Fair combinatorial auction for trade intents

how to design mechanisms without a numeraire

Andrea Canidio

CoW DAO

Numeraire or no numeraire?

Numeraire: an asset that everybody values

- **Offchain**: there is usually a numeraire
  - Example: stocks are traded against the national currency
- Possible exception: forex markets?
  - 100 different currencies, with the top 5 pairs accounting for about 65% of total volume (all involve the US Dollar).

- **Onchain**: usually no numeraire
  - Anonymous participants, could be anywhere in the world
  - May want to swap any asset for any other asset
  - There are thousands of different assets

The absence of a numeraire makes harder to share the gains from collaboration

Example: **trade intent auctions**

- There are additional efficiencies if orders are executed together
- But these efficiencies may materialize in an asset that one of the users does not want
- Do you want to execute the two orders together? And if you do, how do you share the benefit?

New research: *Combinatorial Auctions without a Numeraire: The Case of Blockchain Trade-Intent Auctions*

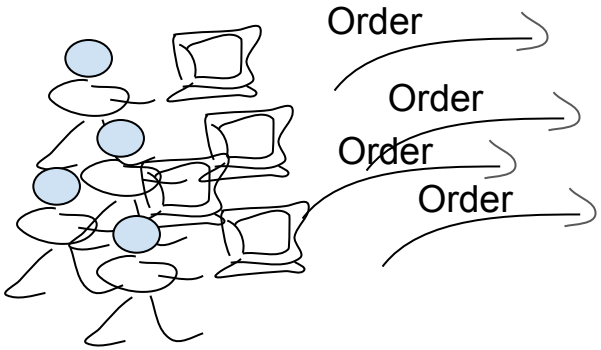
1. A theoretical model of trade intent auction
2. Use it to compare the two most-common auction formats for trade intent: Batch auction vs separate individual auctions
3. Propose a new auction format: the fair combinatorial auction
    - A merge between batch auction and individual trade intent auction
    - If properly designed, the fair combinatorial auction can provide strong farness guarantees

New research: *Combinatorial Auctions without a Numeraire: The Case of Blockchain Trade-Intent Auctions*

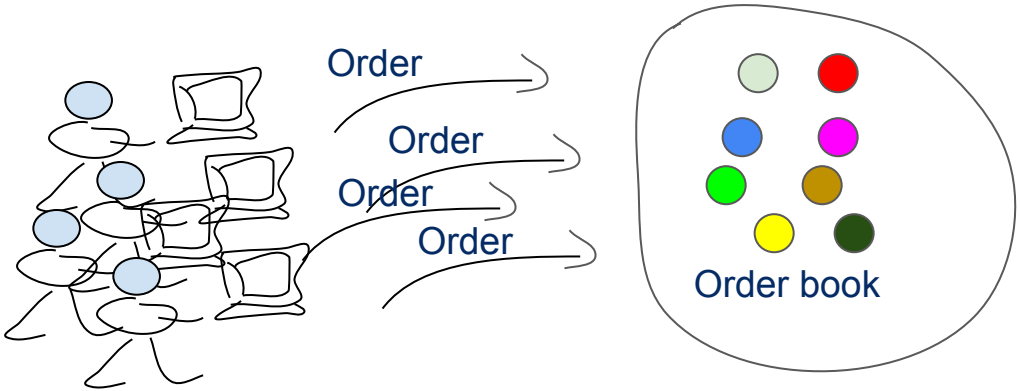
1. ~~A theoretical model of trade intent auction~~
2. ~~Use it to compare~~ *High level discussion* of the two most-common auction formats for trade intent: Batch auction vs separate individual auctions
3. Propose a new auction format: the fair combinatorial auction
    - A merge between batch auction and individual trade intent auction
    - ~~If properly designed, the~~ *a* fair combinatorial auction ~~can~~ that provides strong farness guarantees

Batch auctions (vs separate individual auctions)

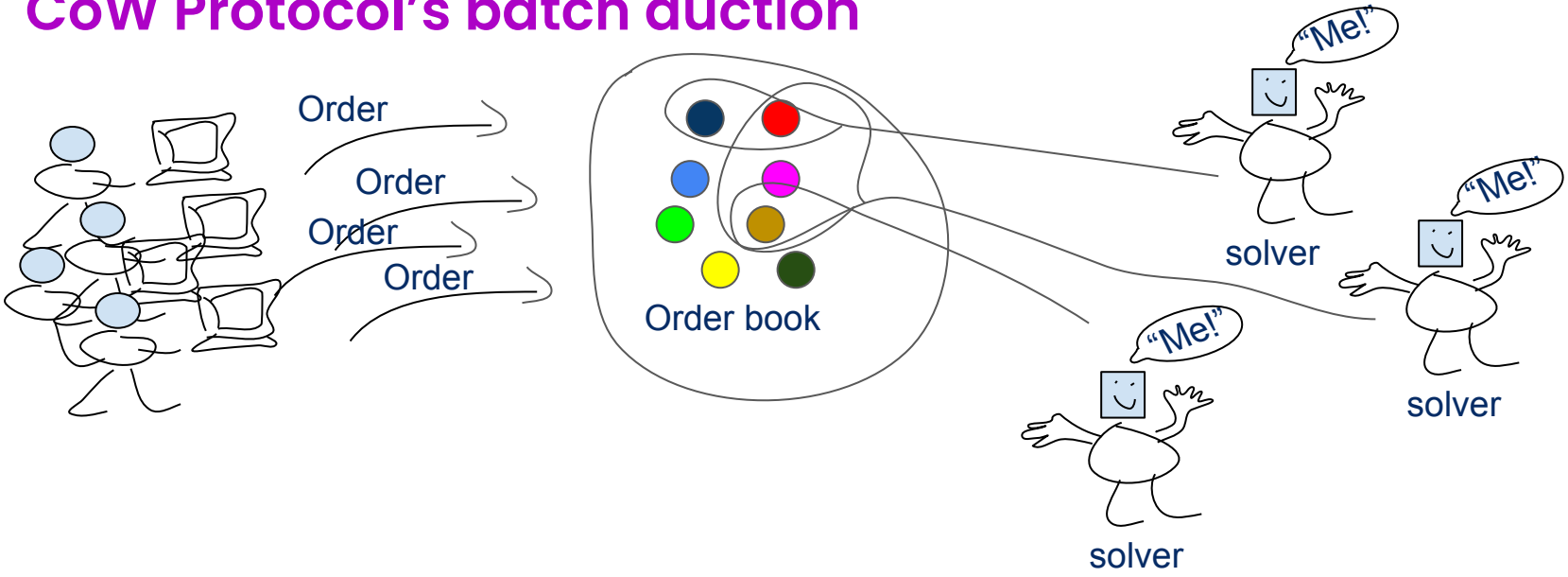
CoW Protocol’s batch auction



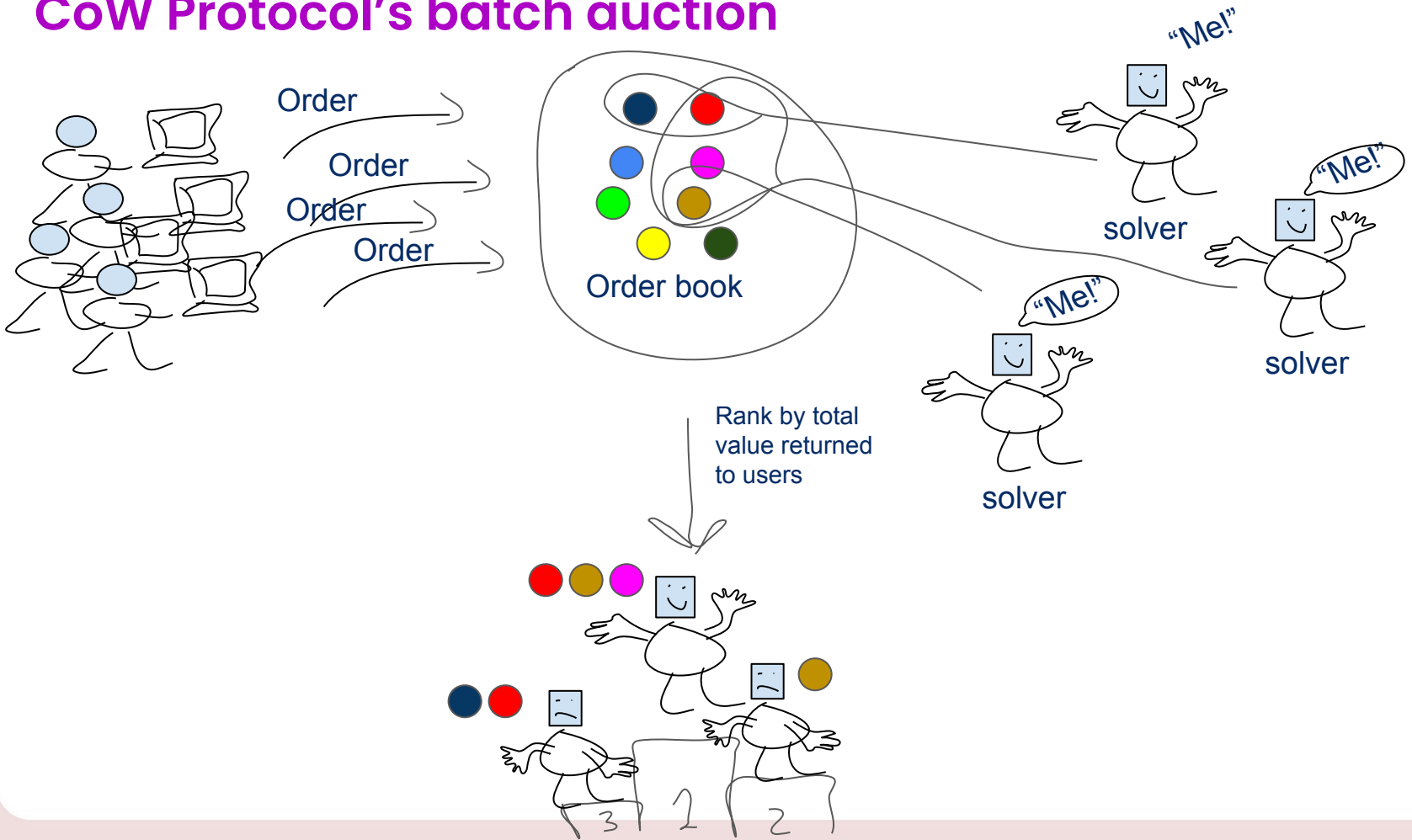
CoW Protocol’s batch auction



CoW Protocol's batch auction



CoW Protocol's batch auction



CoW Protocol's batch auction (vs other protocols)

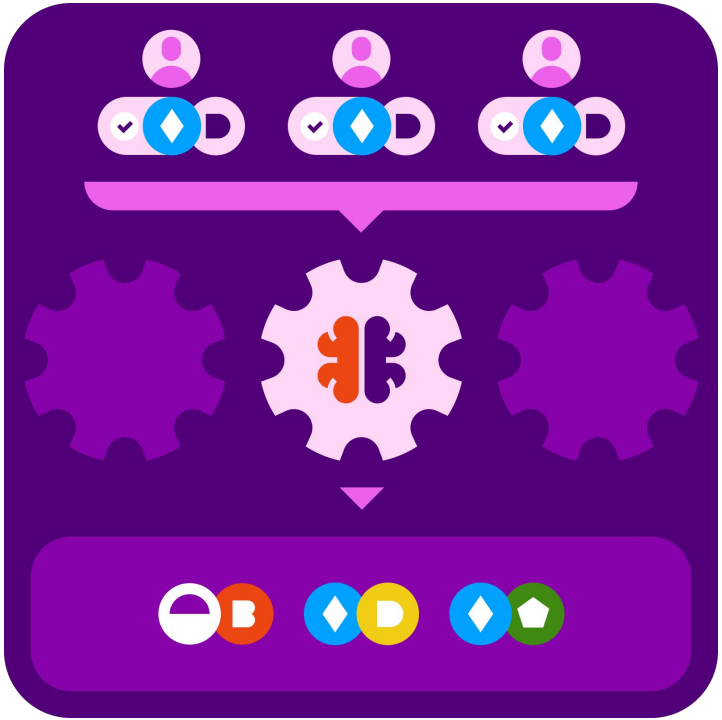
Collects trade intents

Runs an auction among "solvers":

- Each solver proposes a price for each trade on the batch
- The best solution wins
- (note: you need an oracle price)

Other protocols: 1inch fusion and Uniswap X

- a separate (Dutch) auction for each trade



Batch auctions vs individual-trades auctions

Batch auctions: better at exploiting **complementarities between different trades**

Batch auctions: better at generating **competition between solvers**

Individual trades auctions: better at exploiting **solvers' specialization**

Batch auctions can be unfair, i.e., worse for some traders than an individual-trade auction

- Current fix: EBBO

The billion dollars question

Is there a mechanism that can result in **batching or specialization** (depending on what is most efficient), **generates competition** between solvers, and is **fair**?

Fair Combinatorial Auction

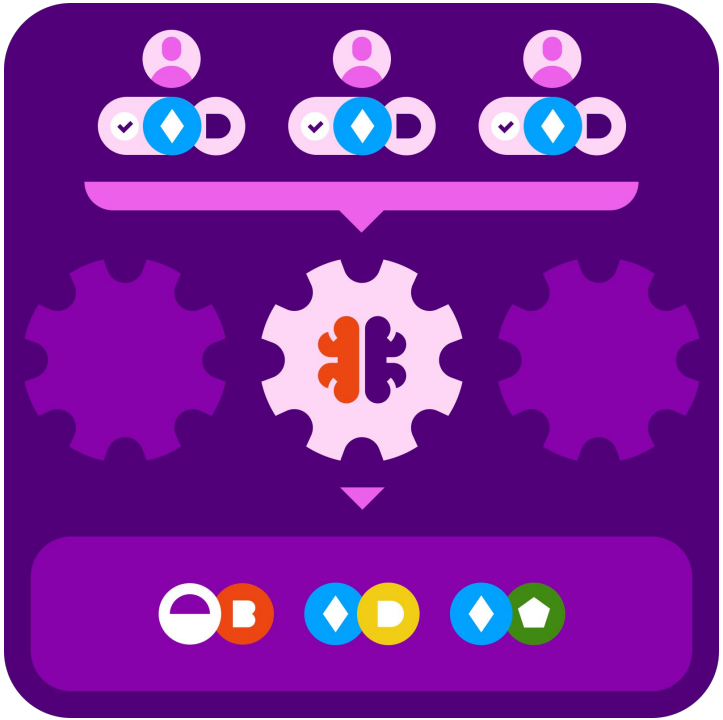
Fair Combinatorial Auction

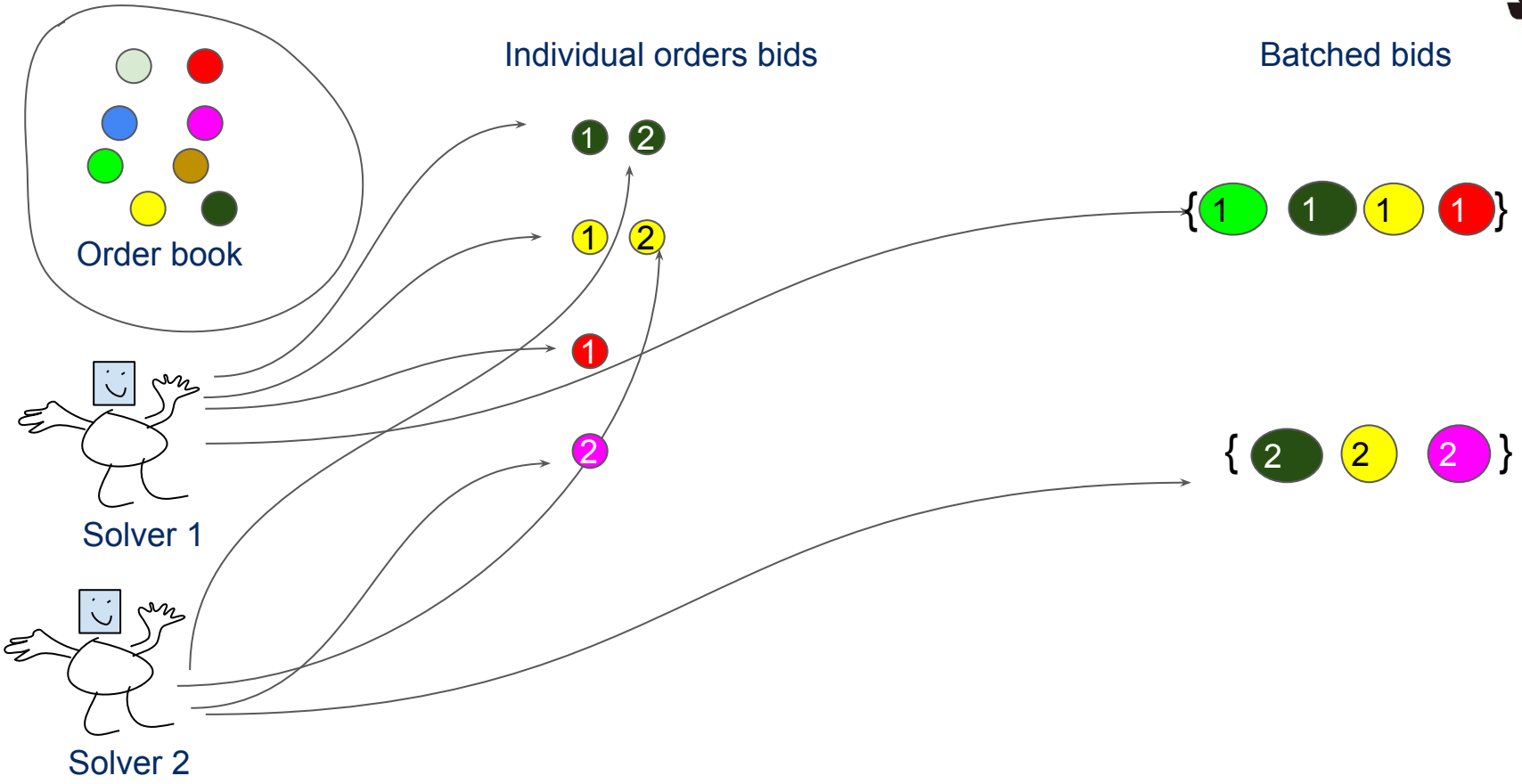
Multiple bids per solver: each solver can bid both on individual trades and on batches of trades

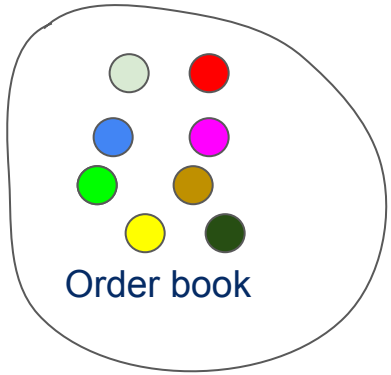
Filtering of batched bids:

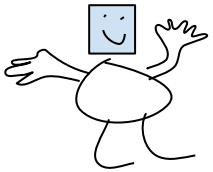
- Consider only the individual trade bids
- Calculate the outcome of several simultaneous ***first price**** auctions with those bids -> reference outcome
- Filter out a batched bid if it is worse than the reference outcome for at least one trader

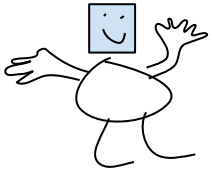
Choose the winner(s) among the surviving batched bids & individual trade bids




Individual orders bids
Batched bids
Order book
Solver 1
Solver 2
1
2
1
2
1
2
{ 1 1 1 1 }
{ 2 2 2 }




Solver 1


Solver 2

Individual orders bids

max(1 ,2)= * (reference outcome)

max(1 ,2)= * (reference outcome)

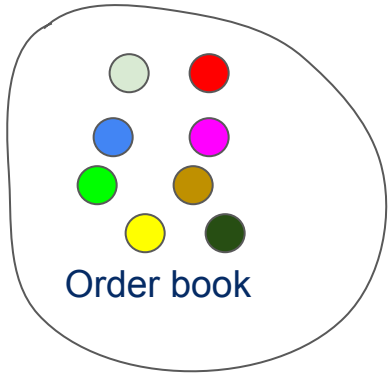
max(1 ,)= * (reference outcome)

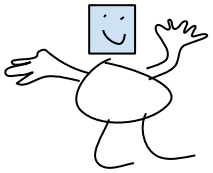
max(2 ,)= * (reference outcome)

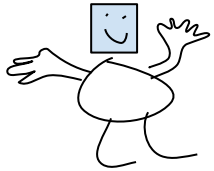
Batched bids

{1 1 1 1}

{ 2 2 2 }







Individual orders bids

max(1 ,2)= *

max(1 ,2)= *

max(1 ,)= *

max(2 ,)= *

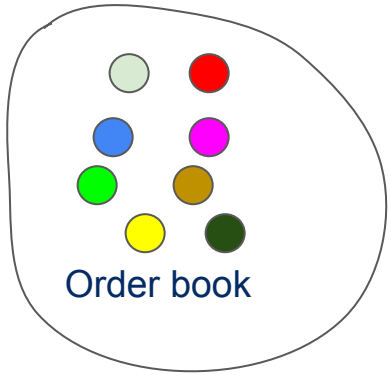
Batched bids

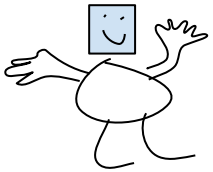
{1 1 1 1}

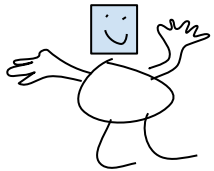
Filter out if 1 < * or 1 < * or 1 < *

{ 2 2 2 }

Filter out if 2 < * or 2 < * or 2 < *







Individual orders bids

max(1 ,2)= *

max(1 ,2)= *

max(1 ,)= *

max(2 ,)= *

Batched bids

{1 1 1 1}

Filter out if 1 < * or 1 < * or 1 < *

{ 2 2 2 }

Filter out if 2 < * or 2 < * or 2 < *

Final winners: {1 1 1 1} 2

Fair combinatorial auction: strategic considerations

- **higher individual trade bids** (than in a standard first price auction) to filter out the opponents' batched bids

- **lower batched bids** (than in a batched auction) if they think that their opponents' batched bid may be filtered out

Fair combinatorial auction: In equilibrium

- **Fairness guarantees:** Solvers return **strictly more** to the traders than in simultaneous first price auctions
- Batching or specialization: depending on their relative benefits
- If the equilibrium is batching, the winning solver may return **less** to the traders than in the batch auction

Fairness guarantees, sometimes at the expense of the total value returns to users

Conclusions

- *Trade intent auction are the future*
- *There are efficiencies from batching*
- *Fairness is a concern*
- *Solution: the fair combinatorial auction*

Paper:

Thank you!

Stay in touch with us!
Moooo

X: **@CoWSwap**
Github: **@CoWProtocol**

Read more: **cow.fi**
Join us: **cow.fi/careers**