Efficient Smart Contracts on UTXO platforms

Smart contract efficienti su piattaforme UTXO

UNIVERSITÀ DI TRENTO

Roberto Zunino

joint work with Massimo Bartoletti,, Dario Maddaloni, Riccardo Marchesin

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
SERICS SECURITY AND RIGHTS IN THE CYBERSPACE

Overview: two main results

Improving smart contracts on UTXO platforms (e.g. Cardano)

- Off-chain UTXO smart contracts execution

  How To Save Fees in Bitcoin Smart Contracts: a Simple Optimistic Off-chain Protocol

  Blockchain: Research and Applications 2025

- Distributing the state for UTXO smart contracts

  Scalable UTXO Smart Contracts via Fine-Grained Distributed State

  Future Generation Computer Systems 2026

Off-chain execution of UTXO smart contracts

Summary: off-chain UTXO smart contracts

- Contract trees: a significant class of smart contracts
- State of the art:
  - On chain execution cost: 1+N transaction fees

    where N = number of smart contract operations
- Our contribution:
  - Optimistic execution protocol: *hope for the best, prepare for the worst*
  - Off-chain execution cost: (3+N-H) transaction fees

    where H = number of protocol steps where all participants behave honestly

    $(0 \leq H \leq N)$

Contract trees

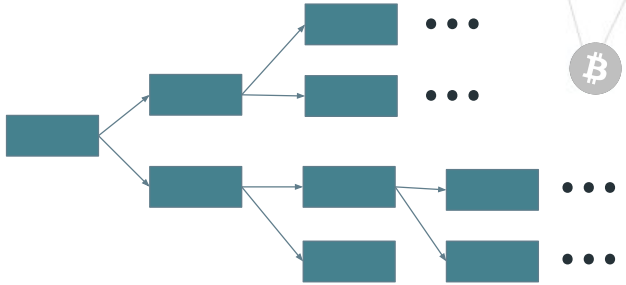
Assumption: all the possible executions of a smart contract can be described by a finite tree of state updates

Examples:

- tracing the production process of some goods
- gambling games
- ...

State updates can require conditions to become enabled:

- authorizations by participants
- time constraints
- ...



On-chain execution

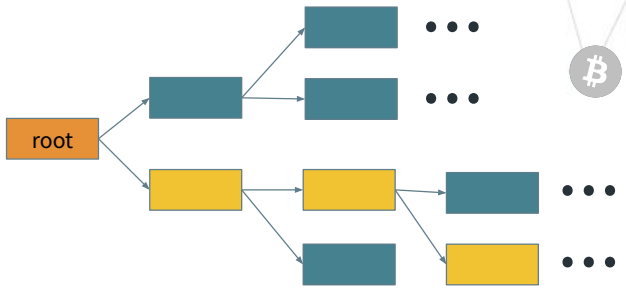
Contract stipulation:

exchange signatures, append root Tx on the blockchain

Contract execution:

choose one enabled Tx, append it, repeat

The chosen tree path forms a contract run



Assumption: at least one contract participant is honest

Off-chain execution

Floating transactions!

Our off-chain execution protocol exploits floating transactions (like the Lightning Network)

Most state updates are now executed off-chain, by exchanging messages (= no fees!)

Security:

- Same assumptions as for the on-chain case
- The contract tree logic is still enforced (no unexpected state updates)
- No deadlocks: in the presence of attacks, we can fall back to on-chain execution
- No state rollbacks: off-chain steps are final

Efficiency:

- Best case: 3 transaction fees (was 1+N in the on-chain case), can be lowered to 2
- Worst case: 3+N transaction fees

Distributing the smart contract state:

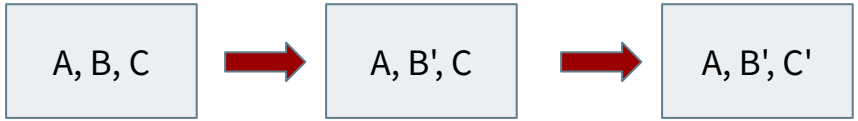
the hUTXO blockchain model

Summary: the hUTXO blockchain model

- State of the art:

  In the eUTXO model (Cardano) a transaction stores the whole contract state

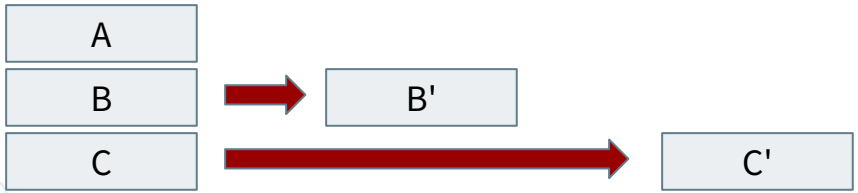
  Any contract operation must store the whole new state, again



- Our contribution:

  Our hUTXO model distribution of the state over several UTXOs

  - Contract operations only affect a part of the state, improving space and time efficiency



On distributing the contract state

A simple idea, yet requires some careful design to make it practical.

- Protecting from new attack vectors
- Storing the contract balance (we used a hybrid account/UTXO-based approach)
- Improving usability (addressed by our HURF smart contract language)

Advantages of hUTXO over eUTXO:

- Better time and space efficiency
- Enables parallel execution / validation on multi-code nodes
- Mitigates so-called "UTXO congestion" issues

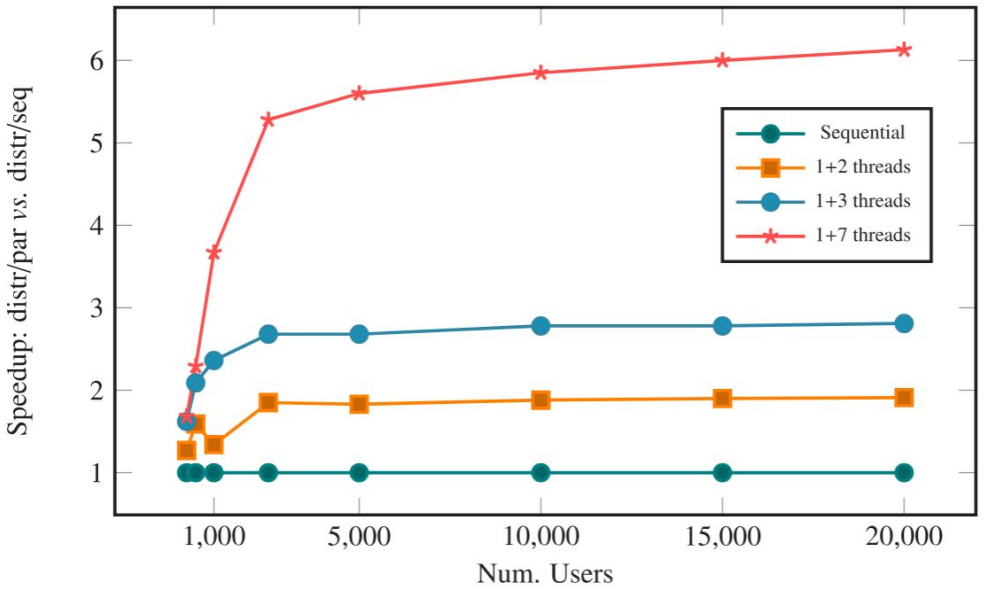
hUTXO implementation and experiments

We implemented a hUTXO simulator

(DiSCo_sim: ~7000 LoC of Rust)

Experiments show near-optimal scalability

on multi-core validator nodes



Thank you!

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
SERICS SECURITY AND RIGHTS IN THE CYBERSPACE