Scaling blockchain cryptocurrencies with leveled zones

{ Borys Gurtovyi and Eduard Sanou }
University of Southern California

Introduction

Current decentralized cryptocurrencies based on **blockchains suffer from a scalability problem** that makes them unfit to replace centralized payment systems such as VISA and PayPal.

We propose an idea to achieve a higher scalability than current cryptocurrencies by creating a **hierarchy of transaction zones at different levels**, in which the transactions have different properties depending on the smallest level shared zone.
Each level would have a fork of a common blockchain that would be merged with sibling zones (those under the same parent) periodically, with the periodicity being different at each level. The periodicity would be lower on higher level zones.

Transactions within the lowest level zones would be applied and verified inmediately whereas those traversing zones would only be applied once the forks of the corresponding zones are merged.
We suggest using 4 levels: global (0), country (-1), state (-2), city (-3).

This design would allow the transaction throughput within a single city to achieve a similar throughput as current blockchains do globally.
In other words, we are **optimizing transaction delays by distance**, which is a common expectation of fiat transactions, while allowing the global transaction throughput to increase significantly.

Scaling with leveled zones

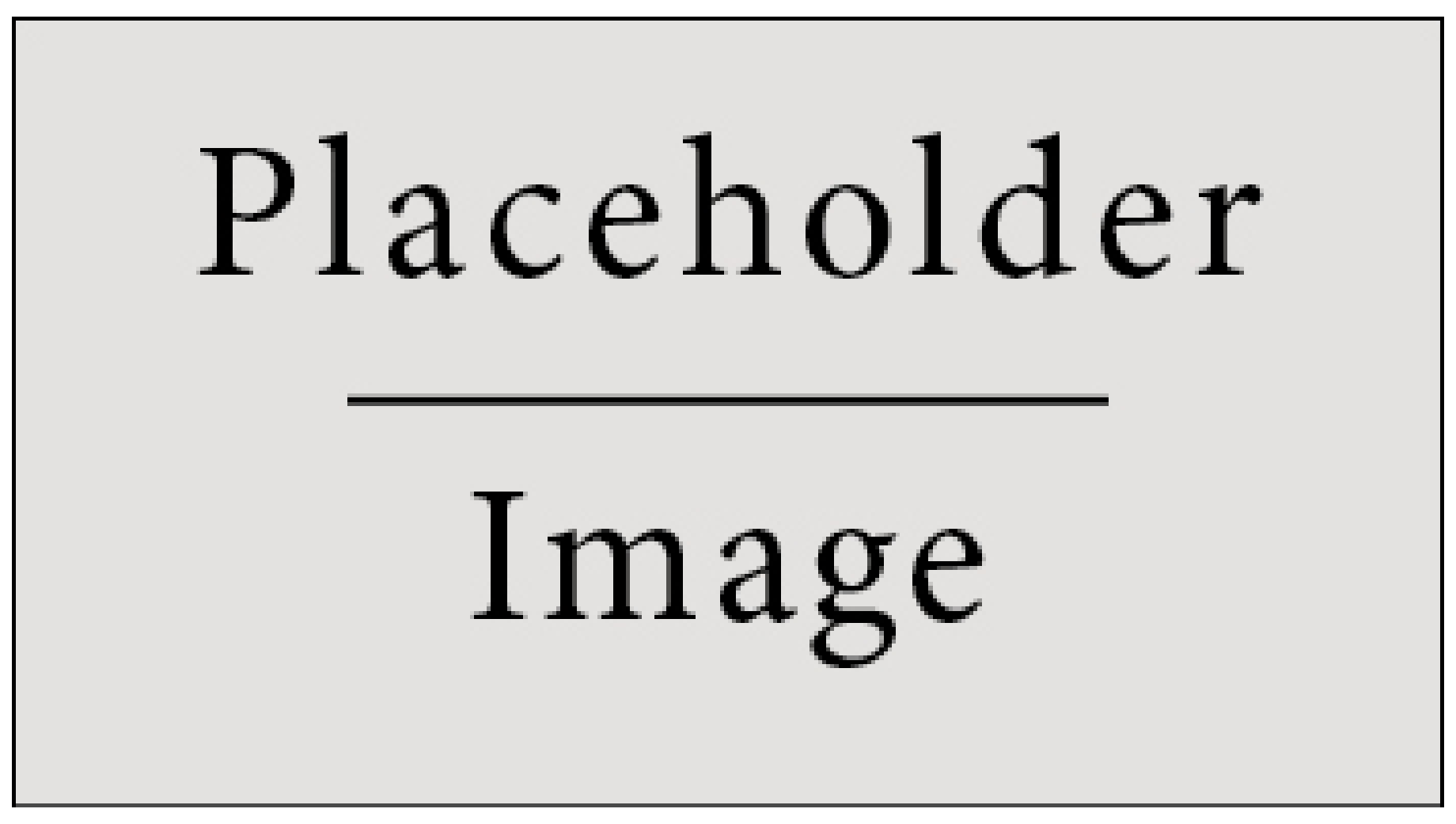

Figure 1: Figure caption

The global blockchain is periodically being forked into fine grained local blockchains and periodically being merged into coarser more-global blockchains.

1. User assigns a wallet public key to a particular city. This information is stored in the global blockchain. However, we don't restrict him from having multiple accounts in different cities. The locality is called "city" in our system if its population exceeds 1 million people. Other smaller localities will be combined into one fork.
2. A wallet can claim to be in a new city any time, but this information will propagate to the global (0) blockchain in the same way as transactions.
3. Every zone creates a fork from the blockchain fork of the upper level periodically.
4. Each zone only accepts transactions that originate from the same zone.
5. In-zone transactions are verified during the merge of the lower level zones (or as they are collected into blocks in the city (-3) level). [add graph here]
6. Out-zone transactions are verified when the lowest common upper level zone merges the zones it entails. [add graph here]
7. Transactions are verified when the forks they belong to are merged into higher zones (or as they are collected into blocks in the city (-3) level).

4 levels (from high to low) : global (0), country (-1), state (-2), city (-3)
Intervals:

- Global (0): 24h
- Country (-1): 2h
- State (-2): 10 minutes
- City (-3): 10 seconds

When user 1 (from city A) sends a cross-city transaction to user 2 (from city B) this transaction is being added to the city A fork as a special âĂIJholdâĂİ transaction. Basically, user 1 cannot double-spend this money, however user 2 will not receive them until city-merge will happen.
After merge Merges will not have conflicts, allowing them to be fast. Only verification against double spending is required.
If a user commutes between zones, they can have an account for each zone and split their money among the accounts.
No need to verify the actual location of the user VS the claimed location. The claimed location only serves to decide in which local fork the account can do transactions.

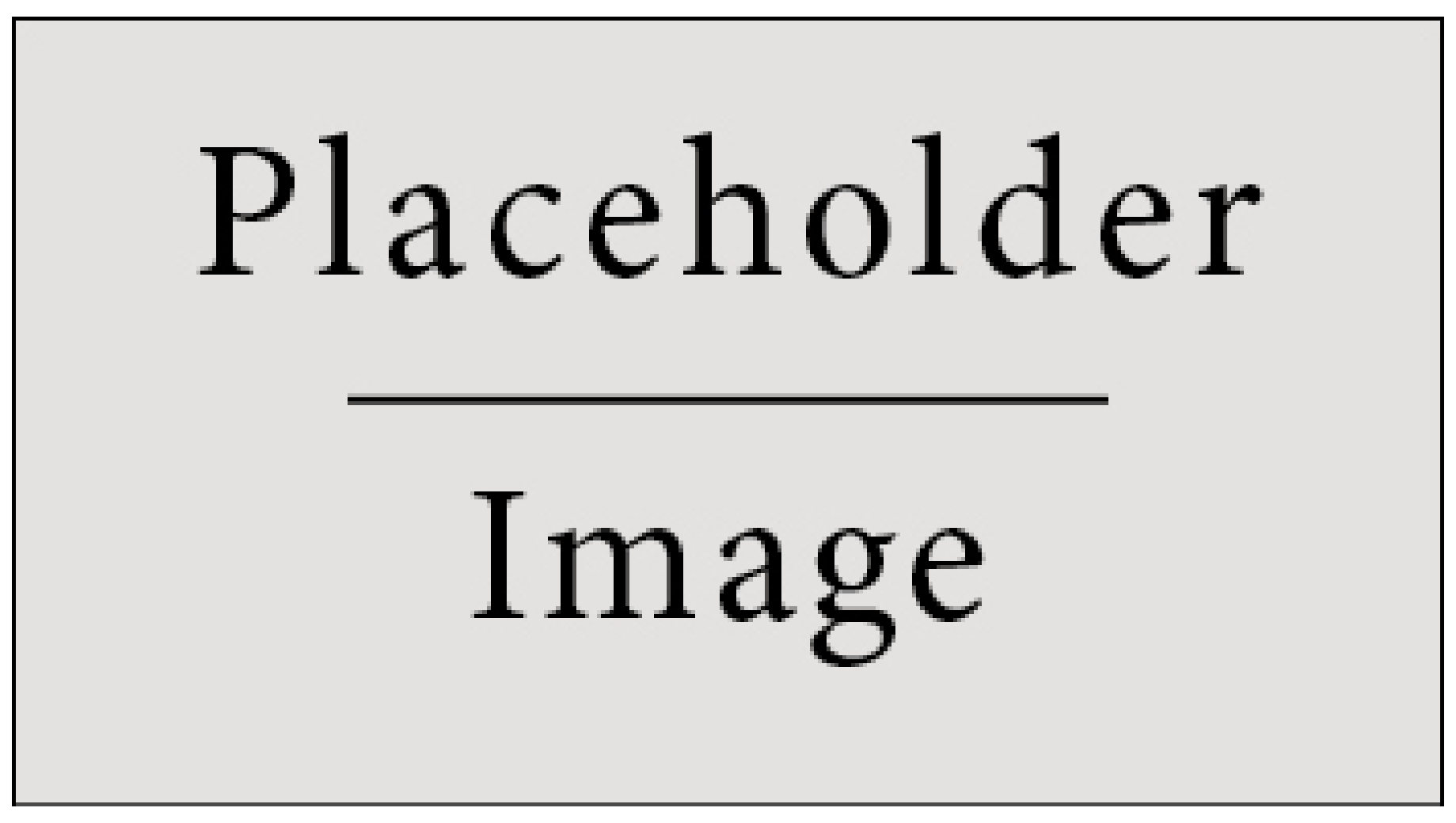

Figure 2: Figure caption

Background

Donec faucibus purus at tortor egestas eu fermentum dolor facilisis. Maecenas tempor dui eu neque fringilla rutrum. Mauris *lobortis* nisl accumsan.
Bitcoin, Ethereum, other cryptocurrencies not scalable. Scalable approaches are bla bla bla, and have reached this throughput bla bla. Widely used payment systems like Visa and Paypal have higher throughput bla bla.
Nulla ut porttitor enim. Suspendisse venenatis dui eget eros gravida tempor. Mauris feugiat elit et augue placerat ultrices. Morbi accumsan enim nec tortor consectetur non commodo.

Payment system	**transactions per second**
Ethereum	20
Bitcoin	3-4
Visa	24,000
Paypal	193

Table 1: Table caption

Challenges

- **The merging process requires more computational work and bandwidth on higher levels**. Not all nodes may be capable enough to verify merges at every level, Flexibility in the amount of contribution by node may be required.
- **When a node performs a merge, it needs to verify the forks of the sibling zones for correctness**. This could require a lot of computation resources. A possible optimization would be that each merge outputs a compressed form of the result that makes verification at the next level faster.
- **At the lowest level, each zone would have a small number of nodes**. This opens the possibility of an attacker disrupting the zone by registering nodes in it with high validation weight (either computational power or stake). Such disruption could be a denial of service or a denial of selected transactions.
  - A way to make make sure that all zones have the same computing power could be useful to solve this.
- **At each level, the merging time between forks should be faster than the periodicity of merges**.

References

[1] J. M. Smith and A. B. Jones. *Book Title*. Publisher, 7th edition, 2012.

[2] A. B. Jones and J. M. Smith. Article Title. *Journal title*, 13(52):123–456, March 2013.