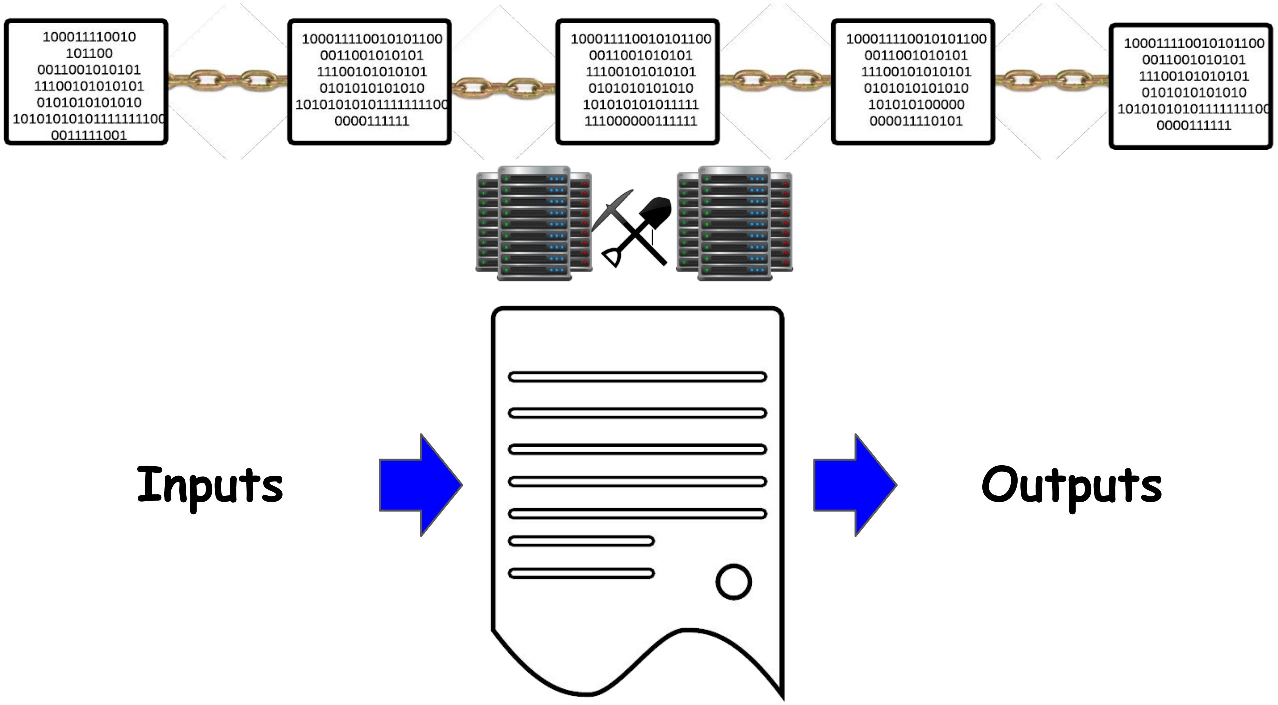
ammBoost: State Growth Control for AMMs

Nicolas Michel, Mohamed Najd, **Ghada Almashaqbeh**

University of Connecticut

TLDR 2024

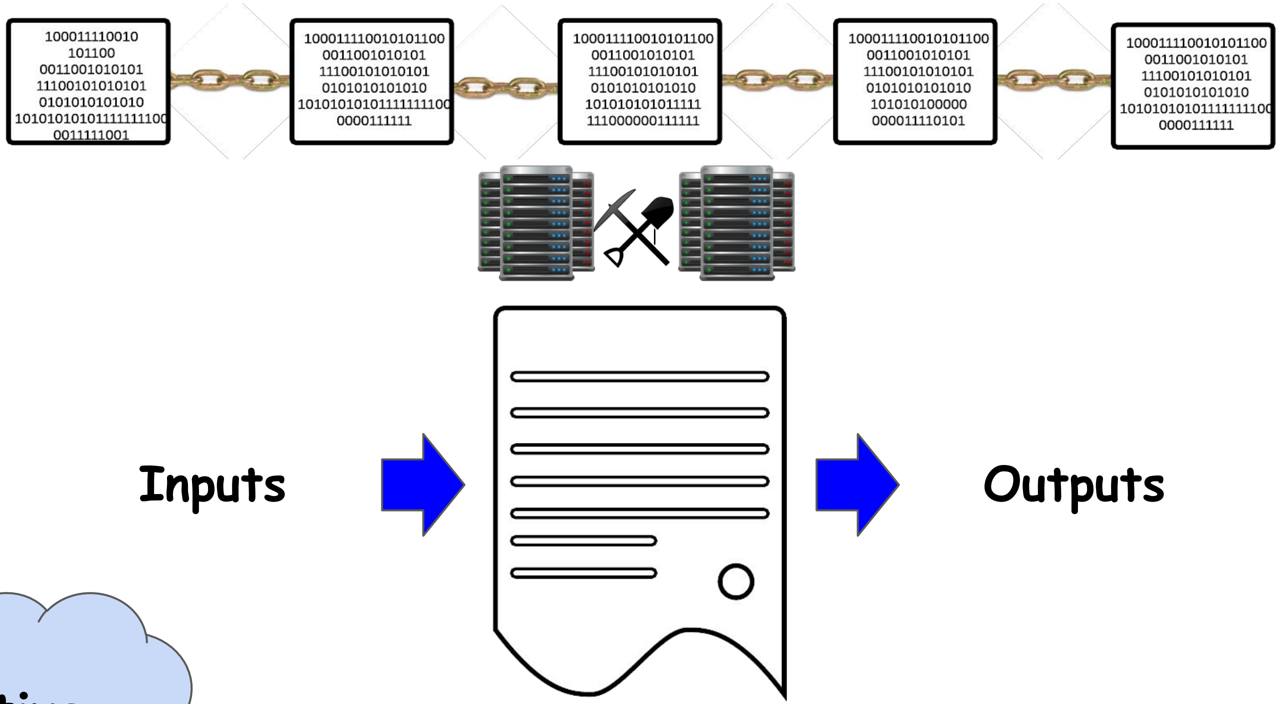
Smart Contract-enabled Blockchains



Inputs

Outputs

Smart Contract-enabled Blockchains

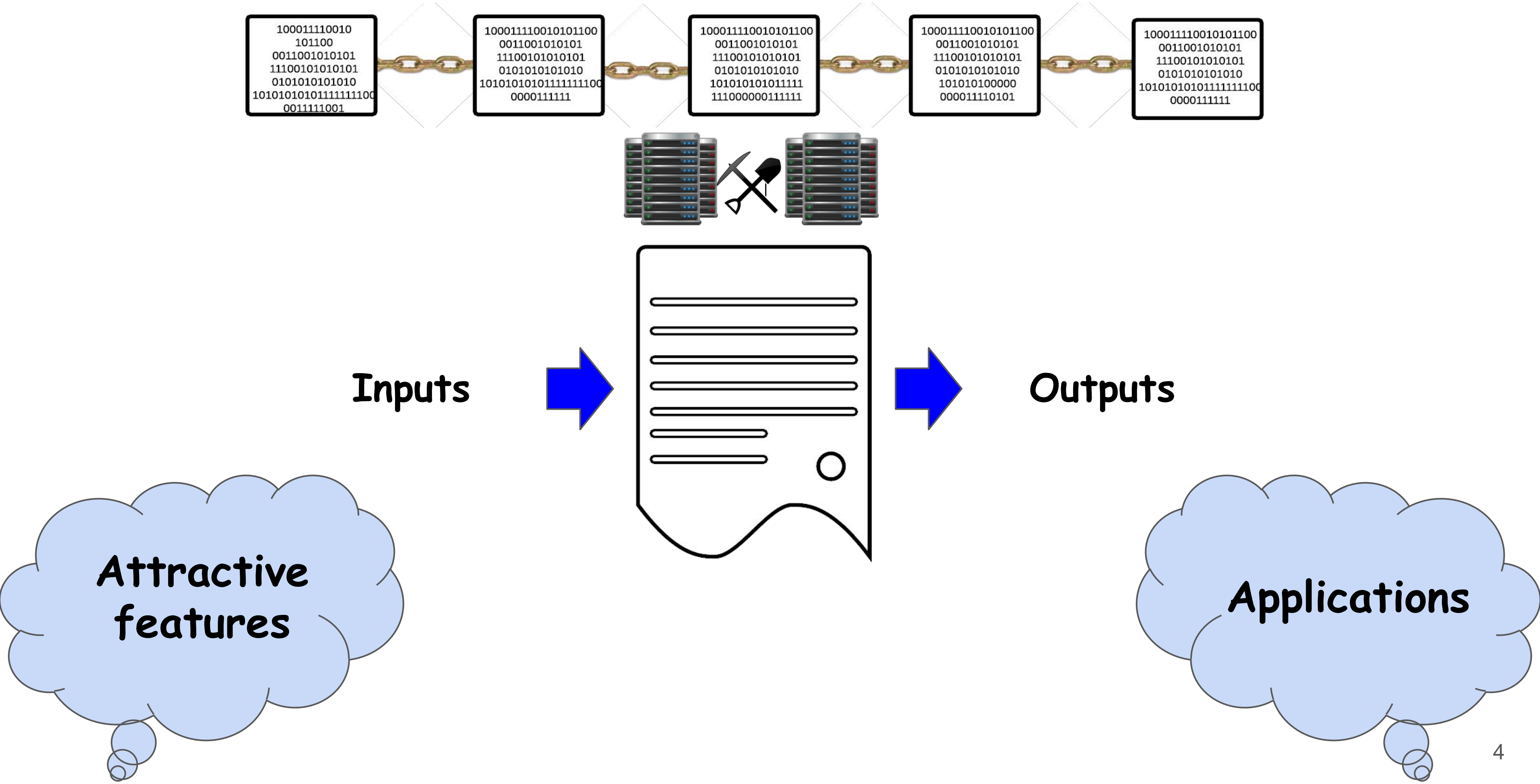


Inputs

Outputs

Attractive features

Smart Contract-enabled Blockchains



Inputs

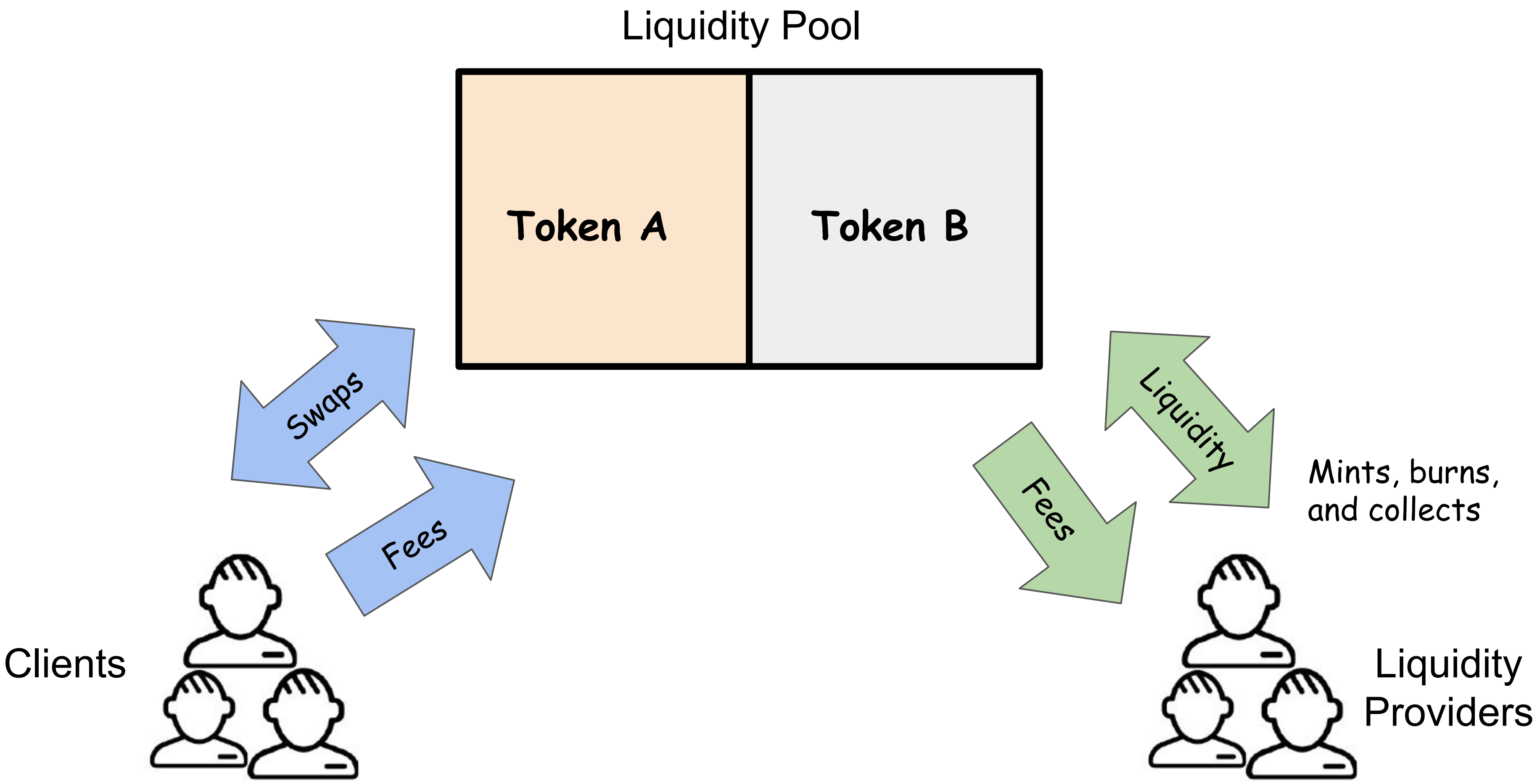
Outputs

Attractive features

Applications

Automated Market Makers (AMMs)

Our focus



AMMs are a Huge Industry

Curve

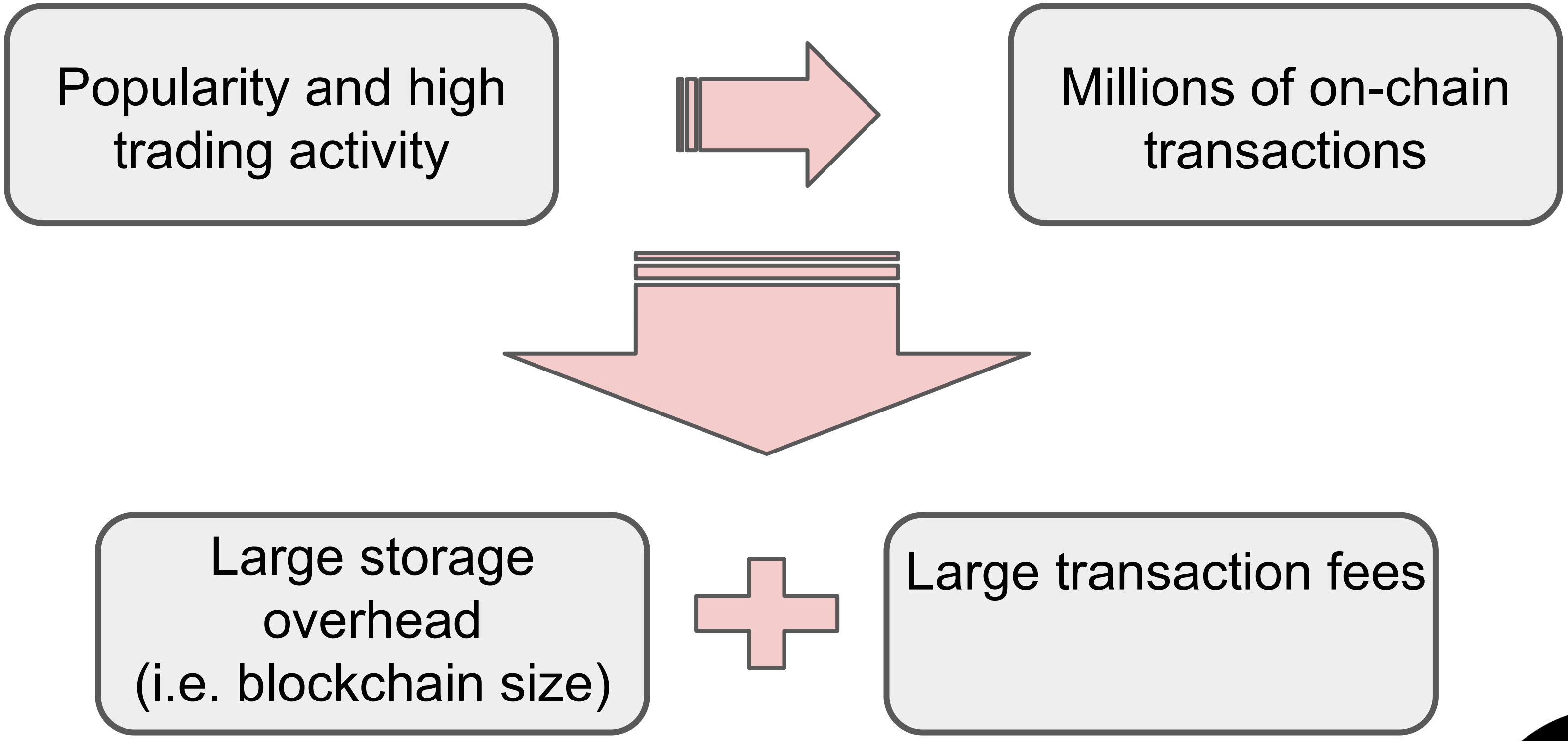
cow Swap

Interesting Topics

- Liquidity
- Maximal extractable value (MEV)
- Optimal trading strategies
- Flash loans
- Pricing functions
- Privacy

Our focus

And a Huge Scalability Problem!



Popularity and high trading activity

Millions of on-chain transactions

Large storage overhead (i.e. blockchain size)

Large transaction fees

Can we control the on-chain state growth while

1. preserving the correct operation of the AMM, and
2. preserving the public verifiability, decentralization, transparency, etc., that are expected of a DeFI protocol?

Limitations of Existing Solutions

Limitations of Existing Solutions

- *Sharding ⇒ How to shard the AMM?*

Limitations of Existing Solutions

- *Sharding ⇒ How to shard the AMM?*
- *Zero-knowledge (ZK) rollups ⇒ ZK proofs are expensive!*
- *Optimistic rollups ⇒ Long contestation periods + incentive compatibility issues!*

Limitations of Existing Solutions

- *Sharding ⇒ How to shard the AMM?*
- *Zero-knowledge (ZK) rollups ⇒ ZK proofs are expensive!*
- *Optimistic rollups ⇒ Long contestation periods + incentive compatibility issues!*
- *Sidechains ⇒ Mainly focused on two-way peg and independent sidechains!*

Still, sidechains have potential to solve the problem!

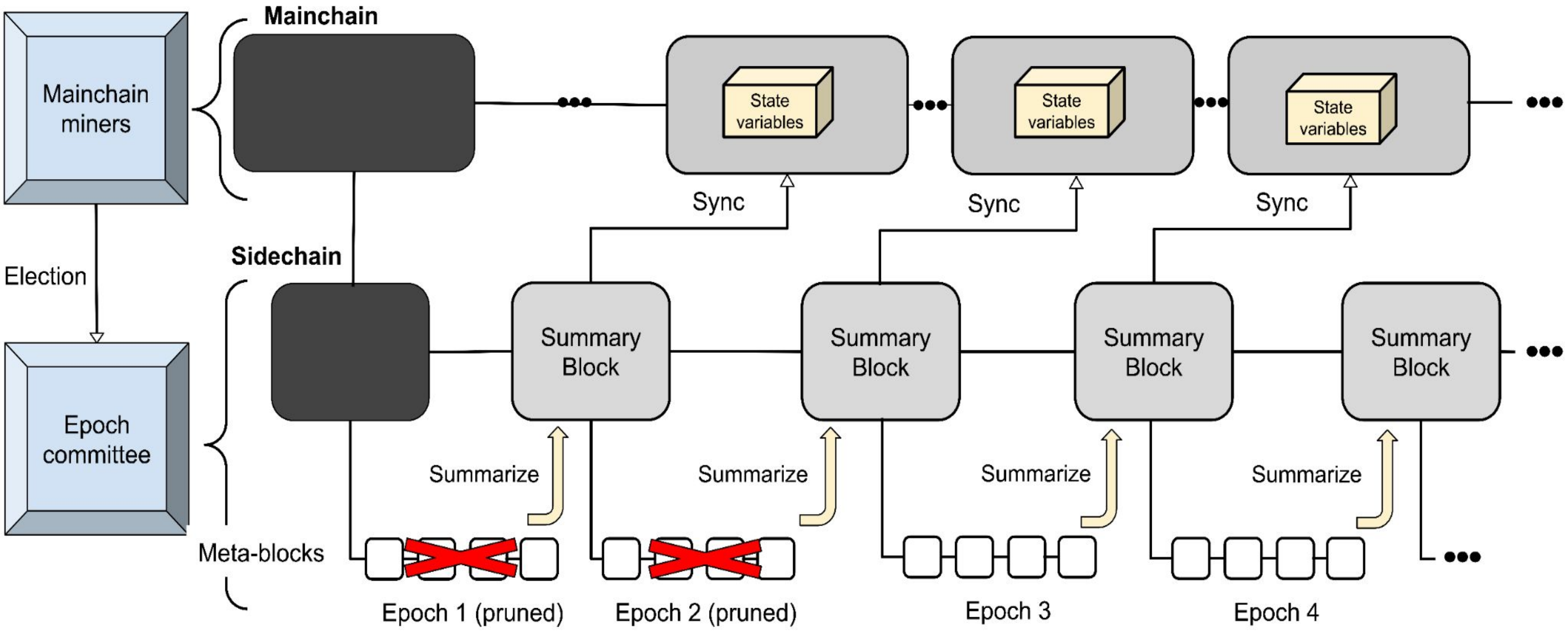
chainBoost—a new dependent sidechain architecture*

*Z. Motaqy, M. Najd, and G. Almashaqbeh, *chainboost: A secure performance booster for blockchain-based resource markets*, in IEEE EuroS&P 2024 (https://arxiv.org/abs/2402.16095).

This work

ammBoost = AMMs + chainBoost

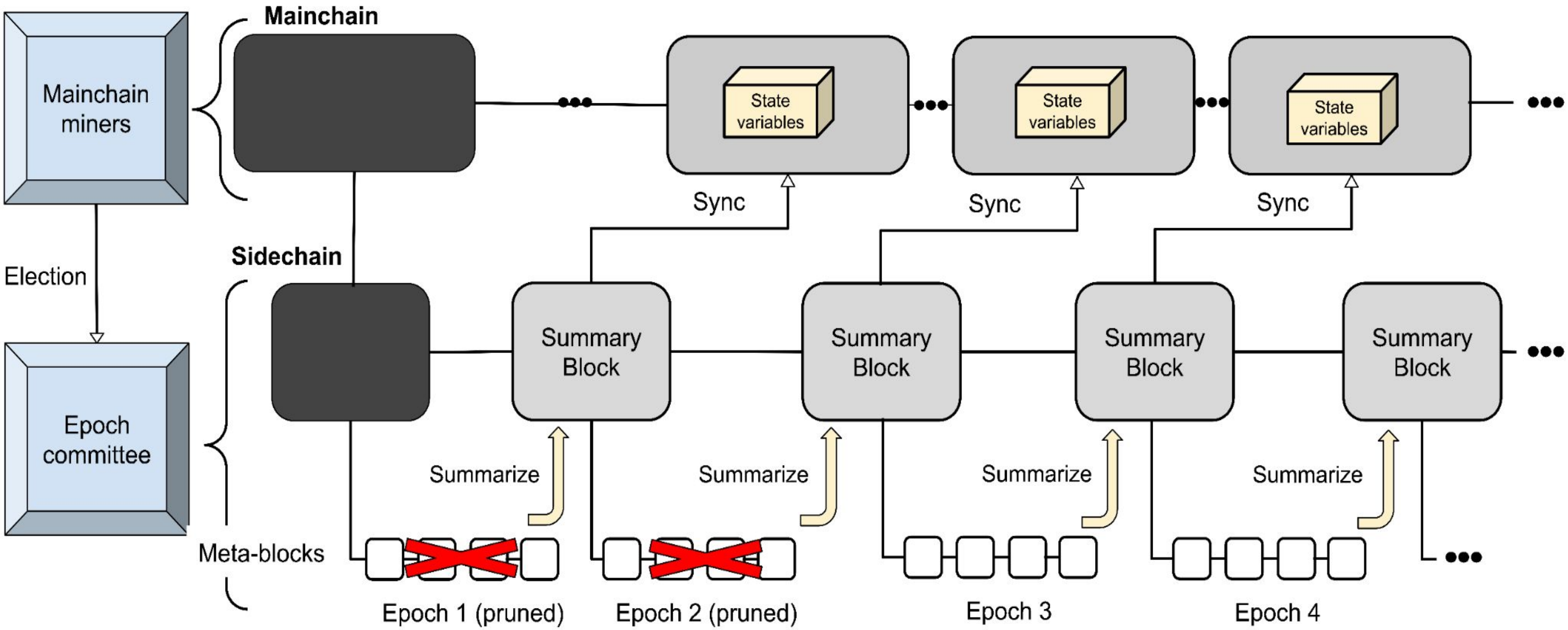
chainBoost Framework



Service-related traffic ⇒ Sidechain

Mainchain traffic ⇒ Mainchain

chainBoost Framework



Service-related traffic ⇒ Sidechain

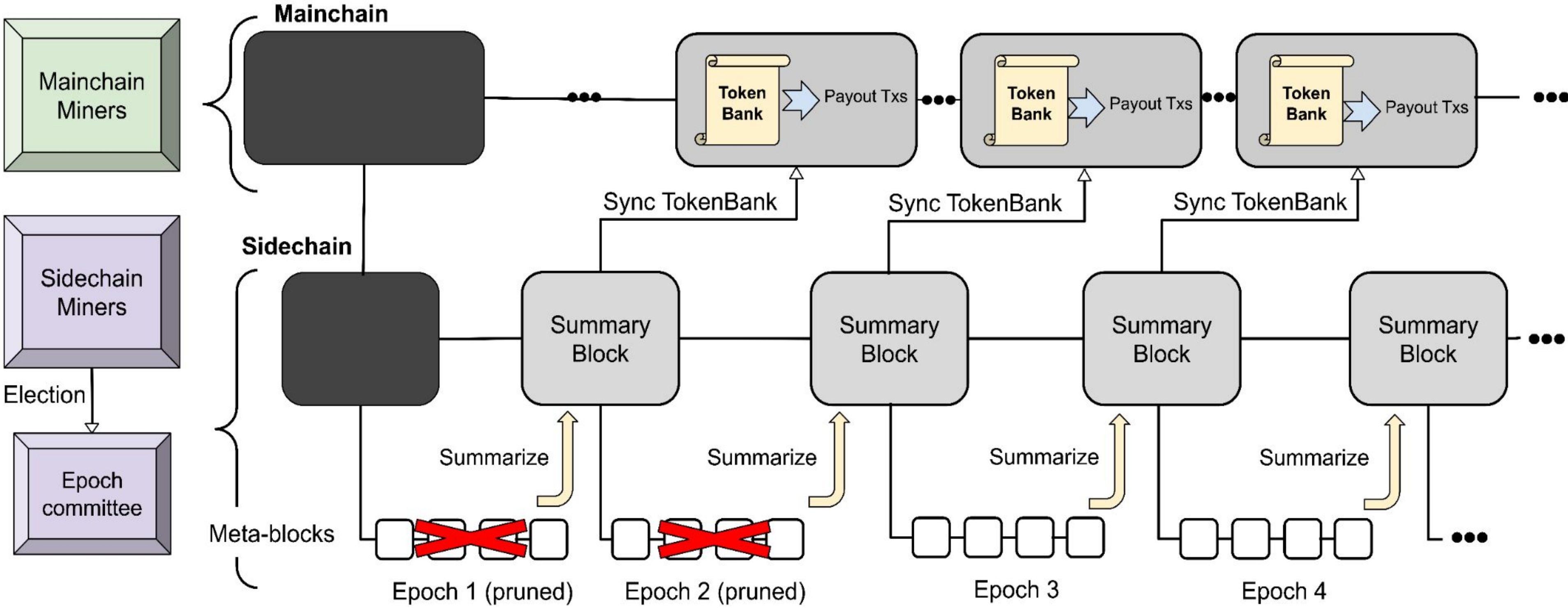
Mainchain traffic ⇒ Mainchain

Mutually-dependent chains

Several Challenges

- Unidirectional dependency on its mainchain
- Mainchain miners are not aware of the sidechain
- No actual tokens on the sidechain
- The syncing process needs authentication

Meet ammBoost!



Swaps, mints, burns, collects ⇒ Sidechain

Deposits, payouts, others (e.g., flashes) ⇒ Mainchain

New Techniques

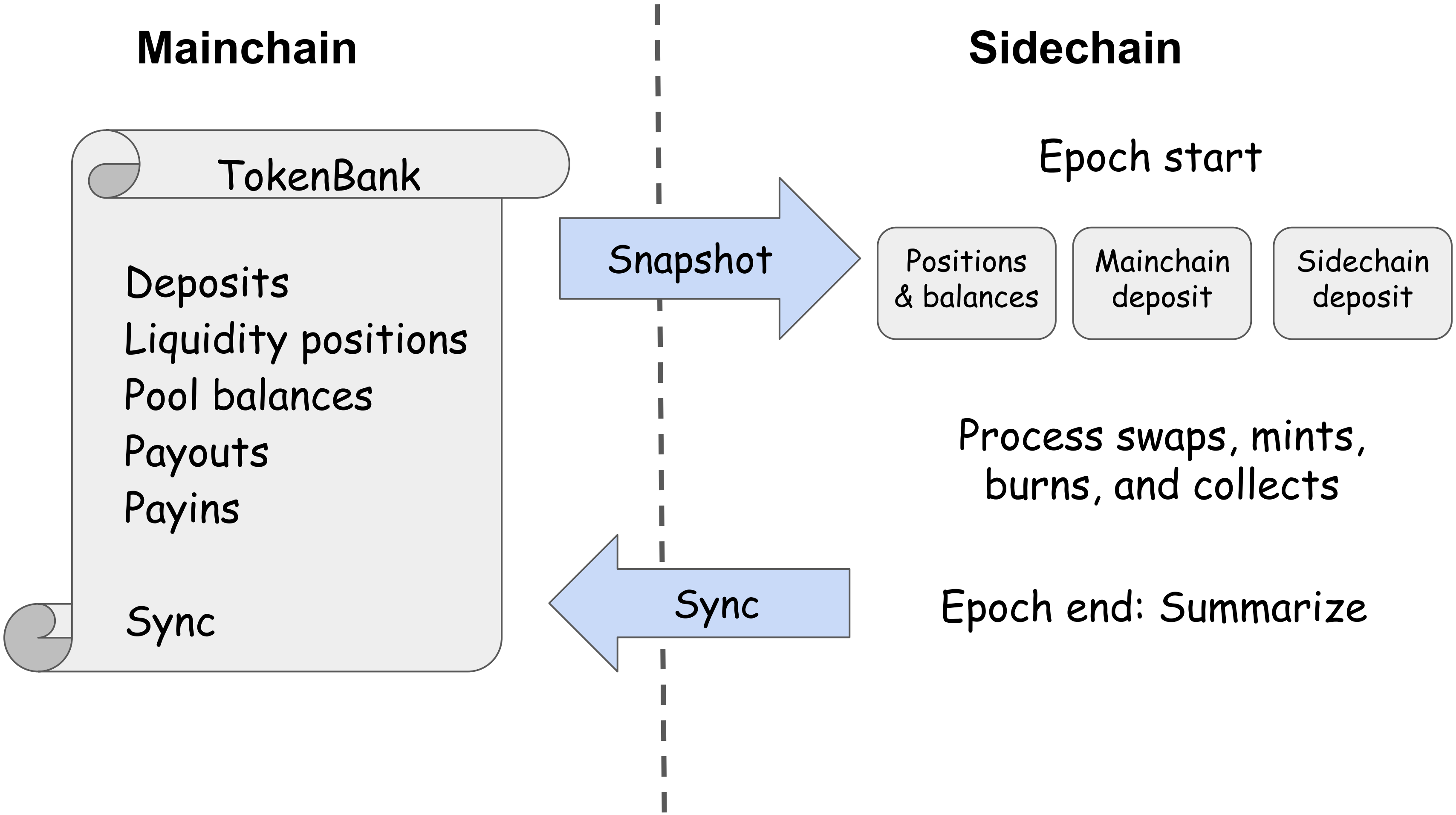
Sidechain miner population

Epoch-based deposits

Snapshot-based and delayed payout trading

Syncing authentication

Workflow



Mainchain

Sidechain

TokenBank

Deposits
Liquidity positions
Pool balances
Payouts
Payins

Sync

Snapshot

Epoch start

Positions & balances

Mainchain deposit

Sidechain deposit

Process swaps, mints, burns, and collects

Sync

Epoch end: Summarize

The Syncing Process

- Includes: updated liquidity positions, updated pool balances, and per-user payouts and payins.
- Done by invoking **`Sync`** in the TokenBank contract.
- TokenBank deducts payins from deposits, and sends payouts to users.
    - Users can withdraw remaining deposits if any.

The Syncing Process

- Includes: updated liquidity positions, updated pool balances, and per-user payouts and payins.
- Done by invoking **`Sync`** in the TokenBank contract.
- TokenBank deducts payins from deposits, and sends payouts to users.
  - Users can withdraw remaining deposits if any.

How to authenticate the Sync call?!

Authentication

Based on *threshold signature-based quorum certificates*:

- Election of committee *e +1* happens during epoch e.
- Committee *e +1* generates a verification key *vk* and shares of the signing key.
  - Sends *vk*, along with election proofs, to committee *e*.
- Committee *e* verifies and records that in a meta-block.
  - Also records *vk* in TokenBank.
- During epoch *e + 1*, committee *e +1* signs **`Sync`** inputs using their shares.
  - TokenBank accepts only if the signature is valid under *vk*.

Security and Performance

- **Security:**
  - Security of sidechain consensus, committee election, BLS threshold signatures, and mechanisms of handling interruptions.
- **Performance evaluation:**
  - A Uniswap-inspired use case.
  - Preliminary results show ~90% reduction of on-chain state size, and scaling to significantly large workloads.

Conclusion and Future Work

- **This work**
  - A secure, sidechain-based framework to control state growth and boost throughput of AMMs.
  - Formal treatment.
  - Implementation/testing.

- **Future work**
  - Look into storage pricing/transaction fees.
  - Extend ammBoost utility to support privacy/anonymity, and functionality extensions.

Thank you!

Questions?

Ghada Almashaqbeh: ghada@uconn.edu
https://ghadaalmashaqbeh.github.io/

Paper full version: coming soon!