the goal for a decentralized network.

2.1.3 Mining Pools

As a consequence of increased mining difficulty, mining pools have formed to increase a group's mining power and improve their probability of mining a block. In the past, mining pools have approached 51% of the network's hashing power and operators diverted compute resources out of good will. However, at the time of writing, if the top three mining pools were to collude, they would be capable of surpassing 50% of the total mining power. This would enable a 51% attack, which could prevent the confirmation of new transactions and worse, reverse completed transactions and double spend coins.

2.1.4 Proof of Stake

Proof of Stake (PoS) proposes a more energy-efficient and computationally-efficient consensus protocol, where participants are fairly selected at random to mine the next block and receive a reward for it. A proof of stake scheme requires participants who wish to become a validator to bond an amount of tokens used to incentivize correct behavior. In comparison to the proof of work scheme in Bitcoin, this protocol is substantially more energy-efficient as participants do not need to run useless computations on dedicated hardware.

2.2 Transaction Throughput

The Bitcoin network averages between 2-5 transactions per second. For comparison, Paypal is able to handle around 500 transactions per second and VISA is capable of processing around 2000 transactions per second. As the number of Bitcoin users grows, the number of transactions is expected to grow as well.

2.2.1 Blockchain Size

At the time of writing, the size of the Bitcoin blockchain is on the order of 100GB and growing. As decentralization is a core requirement of such systems, Bitcoin faces the risk of excluding participants from running full nodes as the size of the blockchain grows.

2.2.2 Block Size

At the time of writing, Bitcoin blocks of 1MB size are produced on average once every 10 minutes. As Bitcoin adoption grows, blocks have become unable to handle the number of unconfirmed transactions in the network. As a short term solution, SegWit (Segregated Witness) was adopted to remove the signature data from inputs within Bitcoin transactions, moving it to a structure at the end of transactions. This technique frees up a significant amount of the block space to accommodate more transactions per block.

3 Practical Limitations

In addition to privacy and scalability issues, Bitcoin as a decentralized cryptocurrency has notable limitations as a day-to-day medium for transactions.

3.1 Confirmation Times

To ensure a transaction succeeds with reasonable probability, Bitcoin transactors must wait 3-6 block confirmations. With a new block added on average once every 10 minutes, Bitcoin transactors are forced to wait on average 30-60 minutes to receive probabilistic assurance that their transaction succeeded. Such long wait times are not practical for normal use cases such as point-of-sale transactions.

3.2 Transaction Reversals

While a traditional financial institution is able to reverse an incorrect or malicious transaction, one cannot receive this assurance in Bitcoin. The immutable nature of Bitcoin transactions is both its strength and its weakness. After waiting sufficient block confirmations, one can be confident that their transaction is permanently recorded on the blockchain. However, should a user make a mistake in payment details prior to publishing a transaction, that mistake is also irreversible. Worse, if a user's Bitcoin private key were to be stolen, the funds associated with that wallet would be stolen as well.

4 TrueBit

Abstract. Bitcoin and Ethereum, whose miners arguably collectively comprise the most powerful computational resource in the history of mankind, offer no more power for processing and verifying transactions than a typical smart phone. The system described herein bypasses this bottleneck and brings scalable computation to Ethereum. Our new system consists of a financial incentive layer atop a dispute resolution layer where the latter takes form of a versatile verification game. In addition to secure outsourced computation, immediate applications include decentralized mining pools whose operator is an Ethereum smart contract, a cryptocurrency with scalable transaction throughput, and a trustless means for transferring currency between disjoint cryptocurrency systems.

TrueBit aims to solve the scalability problem inherent in many blockchains by proposing an interactive verification protocol for scaling computation. Currently, all miners in blockchains such as Bitcoin and Ethereum must validate transactions in blocks. However, TrueBit proposes an architecture where only a small number of participants are required to validate and submit solutions to the network. This significantly reduces the number of redundant computations performed in traditional smart contract systems such as Ethereum. This architecture would allow for applications such as a scalable decentralized exchange, truly decentralized mining, a scalable blockchain, outsourced computations, and scalable peer-to-peer storage.

The fundamental problem TrueBit must solve in its architecture is to ensure the correctness of submitted results. The observation is that placing substantial verification computations on miners results in the Verifier's Dilemma, where verifiers who must perform non-trivial verification tasks may choose to skip the tasks in order to start mining the next block. Naturally the advantage probability increases by spending more time mining the next block rather than validating the current block.

TrueBit uses an interactive verification game involving Solvers, Challengers, and Judges to mitigate the Verifier's Dilemma. In the verification game, the Judge can validate a Challengers claim against the Solvers submission by analyzing the computation steps up to the point of disagreement and run one additional computation step to determine the correct submission.

Any user offering a reward for providing a valid solution to a task is called a Task Giver. The Task Giver offers a task to the Solver, who is chosen randomly from the set of participants in the blockchain. Verifiers are tasked with ensuring the correctness of submitted solutions, and call for a challenge when an error is detected. Verifiers are incentivized with a reward for correctly identifying bugs in a Solver's solution.

TrueBit incorporates an incentive layer that provides a substantial jackpot to Verifiers who correctly report forced errors. Forced errors are imposed in the system as a mechanism to prevent the Verifier's Dilemma. In the system, the Solver is given a random bit that denotes whether to provide an erroneous solution and is not penalized for providing an erroneous solution if forced error is in effect. Verifiers must correctly identify the erroneous solution when forced error is in effect to receive the jackpot payout. Incorporating forced errors allows TrueBit to give the system unpredictability and motivate verification of all tasks.

5 Secure Multiparty Computations on Bitcoin

Abstract. Bitcoin is a decentralized digital currency, introduced in 2008, that has recently gained noticeable popularity. Its main features are: (a) it lacks a central authority that controls the transactions, (b) the list of transactions is publicly available, and (c) its syntax allows more advanced transactions than simply transferring the money. The goal of this paper is to show how these properties of Bitcoin can be used in the area of secure multiparty computation protocols (MPCs).

Firstly, we show that the Bitcoin system provides an attractive way to construct a version of timed commitments, where the committer has to reveal his secret within a certain time frame, or to pay a fine. This, in turn, can be used to obtain fairness in some multiparty protocols. Secondly, we introduce a concept of multiparty protocols that work directly on Bitcoin. Recall that the standard definition of the MPCs guarantees only that the protocol emulates the trusted third party. Hence ensuring that the inputs are correct, and the outcome is respected is beyond the scope of the definition. Our observation is that the Bitcoin system can be used to go beyond the standard emulation-based definition, by constructing protocols that link their inputs and the outputs with the real Bitcoin transactions.

As an instantiation of this idea we construct protocols for secure multiparty lotteries using the Bitcoin currency, without relying on a trusted authority (one of these protocols uses the Bitcoin-based timed commitments mentioned above). Our protocols guarantee fairness for the honest parties no matter how the loser behaves. For example: if one party interrupts the protocol then her money is transferred to the honest participants. Our protocols are practical (to demonstrate it we performed their transactions in the actual Bitcoin system), and can be used in real life as a replacement for the online gambling sites. We think that this paradigm can have also other applications. We discuss some of them.

Secure multi-party computation enables a group of mutually distrusting parties to compute a joint function while keeping their own inputs private, e.g. a coin-tossing protocol. A fair and secure multi-party computation can be constructed as a decentralized protocol using timed commitments and the Bitcoin network.

In a standard commitment scheme, the committer has some secret that it wants to withhold from the receiver. In the commit stage, the committer uses a trapdoor function, such as a hash function, to compute a commitment for the receiver. Later on, the committer submits a value to the receiver to verify the original commitment. In the Bitcoin-based timed commitment scheme, the committer follows the standard commitment scheme and constructs a commit block with a deposit amount. The Bitcoin-based timed commitment scheme enforces that the committer pays the receiver a deposit if the commitment was invalid.

A fair two-party lottery protocol can be constructed with a timed commitment scheme. In this setting, two participants make their commitments, broadcast their input transactions, create a compute transaction, and then open the commitments. Time-locks are used to provide financial compensation to an honest party when the other party misbehaves. Using timed commitments and the Bitcoin blockchain enables the construction of a fair lottery protocol that ensures its correctness, security, and resiliency to malleability attacks.