SPEEDEX: A Scalable, Parallelizable, and Economically Efficient Distributed EXchange

GEOFFREY RAMSEYER, DAVID MAZIÈRES, ASHISH GOEL
Digital Currency Infrastructure for the next 100 years

- Interoperability
  - Efficiently Exchangeable
- Scalable Infrastructure
- Auditability, Reliability
Outline

• Challenges in Existing DEX design

• What is SPEEDEX?
  • Many-Asset Batch Exchange
  • Scalable, High Throughput
  • Replicable, Auditable ⇒ Decentralized (in a Blockchain)
Existing Exchanges: Continuous Double-Auctions

Existing Exchanges: Pairwise Trading

My Currency A

1 A = 10 USD

1 A = 8.9 GBP

Target Currency B

1 GBP = 170 YEN

45 YEN = 1 B

1 USD = 1.01 EURO

1 EURO = 3 B
Swap

1
$12,976.80

19960.4

1 DAI = 0.0000501 WBTC

Expected Output: 19960.4 DAI
Price Impact: 0.04%
Minimum received after slippage (0.30%): 19900.7 DAI
Network Fee: ~$2.06

Auto Router

Best price route costs ~$2.06 in gas. This route optimizes your total output by considering split routes, multiple hops, and the gas cost of each step.

Connect Wallet

Routing

40% >

USDC
Clipper RFQ 100%

Balancer V2 100%

60% >

USDT
Curve V2 100%

DAI
DODO v2 100%

https://app.uniswap.org/#/swap

https://app.1inch.io/#/1/unified/swap/WBTC/DAI
What is SPEEDEX?

Scalable
Parallelizable
Economically Efficient
Distributed EXchange
SPEEDEX Batch Trading

• How can we trade many assets all in one batch?

• Answer:
  • Model a batch as an Arrow-Debreu Exchange Market
  • Compute Equilibrium (Prices \(\{p\}\), Allocations \(\{x\}\))

\[
\begin{align*}
  u(x) &= 1.2 \ x_\varepsilon + x_\$, \\
  x_\$ + \frac{p_\varepsilon}{p_\$} x_\varepsilon &\leq 10
\end{align*}
\]

Sell $10 for at least €1.20/$1
Why Many-Asset Batch Trading?

- No Risk-Free Front Running
  - Trade at equilibrium price, not limit price
- No Routing Problem, No Internal Arbitrage
  - No reason to hold intermediate assets
  - $\frac{p_A}{p_B} \times \frac{p_B}{p_C} = \frac{p_A}{p_C}$
- Liquidity between infrequently traded assets
Why Many-Asset Batch Trading?

• SPEEDEX trading logic is unaffected by trade ordering
• Trade offers in one block *commute* with one another
• Scalability through Parallelism
Core Challenge: Equilibrium Computation

- Number of assets $\ll$ Number of Trade Offers
- $\exists$ Convex Program [DGV16]
- Iterative method: Tâtonnement [CMV05]
  - Query “How close is a set of prices to equilibrium?”
  - Adjust (Economics 101)
  - Repeat
- Round Results with Linear Program
Core Challenge: Equilibrium Computation

• Need ~1,000-10,000 loop repetitions for high accuracy
• Expensive: “Try to clear market with candidate valuations”
• Sort + preprocess + binary search: $O(\log(\text{#offers}) \times \text{#assets}^2)$
• Linear program size: $O(\text{#assets}^2)$
Overall Transaction Rate
SPEEDEX: A New Distributed Exchange

• Batch offer matching eliminates internal arbitrage and front-running.
• Replicable and Auditable: Can run in a blockchain “Layer-1”
• Scalability via a transaction semantics and a trading engine in which operations commute.
• More than 200,000 transactions/second on a 48-core machine with 10s of millions of open offers
Open Questions

• Efficient Market-Making?
• Cost of Liquidity Provision?
• Batch Frequency?

geoff.ramseyer@cs.stanford.edu