Global CBDC Consensus

Hierarchical Stellar Consensus Protocol

Adrian Ross – Luke Hale – Yiannis Karamitros – Zeeshan Khan
Professor Hank Korth

I-DISC INSTITUTE FOR DATA, INTELLIGENT SYSTEMS, AND COMPUTATION

LEHIGH UNIVERSITY

Building Global Trust Without Central Authority

How can we get countries' central bank digital currencies (CBDCs), digital versions of a country's fiat currency, to settle across autonomous networks?

Numerous CBDCs are being proposed and prototyped by different nations. However, many rely on a high degree of central control. This research addresses a widely perceived need for decentralized control of a global currency that preserves the following traits:

- National/regional autonomy over currency and policies
- Varying degrees of trust across nodes and networks
- Fast and correct transactions between currencies

However, these goals may be difficult to achieve within a single, standard blockchain:

- Varying national choices between privacy and government surveillance
- Nonuniform trust model among nodes participating in a CBDC
- Performance bottlenecks in blockchain consensus

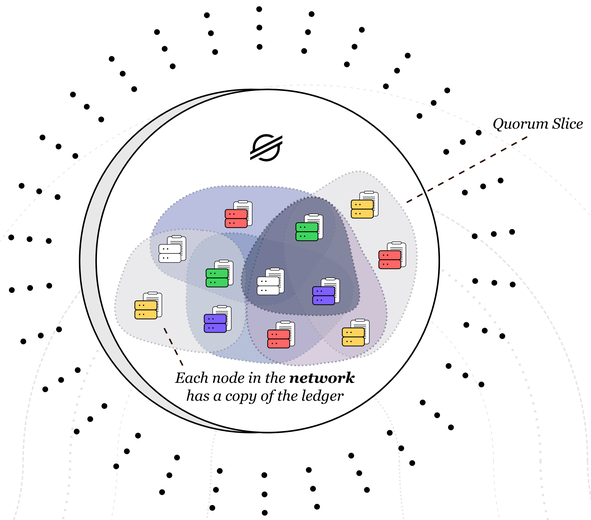
We present a system where individual sub-networks (subnets) have control over internal transactions policies while being able to prove correctability and achieve consensus at a global level.

Stellar: Circles of Trust Consensus

We base our implementation on the **Stellar Consensus Protocol (SCP)** because of SCP's ability to extend local trust to global trust. This model mirrors international financial relationships.

SCP is built on the federated Byzantine protocol, which reaches agreement based on the overlap of circles of trust, also known as quorum slices. A quorum slice is a group of nodes trusted by a particular node. Each node gets to define its own quorum slice.

While this paradigm works well, it has a scalability issue.



How Can We Scale?

The **Blockchain Trilemma** refers to the trade-off between 3 critical aspects of blockchain technology: security, scalability, and decentralization.

Common scalability solutions replace decentrally managed parallelism with centrally managed parallelism. Our scalability solution bridges this gap by relying on the fact that all nodes do not need to know about all transactions.

Soroban Smart Contracts

Smart contracts are code stored and run on a blockchain. This project uses the Stellar's smart contract platform, Soroban.

- Smart contracts can be autonomous
- Together, these contracts enable regional networks to operate in a decentralized manner, coordinating to reach consensus only when needed for global transactions

Our smart contracts construct the hierarchical consensus between different Stellar nodes, and their transparency allows the different nodes to trust the smart contracts.

Transaction Semantics

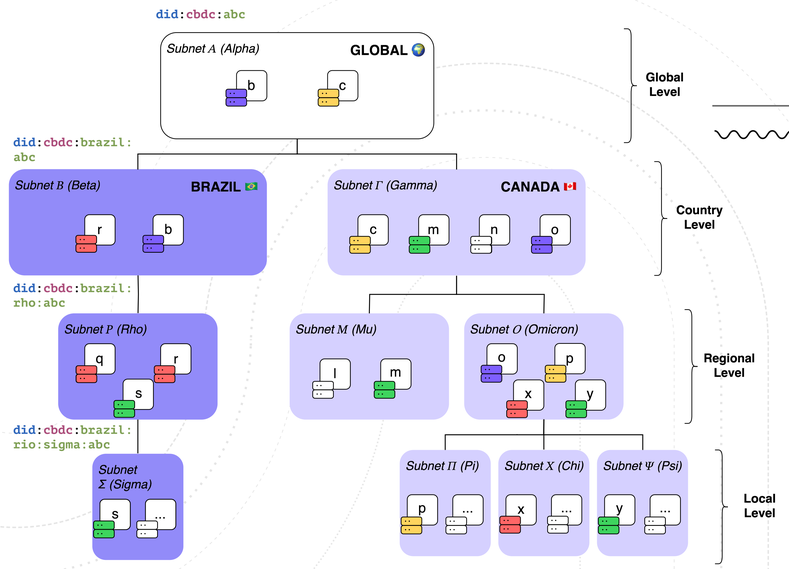
Beyond Consensus: Semantic Correctness for High-Performance Transactions

While consensus ensures all nodes agree on the order of transactions, it doesn't guarantee the overall outcome is valid. Traditional database transactions achieve correctness through serializability, but this often leads to high contention and impacts performance.

Our approach leverages **semantic correctness**, a model that prioritizes the meaning and outcome of transactions over a strict, serialized order.

- **Exploiting Commutativity:** Operations that don't conflict can be performed in any order without affecting the final state.
- **Failure Recovery via Compensation:** Instead of undoing transactions, specific actions can be "compensated for" if they fail.

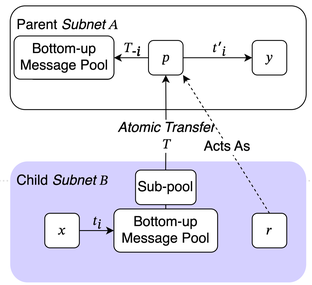
The foundation for semantic correctness lies in extended trust relationships within the network. Trust allows a node to be confident that its part of a multi-node transaction be compensated for if the full transaction later fails. This increases concurrency and reduces performance bottlenecks.



Implementation Features

By organizing networks in a hierarchical structure, we expand mutual trust to a global ecosystem and scope information exchange. Beyond SCP and Soroban smart contracts, we are building additional features to facilitate multi-network (multinet) consensus mechanisms and message passing. Some techniques that are key to our implementation include:

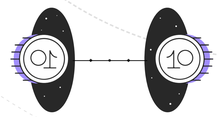
- Cross-network messages (Xmsgs): bridging communication between subnets
- Verifiable Random Functions (VRFs): providing trustless Xmsg proofs
- Decentralized Identifiers (DIDs): clear cut identity in multinet system, and ability to adopt others' relationships



W3C -DIDs

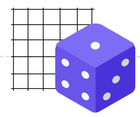
The entities in our network need a global, unique identifier. This is provided via the W3C Decentralized Identity standard (W3C DID). Each node can assign control of its identity through DIDs, facilitating cryptographic sortition and proxy in Xmsgs. DID format is based on node position within the hierarchy.

Cross-Network Msgs

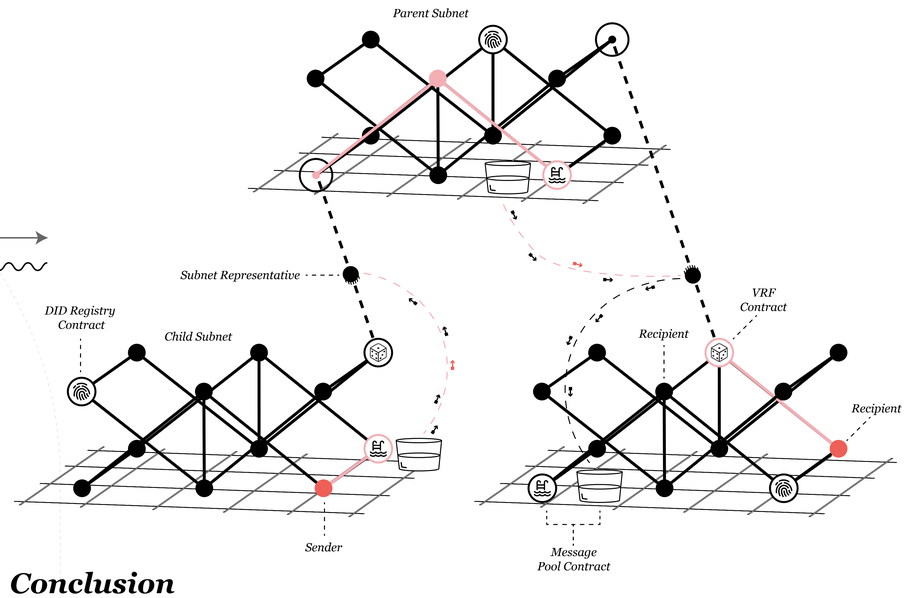


We establish a message-passing system, not only in a regional subnet but also cross-network following the hierarchical structure of our consensus network. Xmsgs include top-down (TD) or bottom-up (BU) depending on where they propogate.

Algorand VRFs



When consensus spans subnets, that higher-level consensus is reached among representatives of each subnet. To prove that representatives are properly chosen in a decentralized setting, we use Micali, et al.'s verifiable random functions (VRFs) based on Algorand's VRF and cryptographic sortition implementation.



Conclusion

- Our hierarchical consensus model provides a means to create a global CBDC with decentralized control. That decentralization of control enables guarantees of some degree of transactional privacy.
- By incorporating zero-knowledge technology to our framework (being studied in another Lehigh Blockchain project), we can design a CBDC with decentralized control along with transactions that are private, yet transparent, and provably compliant with regulations.
- While considerable work remains in developing a fully operational system, future efforts will focus on establishing cross-network connections and implementing network semantic correctness.