A Review of 'Blockchains from a Distributed Computing Perspective'

Prashanth R Duggirala, Article by Maurice Herlihy

Summary

The article attempts to accessible and expository in explaining blockchain research from the perspective of distributed computing, intended mostly to appeal to blockchain novices.. Herhily states that the paper is "*colored by the perspective that much of the blockchain world is a disguised, sometimes distorted, mirror-image of the distributed computing world*". Many of the basic algorithms and techniques used in blockchains are best understood as variations on familiar algorithms and techniques from classic distributed computing. The paper introduces the concept of a ledger using the case of Alice's online news service which is implemented on a simple linked list and a primitive consensus algorithm using multi threads (called miners). The article says that this kind of system can be advantageous by being universal; it can be implemented using any type of data structure and all issues of concurrency and fault-tolerance are compartmentalized in the consensus protocol. This shared-memory universal construction is not yet a blockchain, because although it is concurrent, it is not distributed. From the log based centralized implementation, the article moves on to blockchain implementation where unlike the shared-memory universal construction, a linked list served as a ledger, and an atomic memory operation served as consensus, a list is kept in cloud storage serves as a ledger, and a combination of Byzantine fault-tolerant voting and human signatures serves as consensus. Through an example of a public blockchain, the article reiterates the immutable property of a blockchains and transactions facilitated by the use of public and private keys. Here, instead of having threads doing the mining, the blockchain management is crowdsourced by offering a fee to whoever volunteers to be a miner for the purpose of running a consensus protocol.

The article introduces the proof of work idea to tackle the threat of dishonest miners who can compromise the data in the ledger. PoW is a form of costly signaling: it is expensive in terms of time wasted and resources spent. As a famous example, Bitcoin uses PoW for consensus: only a miner which has successfully solved a computationally hard puzzle (finding the right nonce for the block header) can append to the blockchain. Although PoW was invented by Dwork and Naor [6] as a way to control spam, Nakamoto's application of PoW to large-scale consensus was a genuine innovation, one that launched the entire blockchain field. Private blockchains are better suited for business applications, particularly in regulated industries, like finance. Public blockchains are appealing for applications such as Bitcoin, which seek to ensure nobody can control who can participate, and participants may want to stay anonymous. Most blockchain systems also provide some form of scripting language to make it easier to add functionality to ledgers. Bitcoin provides a rudimentary stack-based language, while Ethereum provides a language similar to JavaScript. Such programs are called smart contracts. The varied functionality of Smart contracts is introduced like objects, Monitors and Read-Modify-Write operations. The smart contracts are not devoid of challenges either. The paper talks about the DAO attack, and makes a great point about the importance of concurrency control for smart contracts.

Strong points

1. The article attempts to accessible and expository in explaining blockchain research in terms of the many similarities, parallels, semi-reinventions, and lessons not learned from distributed computing.

2. The article gives several concise examples using a set of recurring characters, scenarios and code snippets to explain the blockchain concepts. Especially the explanation of the role of miners and the reward concept is excellent and easy to understand.

3. The article talks about a number of promising application areas beyond just coins by introducing smart contracts.

Weak points

1. The article assumes only some miners might be malicious, but there is a chance that a majority is malicious and the article does not talk about cases like that.

2. The article does not make any reference to or acknowledge the usage of hash trees in blockchain.

3. The paper talks about ethereum blockchain implements a Turing-complete imperative language for adding functionality to ledgers, this is not entirely true.

Comments expanding upon weak points

The article mentions that a consensus protocol involves a collection of parties, some of whom are honest, and follow the protocol, and some of whom are dishonest, and may depart from the protocol for any reason but most of the cases assume that a majority of parties are honest. The article does not talk about cases having a majority of users with malicious intent, for example there can be huge corporations or countries which own huge compute power who can wield unwanted influence over the blockchain.

Second, hash trees can be used to verify any kind of data stored, handled and transferred in and between computers. Currently the main use of hash trees is to make sure that data blocks received from other peers in a peer-to-peer network are received undamaged and unaltered, and even to check that the other peers do not lie and send fake blocks. Merkle trees are used in the Bitcoin blockchain but the article does not make any reference or acknowledge its usage in bitcoins blockchain.

Finally, Gas is the unit of measurement used to measure the cost of running an operation on the Ethereum blockchain. The concept of gas exists to separate computational cost of running an operation from the market value of Ethereum and we know that a true Turing machine works with unlimited resources and the concept of Gas prevents the Ethereum blockchain to be truly Turing complete.

References:

1. https://en.bitcoinwiki.org/wiki/Merkle_tree
2. https://muratbuffalo.blogspot.com/2018/02/blockchains-from-distributed-computing.html
3. https://news.ycombinator.com/item?id=16191506
4. https://www.ethos.io/understand-blockchains-from-a-distributed-computing-perspective
5. https://hackernoon.com/turing-completeness-and-the-ethereum-blockchain-c5a93b865c1a