Maximal Matching for Double Auction

Dongmo Zhang[1] Dengji Zhao[1,2] Md Khan[1] Laurent Perrussel[2]

[1] Intelligent Systems Laboratory
University of Western Sydney, Australia

[2] IRIT, University of Toulouse, France

AI'10 - Dec 2010

Stock Exchanges

How to Choose Stock Exchanges

Question

Which stock market will you choose?

How to Choose Stock Exchanges

Question

Which stock market will you choose?

The one...

- you can earn more money
- has higher chance to get traded
- most other people go

How to Choose Stock Exchanges

Question

Which stock market will you choose?

The one...

- you can earn more money
- has higher chance to get traded
- most other people go

How do you know?

How to Choose Stock Exchanges

Question

Which stock market will you choose?

The one...

- you can earn more money
- has higher chance to get traded
- most other people go

How do you know?

- **Market Liquidity**

What is Market Liquidity?

1. number of transactions
2. trade volume (buy/sell-volume)
   - the sum of the price of transacted orders

What is Market Liquidity?

1. number of transactions
2. trade volume (buy/sell-volume)
   - the sum of the price of transacted orders

Question

Can a stock market owner improve market liquidity to get more traders and more profit?

What is Market Liquidity?

1. number of transactions
2. trade volume (buy/sell-volume)
   - the sum of the price of transacted orders

Question

Can a stock market owner improve market liquidity to get more traders and more profit?

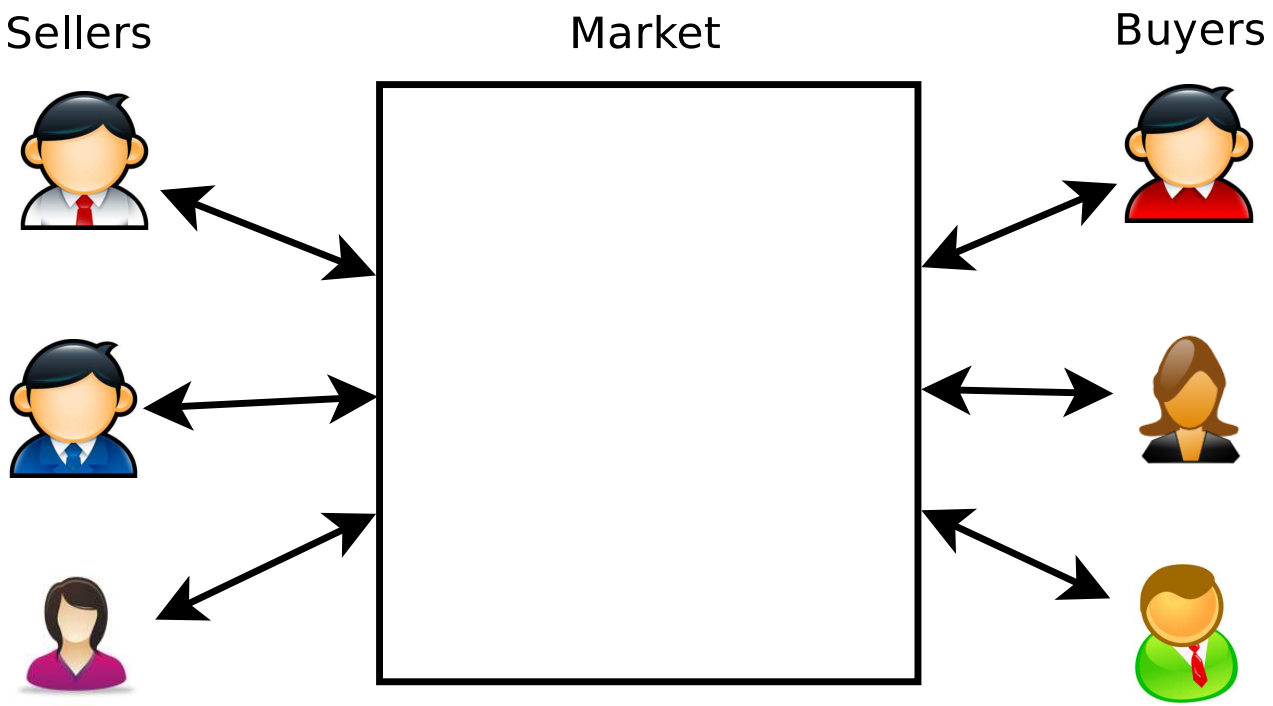
Double Auction

Outline

Outline

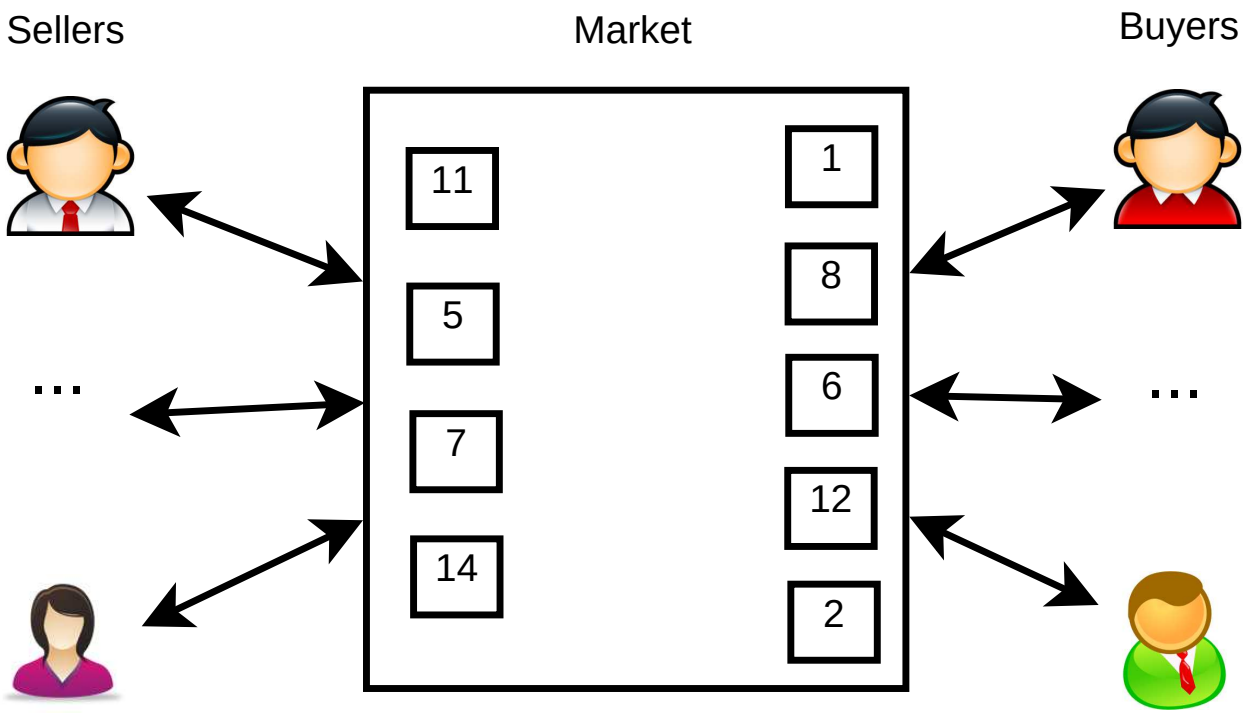
Model

- Three roles: **seller**, **buyer**, and **market maker**.
- One commodity, e.g. google's stocks.



Model

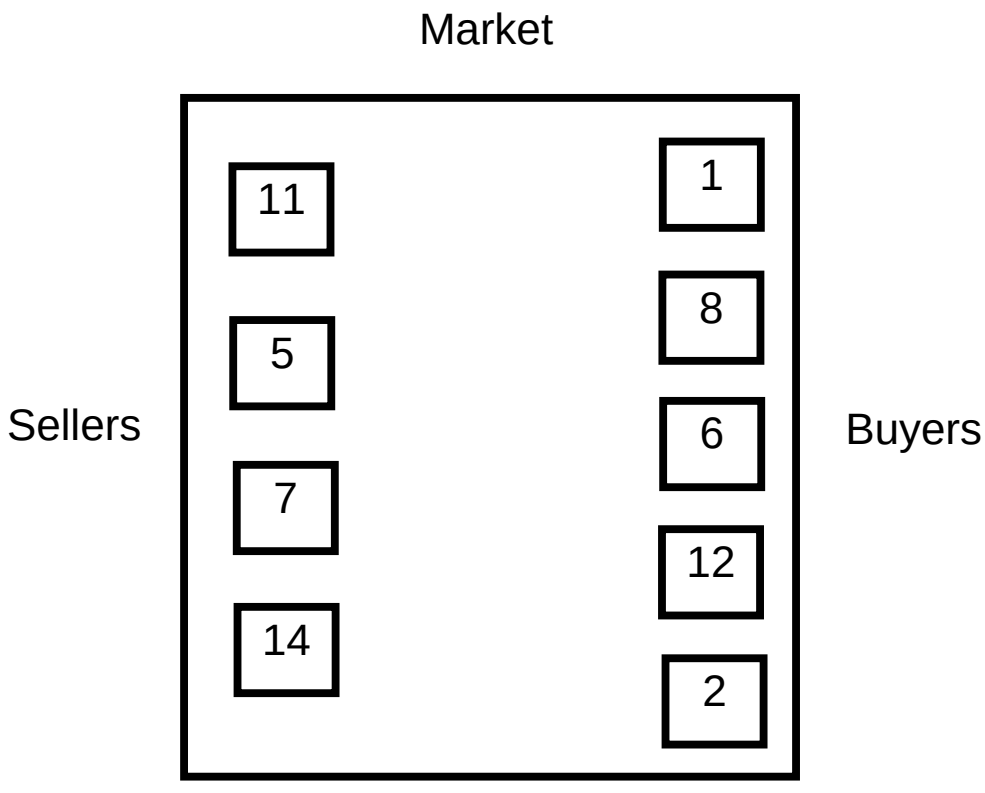
- Three roles: **seller**, **buyer**, and **market maker**.
- One commodity, e.g. google's stocks.



Exchanging Rules

For market maker:

- which sell offer to be matched with which buy offer?
- what is the price for each match?

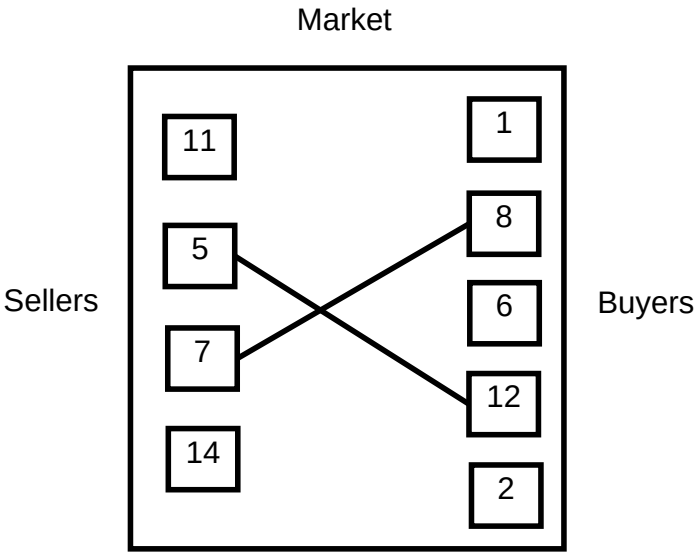


Definitions

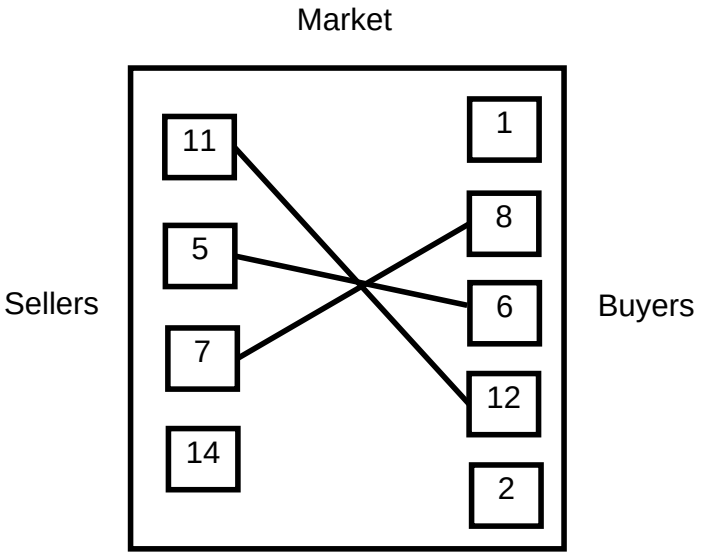
- **Ask**: offer (price) submitted by a seller, the minimum price willing to sell
- **Bid**: offer (price) submitted by a buyer, the maximum price willing to buy
- **Matching**: a set of pairs of ask and bid, where in each pair
  - bid's price $\geq$ ask's price
  - no bid or ask belongs to more than one pair
- **Market Liquidity**:
  - number of transactions (matching size)
  - trade volume (buy/sell-volume)
    - buy-volume: the sum of the price of transacted bids
    - sell-volume: the sum of the price of transacted asks
- **Auctioner's Profit**: The price difference between matched bids and asks

Matching Examples

- bid's price $\geq$ ask's price
- no bid or ask belongs to more than one pair



Profit = (12+8)-(5+7) = 8



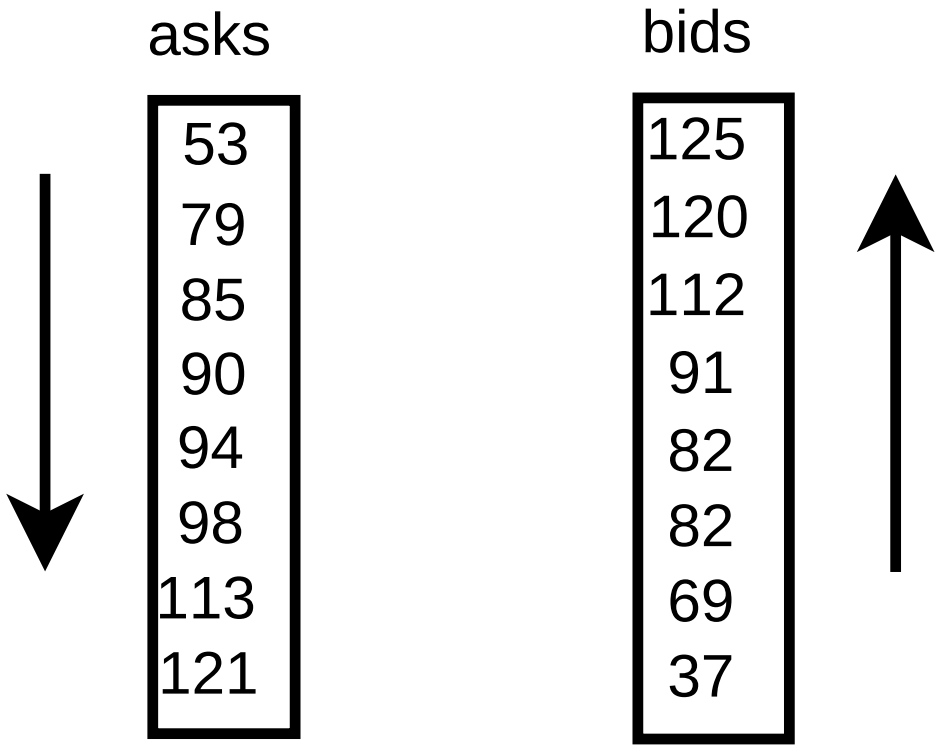
Profit = (12+8+6)-(5+7+11) = 3

Outline

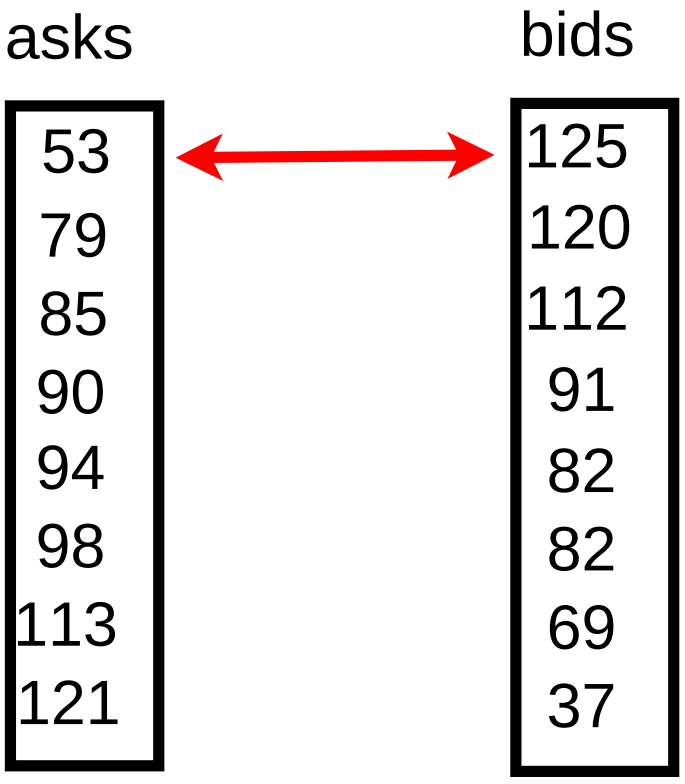
Main Idea

1. Sort all asks (bids) in ascending (descending) order w.r.t. their price.
2. Based on this sort order, starting at the top, add each ask-bid pair to the result matching, if ask's price $\leq$ bid's price.

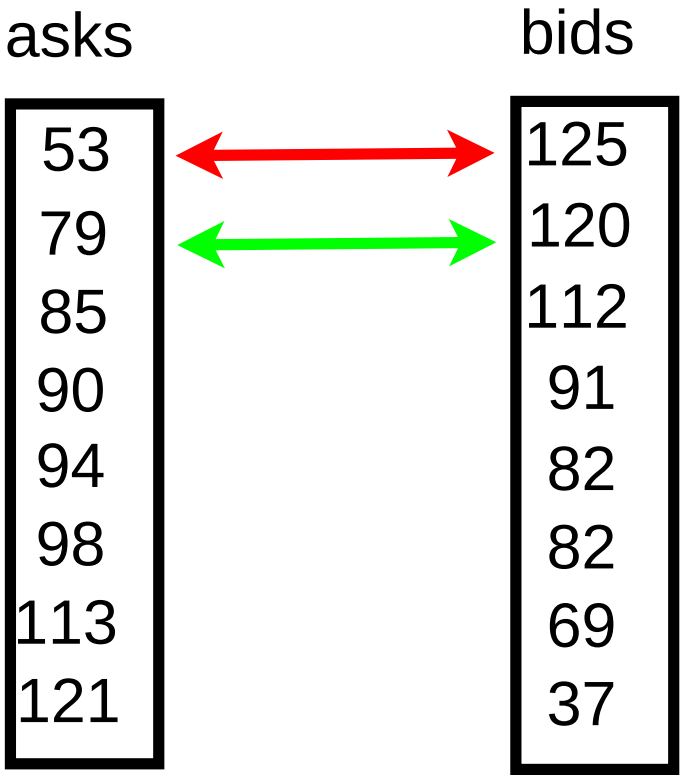
Main Idea



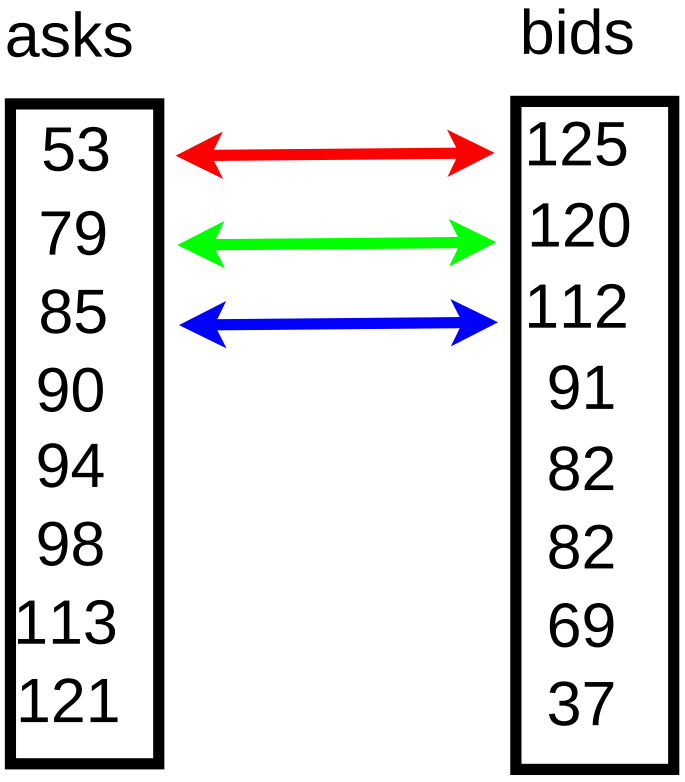
Main Idea



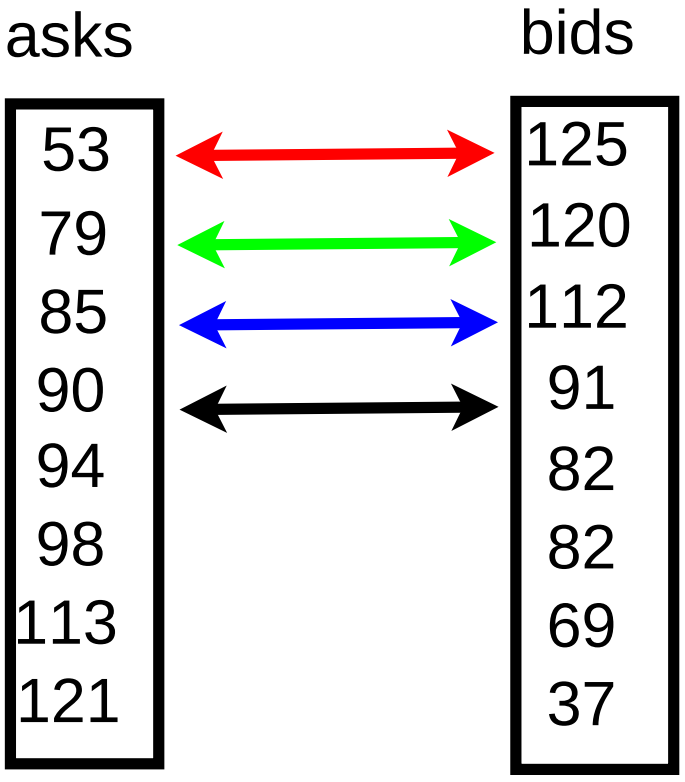
Main Idea



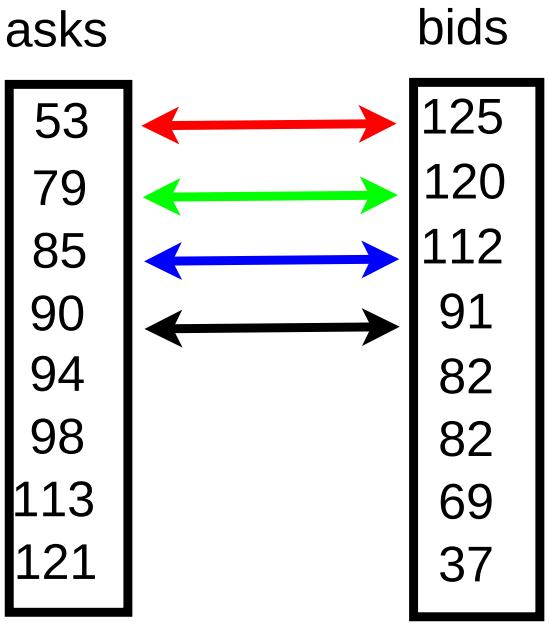
Main Idea



Main Idea

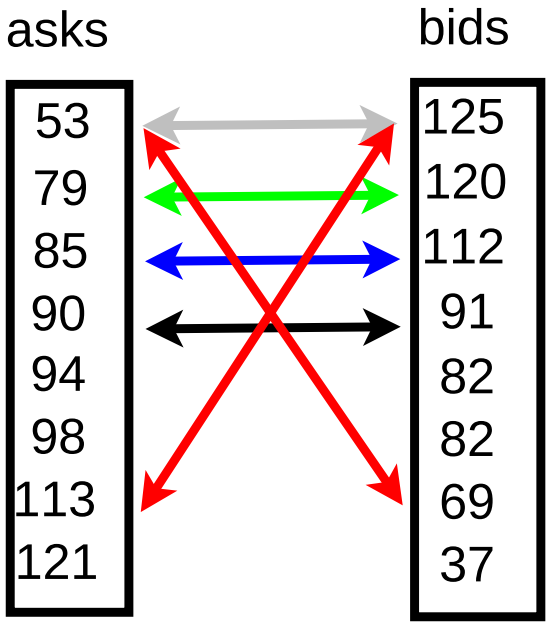


Properties of Equilibrium Matching



1. profit maximizing (141)
2. market liquidity can be improved
   1. transactions: 4
   2. buy/sell-volume: 448/307

Properties of Equilibrium Matching



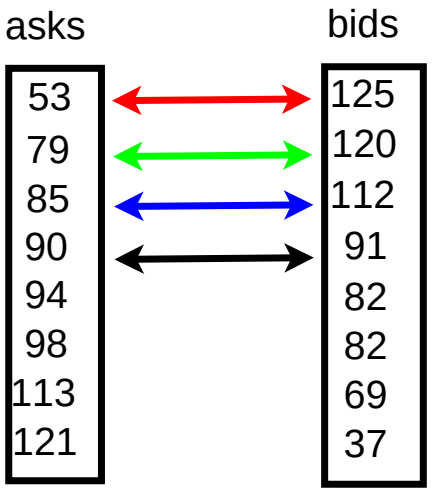
1. profit maximizing (141)/97
2. market liquidity can be improved
   1. transactions: 4
   2. buy/sell-volume: 448/307

Outline

What We Want?

1. Maximizing market liquidity
2. Keeping as much profit as we can

General Idea



Equilibrium Matching

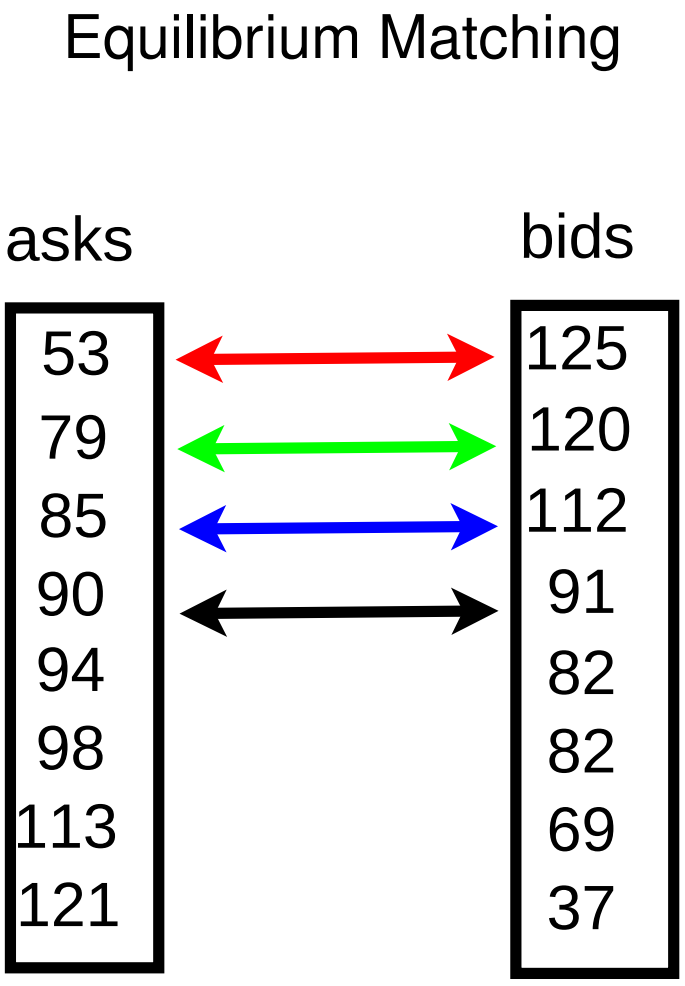
1. Starting from equilibrium matching
2. Matching unmatched shouts (asks and bids) as many as we can
3. Decreasing profit as less as we can

Looking Extra Matchable Shouts

Question

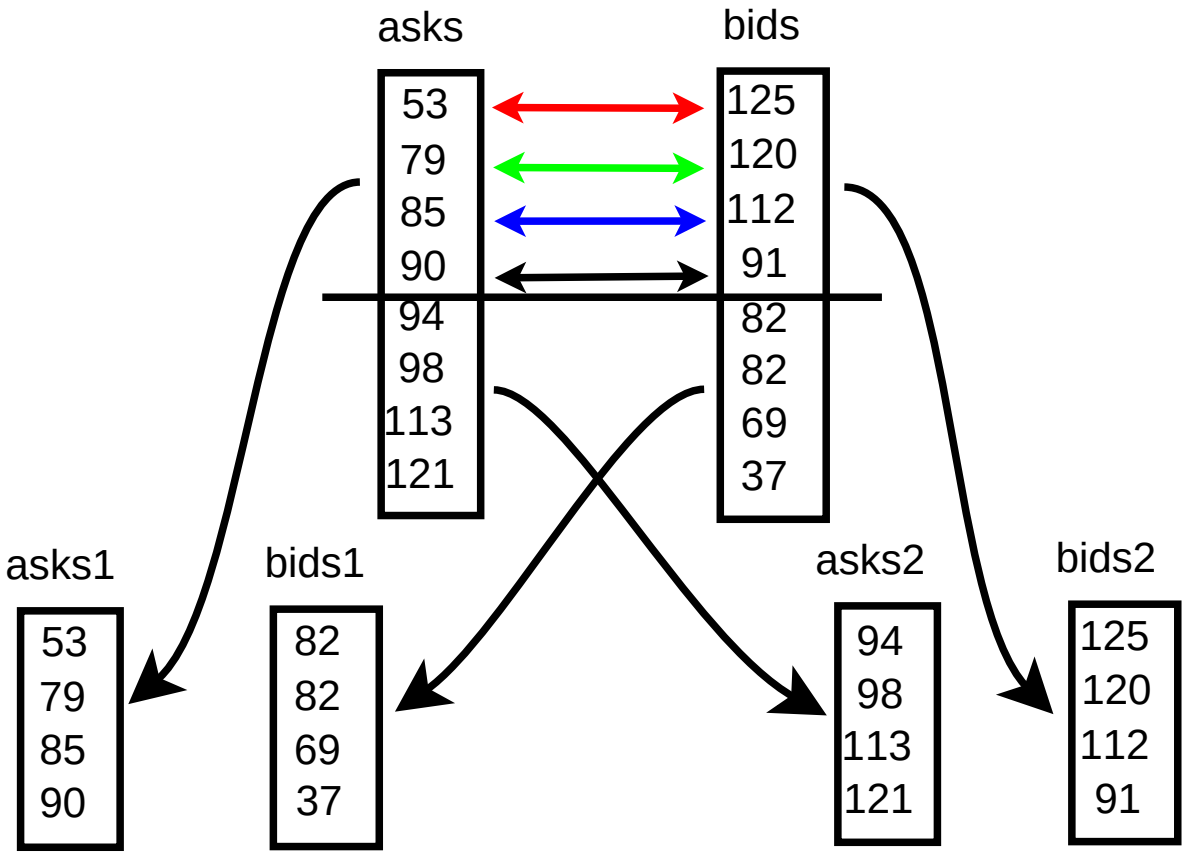
How to find extra matchable shouts?

Looking Extra Matchable Shouts



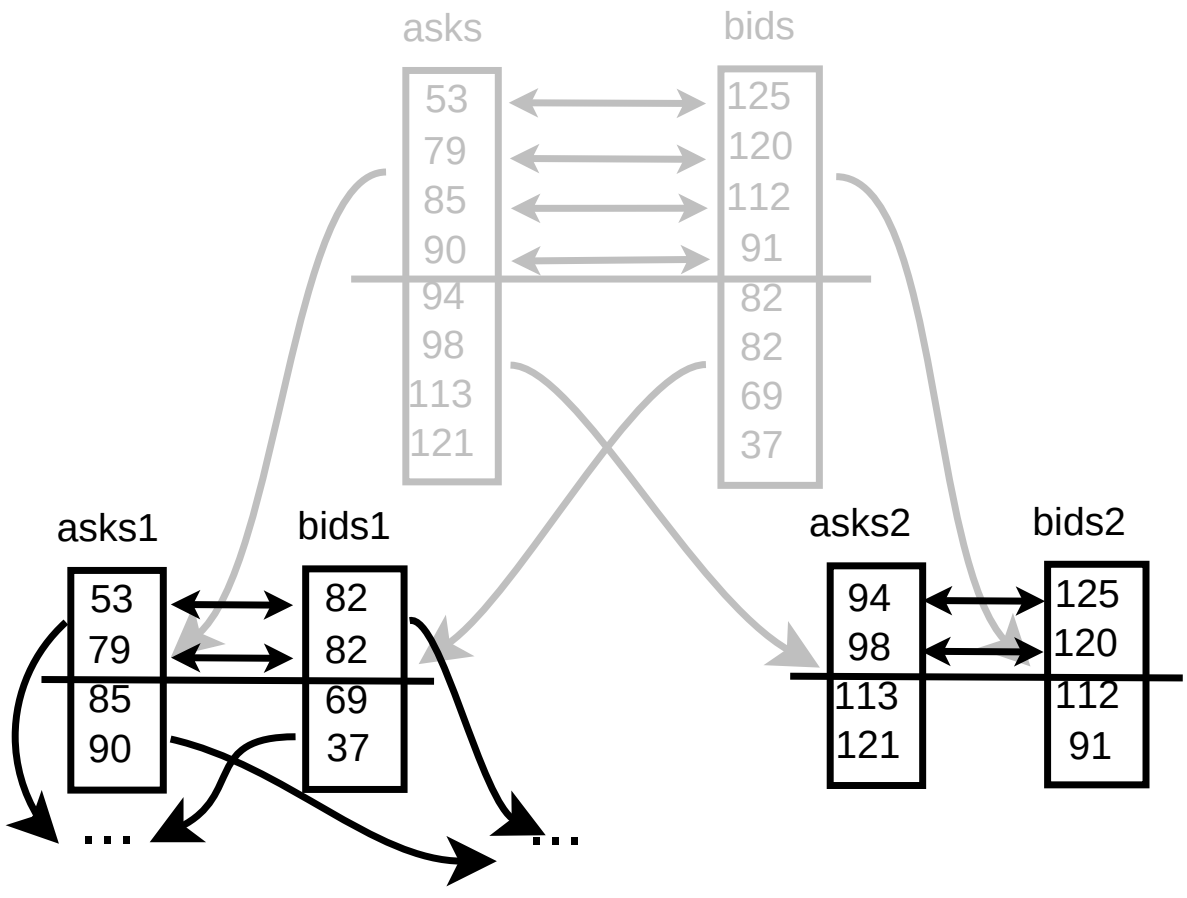
Looking Extra Matchable Shouts

Decompose Matching



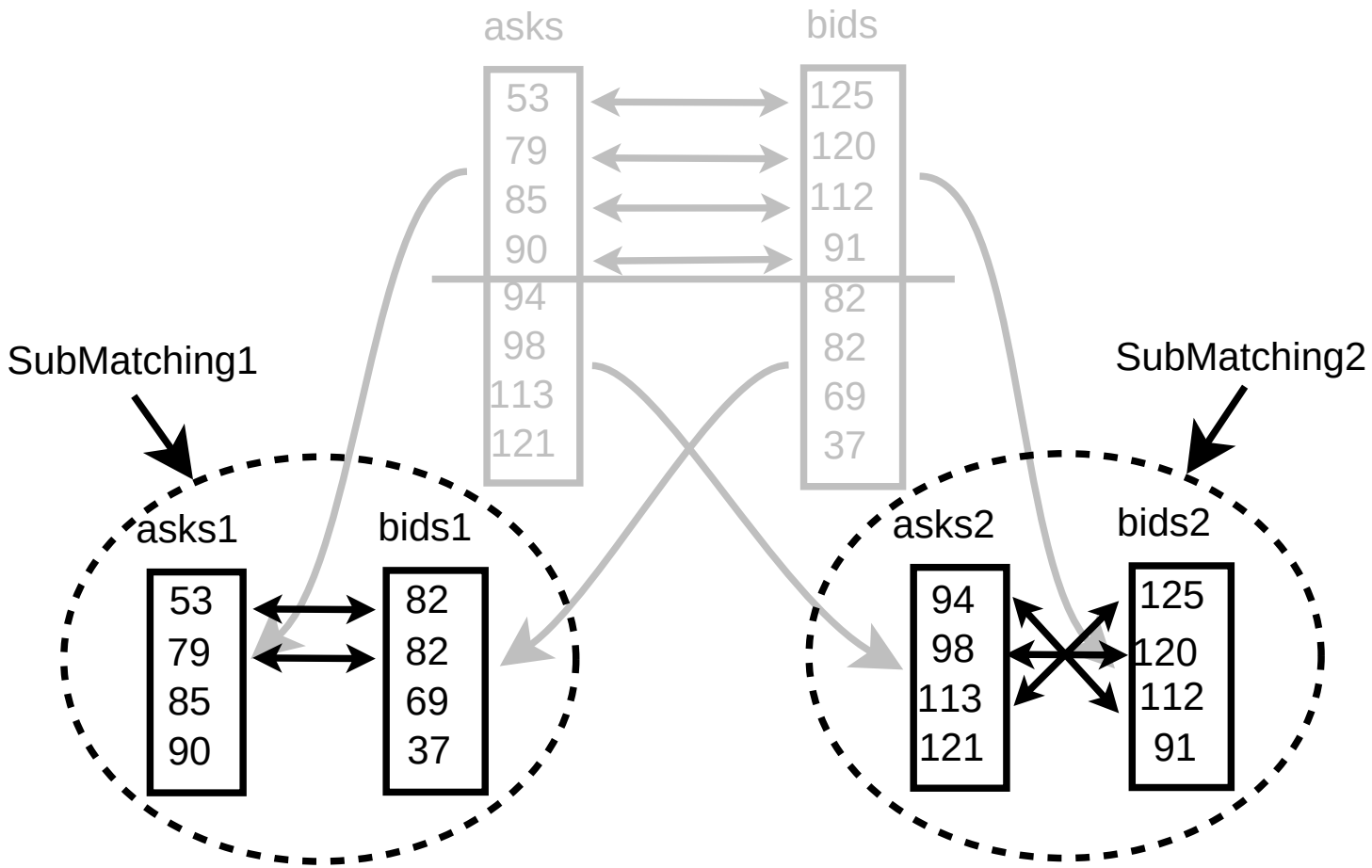
Looking Extra Matchable Shouts

Maximal Matching in Sub-matchings

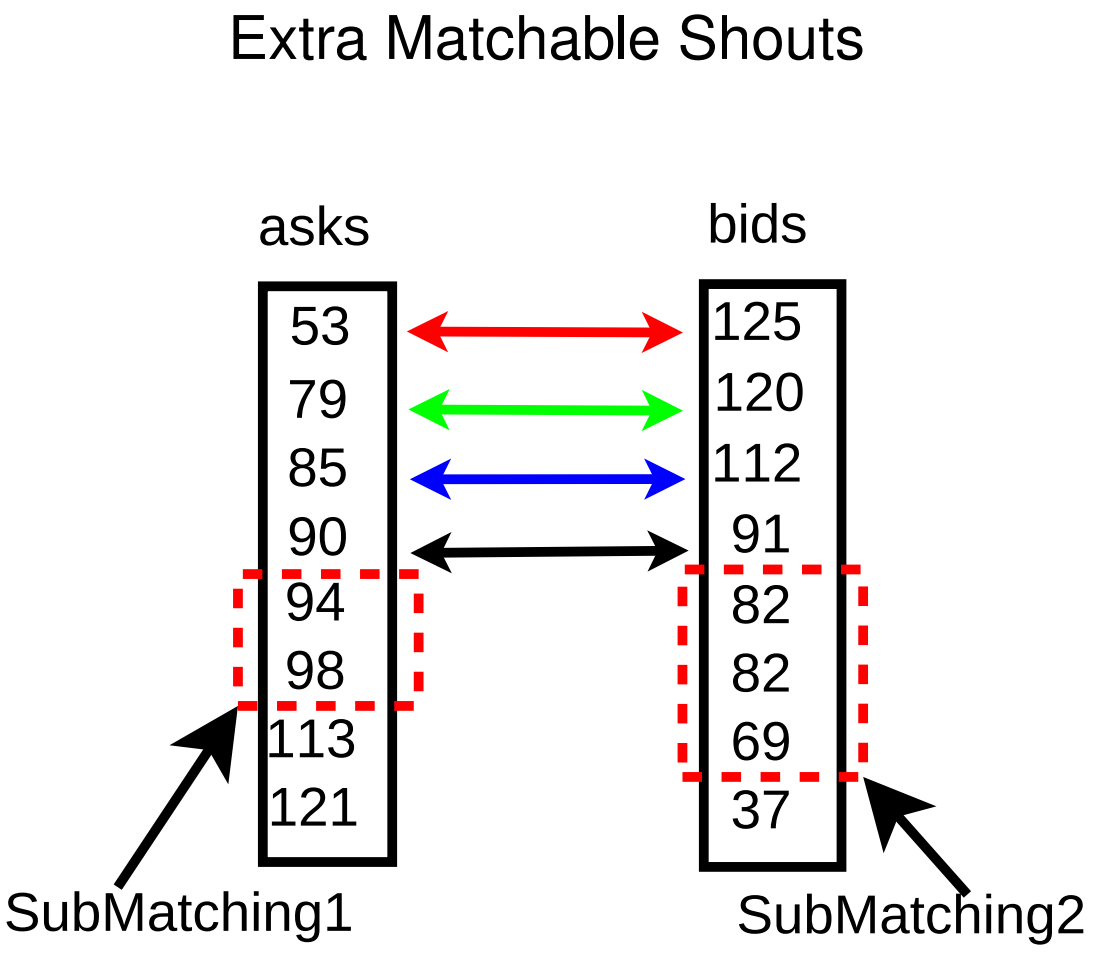


Looking Extra Matchable Shouts

Maximal Matching in Sub-matchings



Looking Extra Matchable Shouts

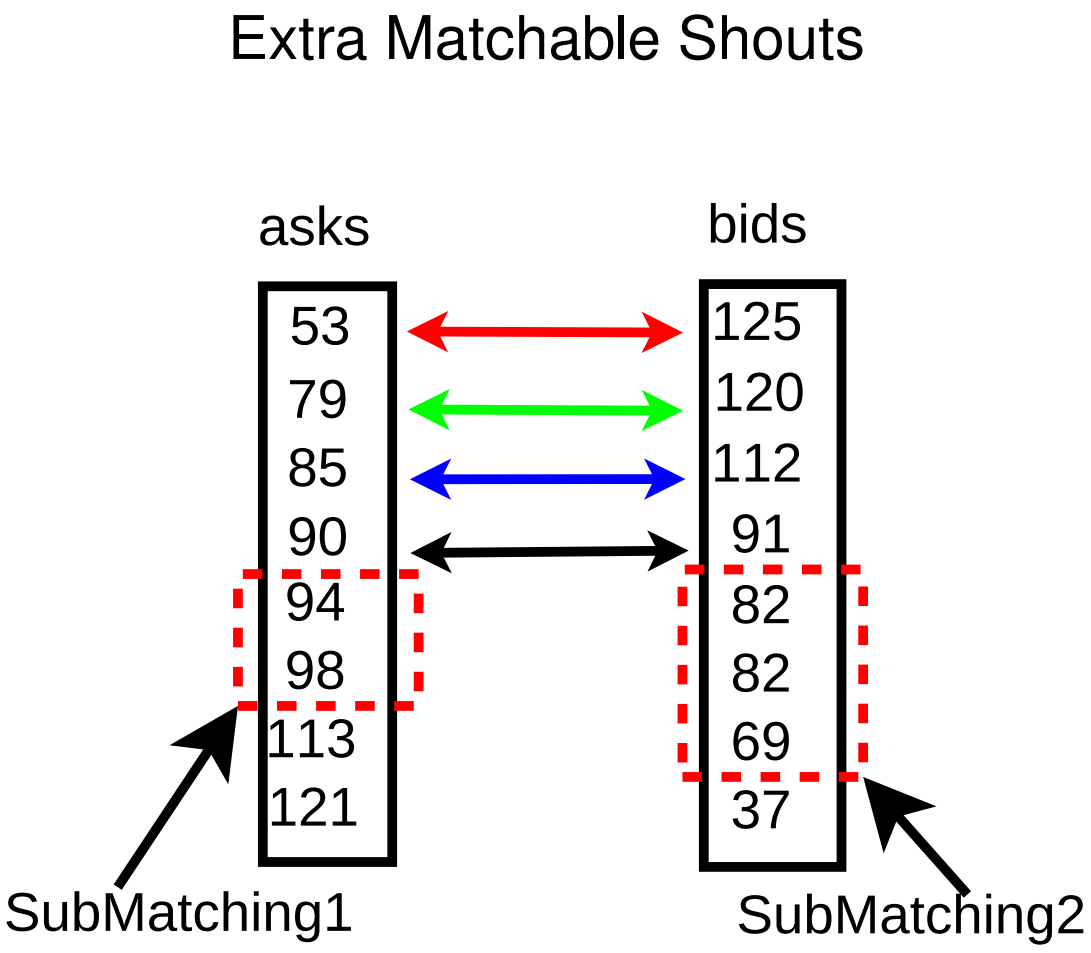


Decreasing Profit as Less as We Can

Question

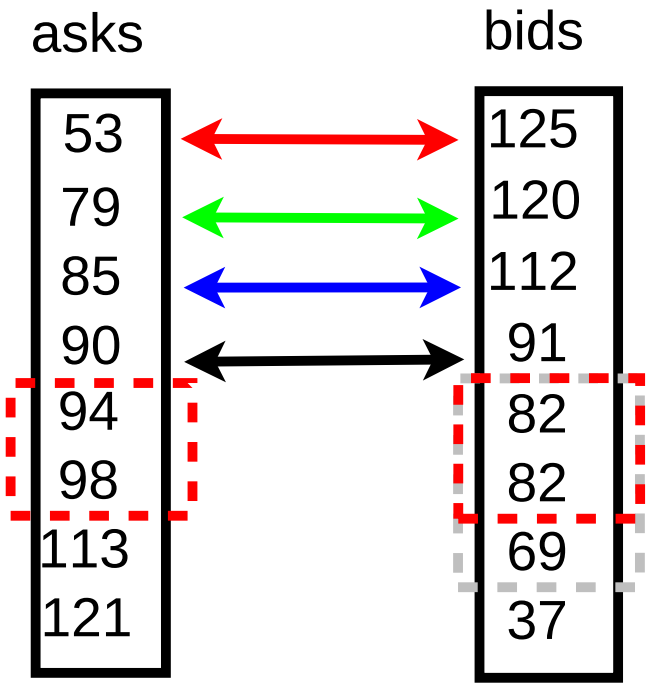
How to keep profit as much as we can?

Decreasing Profit as Less as We Can



Decreasing Profit as Less as We Can

Removing Bad Ones (Balancing)



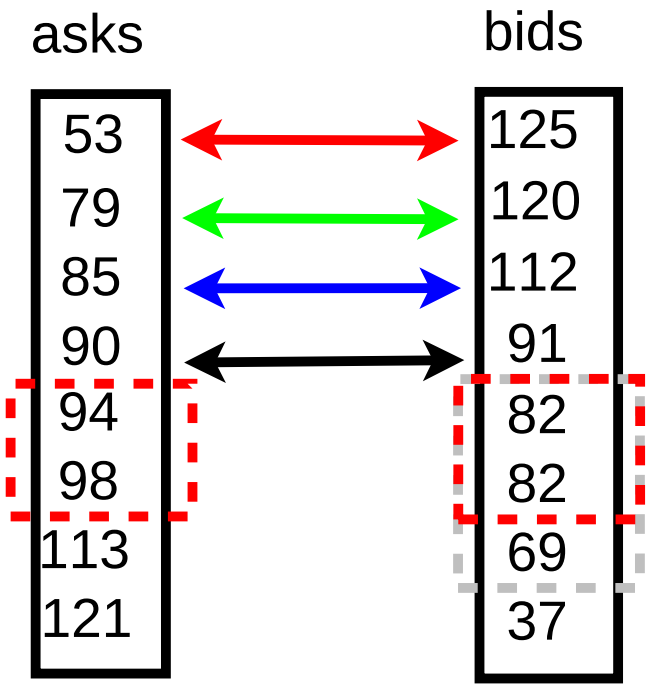
Cross-Match Matchable Shouts

Question

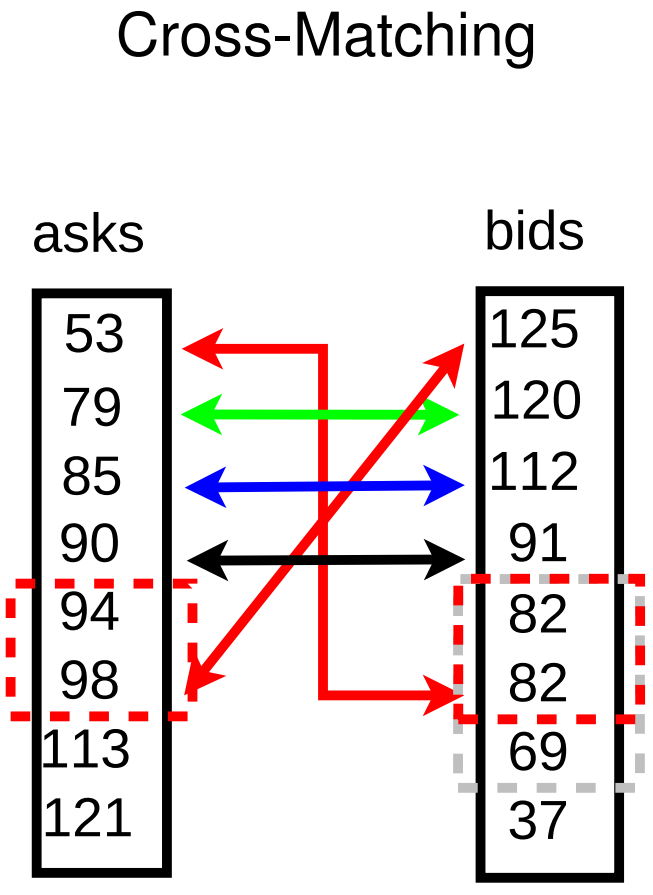
How to match extra matchable ones?

Cross-Match Matchable Shouts

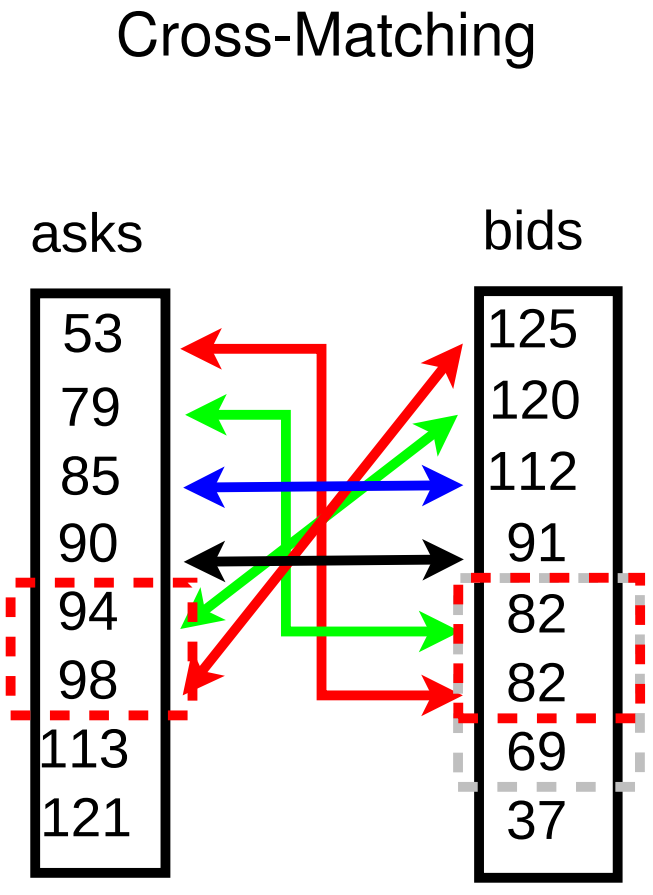
Final Extra Matchable Shouts



Cross-Match Matchable Shouts



Cross-Match Matchable Shouts



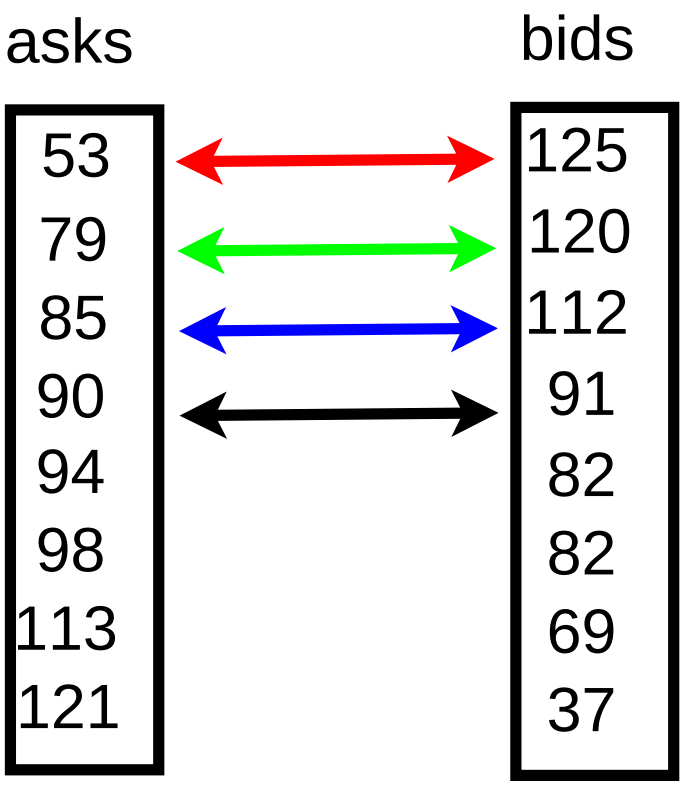
The Algorithm

Algorithm 3.1: MaximalMatching

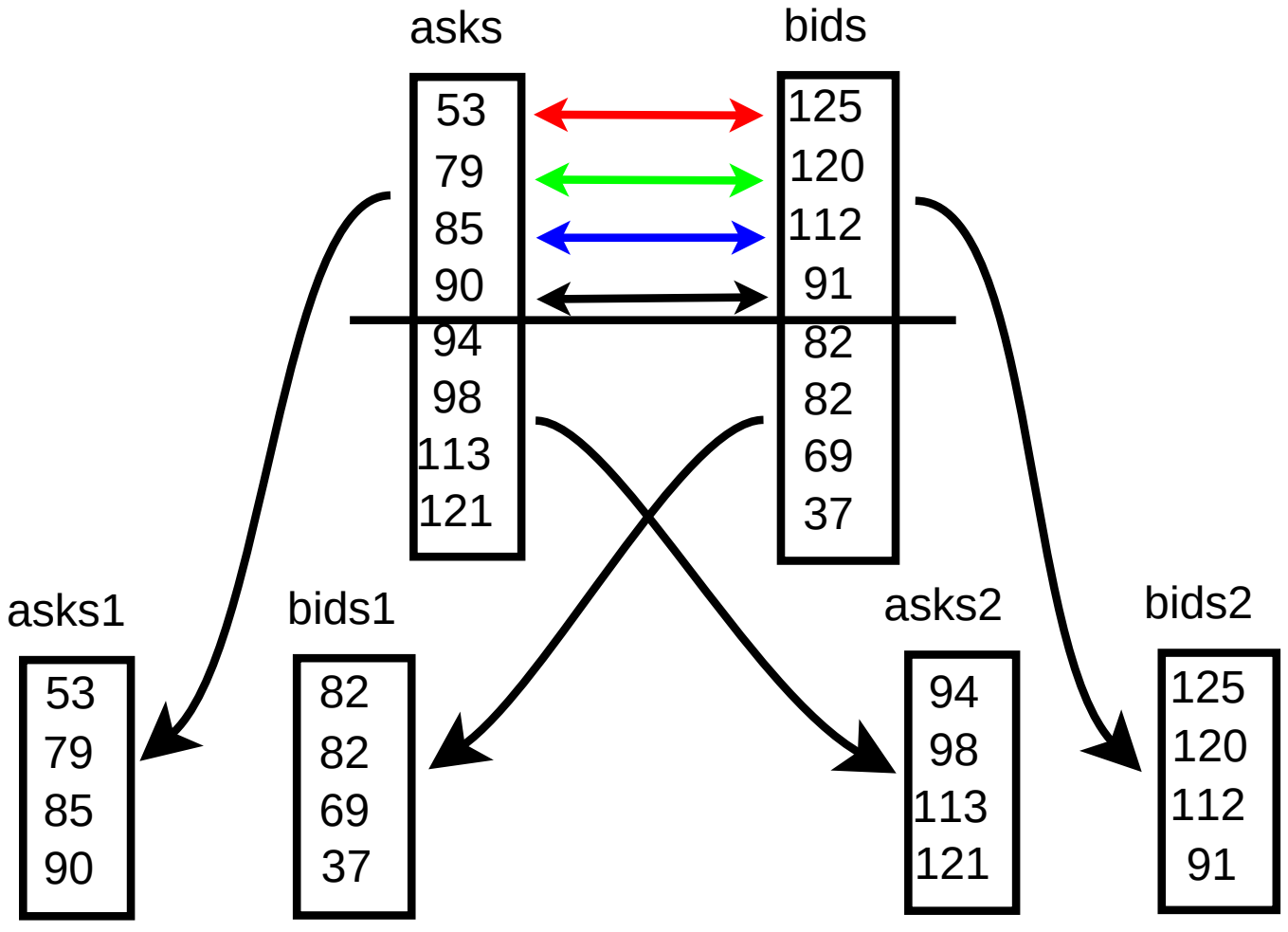
1 $Matching \leftarrow$ EquilibriumMatching($Asks$, $Bids$);
2 $MatchedAsks \leftarrow$ all asks from $Matching$ in ascending order;
3 $MatchedBids \leftarrow$ all bids from $Matching$ in descending order;
4 $MM1 \leftarrow$ MaximalMatching($MatchedAsks$, ($Bids \setminus MatchedBids$));
5 $MM2 \leftarrow$ MaximalMatching(($Asks \setminus MatchedAsks$), $MatchedBids$);
6 $NumberOfExtraMatchableMatches \leftarrow Min(|MM1|, |MM2|)$;
7 Cross-match extra matchable asks and bids;

The Algorithm

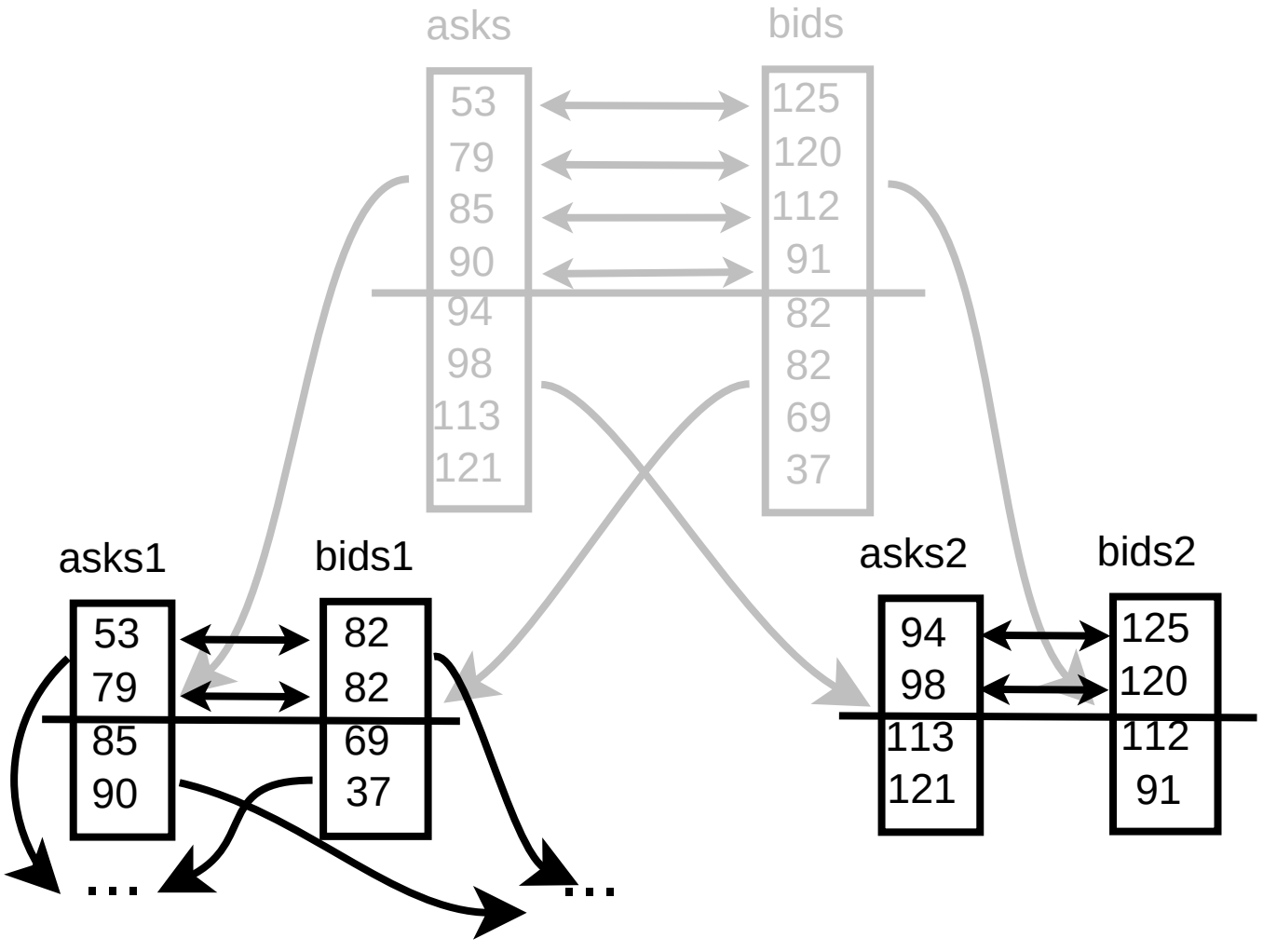
Equilibrium Matching



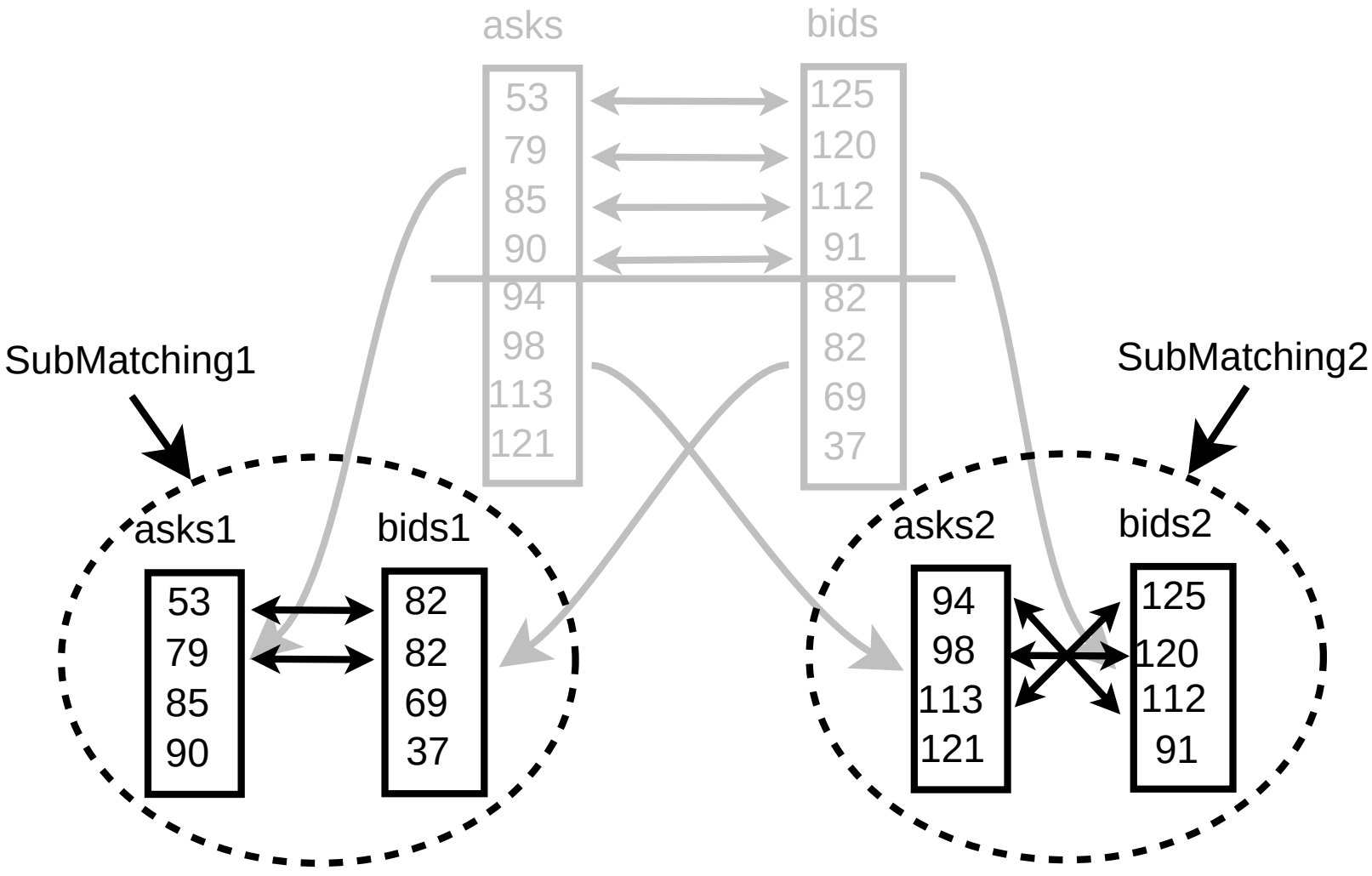
The Algorithm



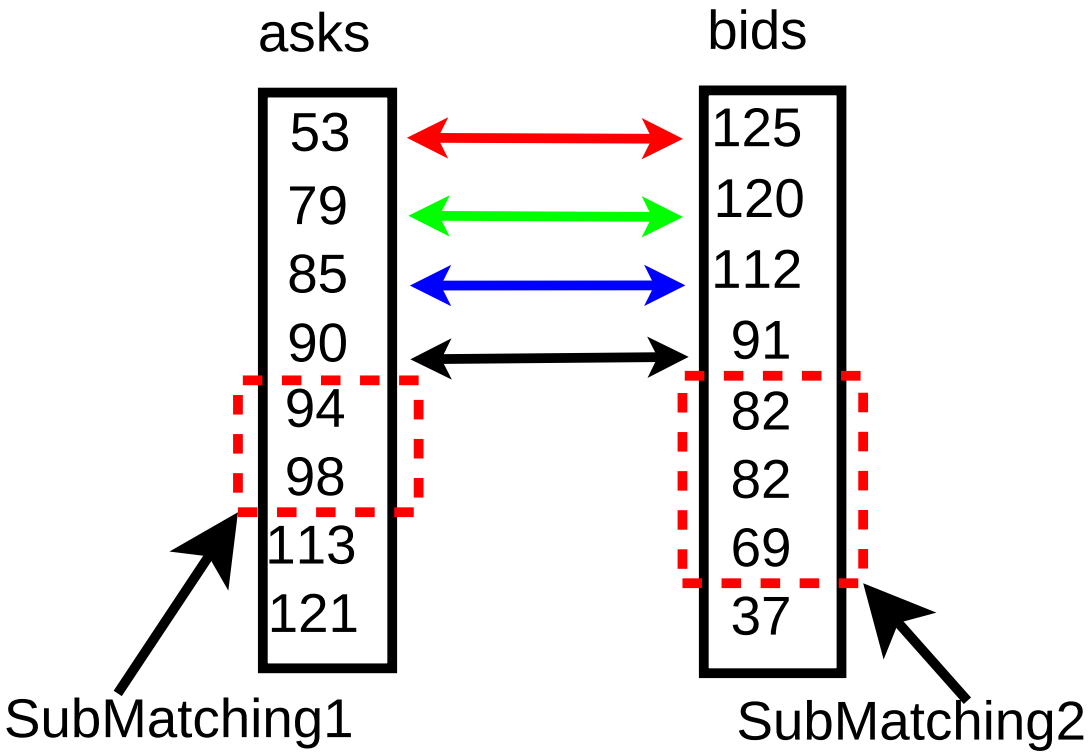
The Algorithm



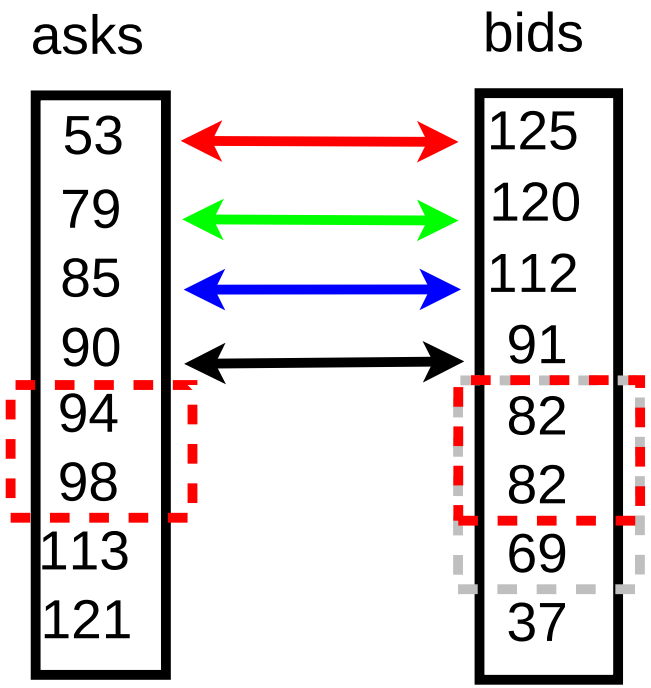
The Algorithm



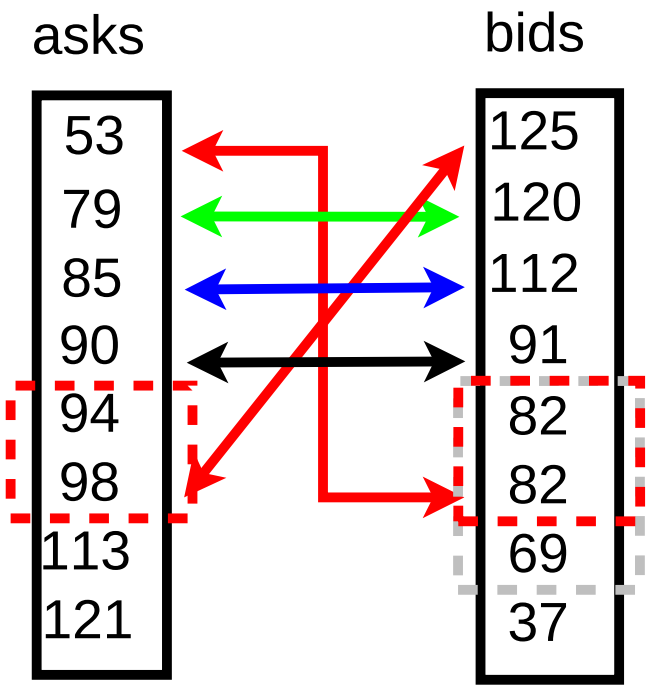
The Algorithm



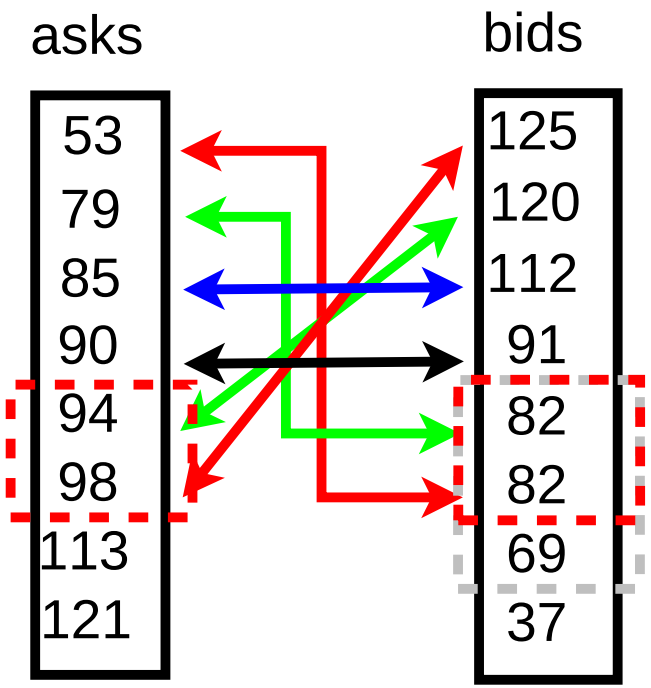
The Algorithm



The Algorithm



The Algorithm



Complexity of Maximal Matching

Worst case performance:

- $O(\max(n_a, n_b) \log \max(n_a, n_b) + \min(n_a, n_b)^2)$, where n_a (n_b) is the number of asks (bids).

The Main Result

Theorem

Maximizing Trasactions: *Given a set of shouts, the size of maximal matching is maximal.*

The Main Result

Theorem

Maximizing Trasactions: *Given a set of shouts, the size of maximal matching is maximal.*

Sketch proof.

1. Induction:
   - assume the two sub-matchings are maximal
   - then the parent matching is also maximal
2. Base:
   - no bid's price $\geqslant$ any ask's price

□

The Main Result

Theorem

Maximizing Trasactions: *Given a set of shouts, the size of maximal matching is maximal.*

Theorem

Maximizing Sell-Volume/Profit: *Given a set of shouts, maximal matching maximizes buy-volume and minimizes sell-volume. Auctioneer's profit is maximal among all matchings with the same size.*

The Main Result

Theorem

Maximizing Trasactions: *Given a set of shouts, the size of maximal matching is maximal.*

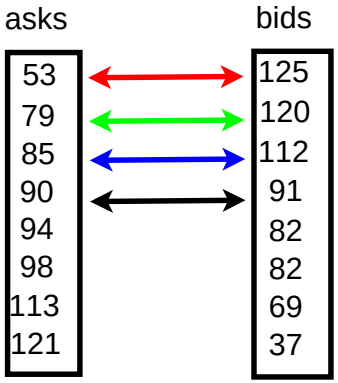
Theorem

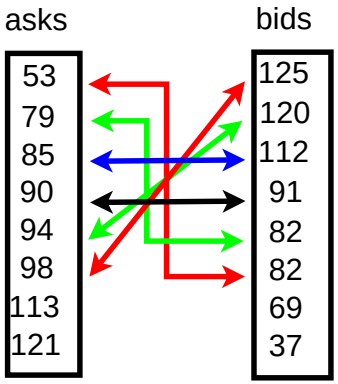
Maximizing Sell-Volume/Profit: *Given a set of shouts, maximal matching maximizes buy-volume and minimizes sell-volume. Auctioneer's profit is maximal among all matchings with the same size.*

Proof.

Because of the descending (ascending) order of bids (asks), and maximal matching always matches the first n biggest (smallest) bids (asks), where n is the size of the matching. □

Equilibrium Matching vs Maximal Matching





Equilibrium Matching

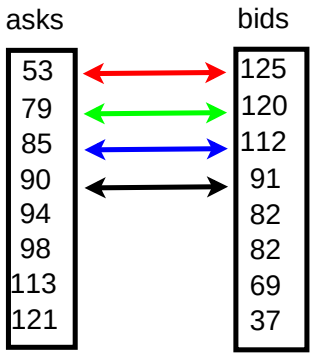
- profit maximizing (141)
- market liquidity can be improved
  1. transactions: 4
  2. buy/sell-volume: 448/307

Maximal Matching

- profit maximizing (conditional) (113)
- market liquidity maximizing
  1. transactions: 6
  2. buy/sell-volume: 612/499

Maximal Matching is Really Nice?

From economic point of view:

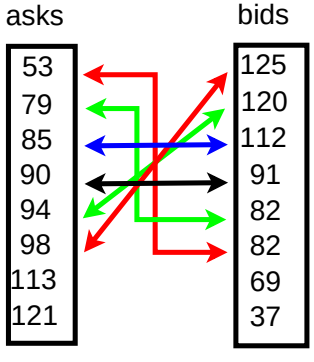


Equilibrium Matching

1. either *incentive compatible*
2. or *efficient*
3. profit maximizing

Maximal Matching

1. not *incentive compatible*
2. not *efficient*
3. less profit

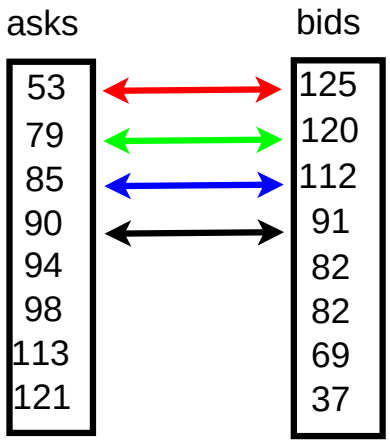


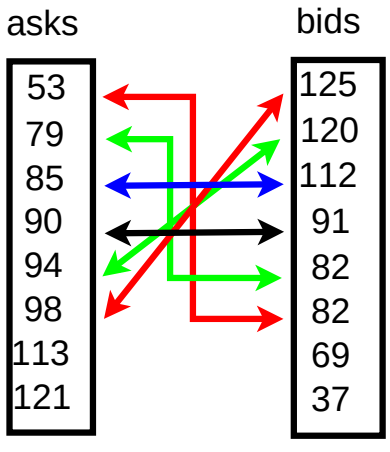
Empirical Findings in the Long Term

Question

Can a stock market owner improve market liquidity to get more traders and more profit?

Empirical Findings in the Long Term





Equilibrium Matching

1. less traders
2. less profit

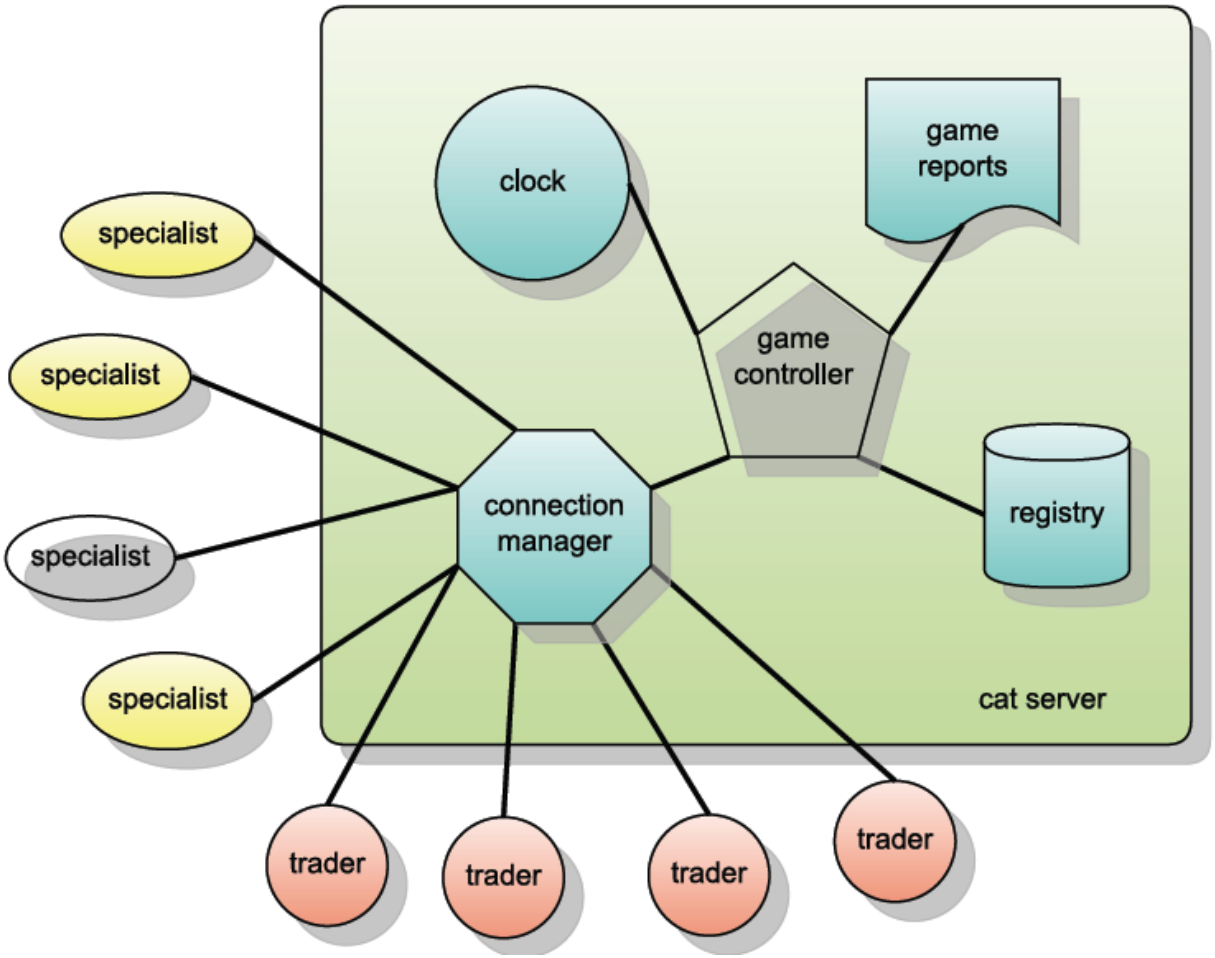
Maximal Matching

1. more traders
2. more profit

Outline

Test Platform

Trading-Agent Competition Market Design (CAT)



Test Settings

Markets:

1. *EM*: with equilibrium matching
2. *MM*: with maximal matching

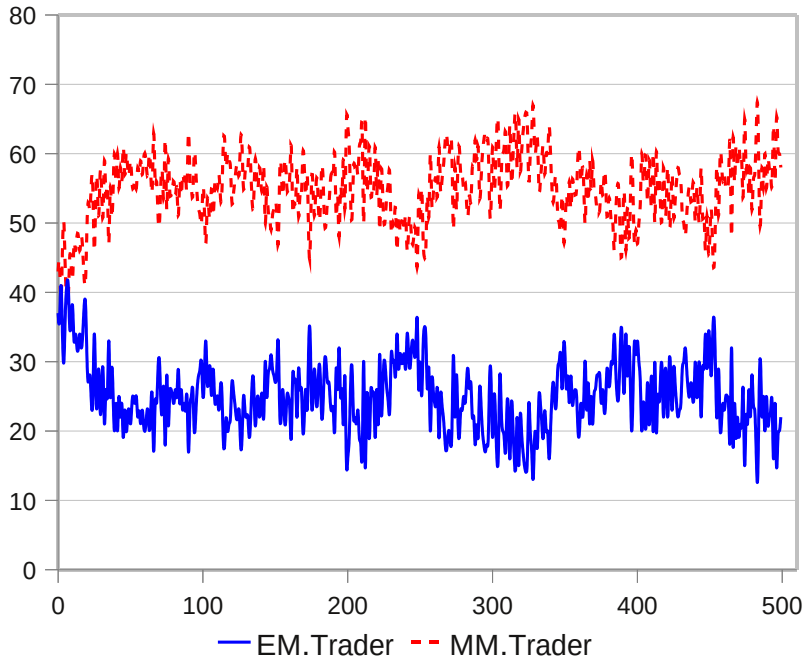
Traders:

- 80 (40 sellers and 40 buyers) with profit seeking strategies
- they can only submit offers to sell or buy one goods
- not allowed to have more than one offer at the same time

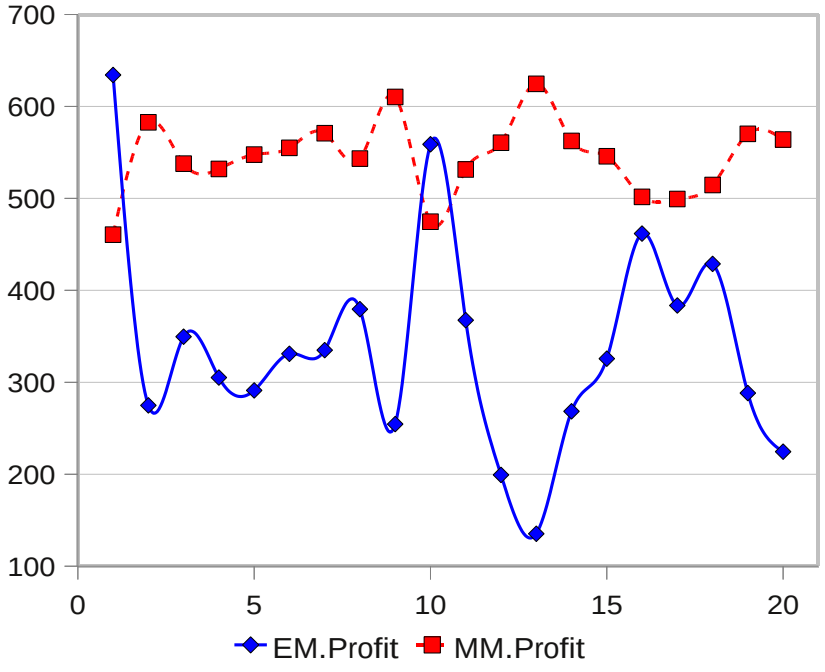
Others:

- 500 virtual days, 10 rounds in each day
- each trader only chooses one market to trade in each day
- each trader has a fixed number of goods, say 3, to trade in each day

Experimental Results



Trader Distribution



Auctioneer's Profit (avg/25ds)

Our CAT Specialist: *jackaroo*

jackaroo achievements (leader: Dongmo Zhang, UWS):

- CAT 2007: 4th
- CAT 2008: 3rd
- CAT 2009: Champion
- CAT 2010: 2nd

Outline

Summary

Matching for Double Auction

- Equilibrium Matching
- Maximal Matching
  - maximizes market liquidity
  - maximizes profit (conditional)
  - empirical findings
    - attracting traders
    - increasing profit

Further research directions:

- Online Double Auction, i.e. adding temporal information,
  - e.g. a sell offer would look like...
    "I want to sell a house only between Jan 2011 and March 2011."

Acknowledgments

- **People**: Dongmo Zhang, Laurent Perrussel, Md Khan, *jackaroo* team, and anonymous reviewers.
- **Funding**: the Australian Research Council Discovery Project DP0988750.

Acknowledgments

- **People**: Dongmo Zhang, Laurent Perrussel, Md Khan, *jackaroo* team, and anonymous reviewers.
- **Funding**: the Australian Research Council Discovery Project DP0988750.

Thank you for your attention!

Outline

Complexities

Worst case performance:

- Maximal Matching:
  - $O(\max(n_a, n_b) \log \max(n_a, n_b) + \min(n_a, n_b)^2)$, where n_a (n_b) is the number of asks (bids).
- the Hopcroft-Karp algorithm:
  - $O(|E|\sqrt{n_a + n_b})$, where $|E| \geq (n_{em})^2$, n_{em} is the size of equilibrium matching in our model.

Outline

Why Online?

Mechanism Design has focused on static settings where

- no uncertainty,
- the participants are known and independent,
- (mostly) only one decision to make.

But many real environments are dynamic in the sense of that

- the number of participants is changing,
- the private information of participants is changing.

Examples

- stock exchanges.
- peer-to-peer file sharing (e.g. BitTorrent).
- allocating computational resources (e.g. CPU time) to jobs arriving over time.

Why Online?

Mechanism Design has focused on static settings where

- no uncertainty,
- the participants are known and independent,
- (mostly) only one decision to make.

But many real environments are dynamic in the sense of that

- the number of participants is changing,
- the private information of participants is changing.

Examples

- stock exchanges.
- peer-to-peer file sharing (e.g. BitTorrent).
- allocating computational resources (e.g. CPU time) to jobs arriving over time.

Why Online?

Mechanism Design has focused on static settings where

- no uncertainty,
- the participants are known and independent,
- (mostly) only one decision to make.

But many real environments are dynamic in the sense of that

- the number of participants is changing,
- the private information of participants is changing.

Examples

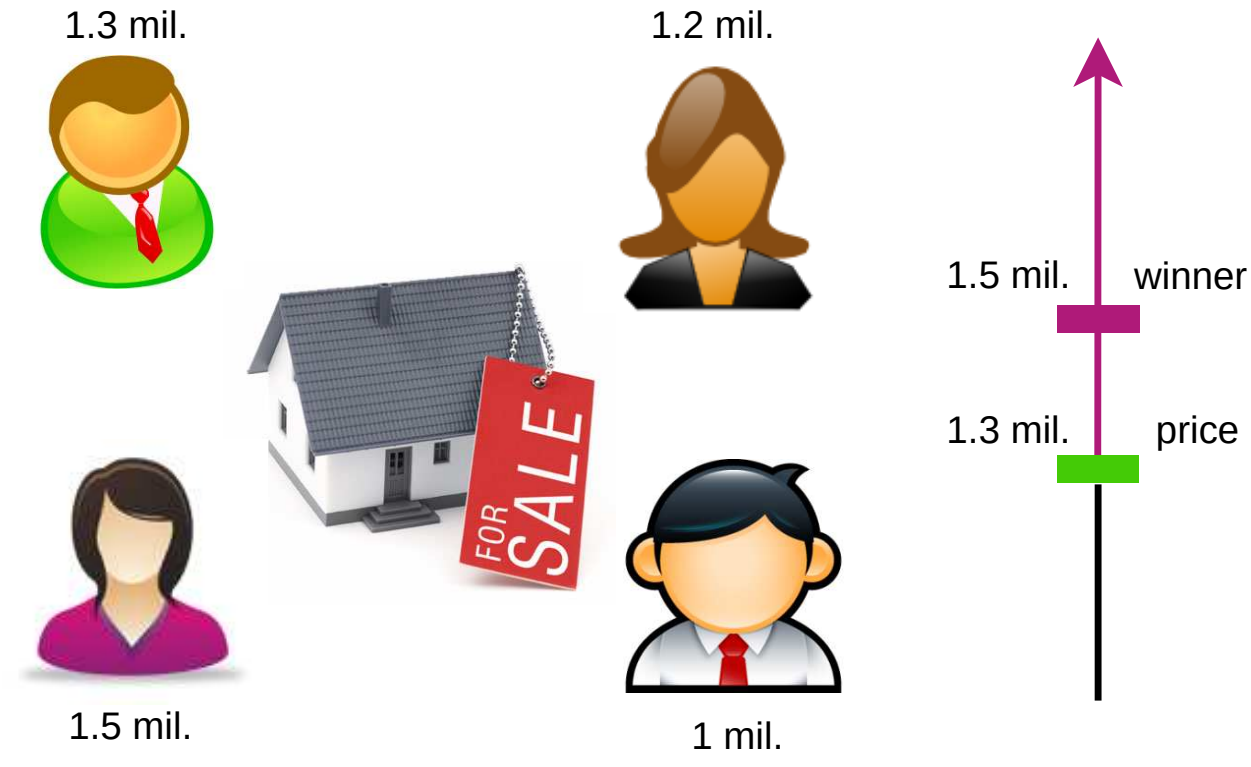
- stock exchanges.
- peer-to-peer file sharing (e.g. BitTorrent).
- allocating computational resources (e.g. CPU time) to jobs arriving over time.

Example I (Dynamic Buyers)

Online Vickrey Auction

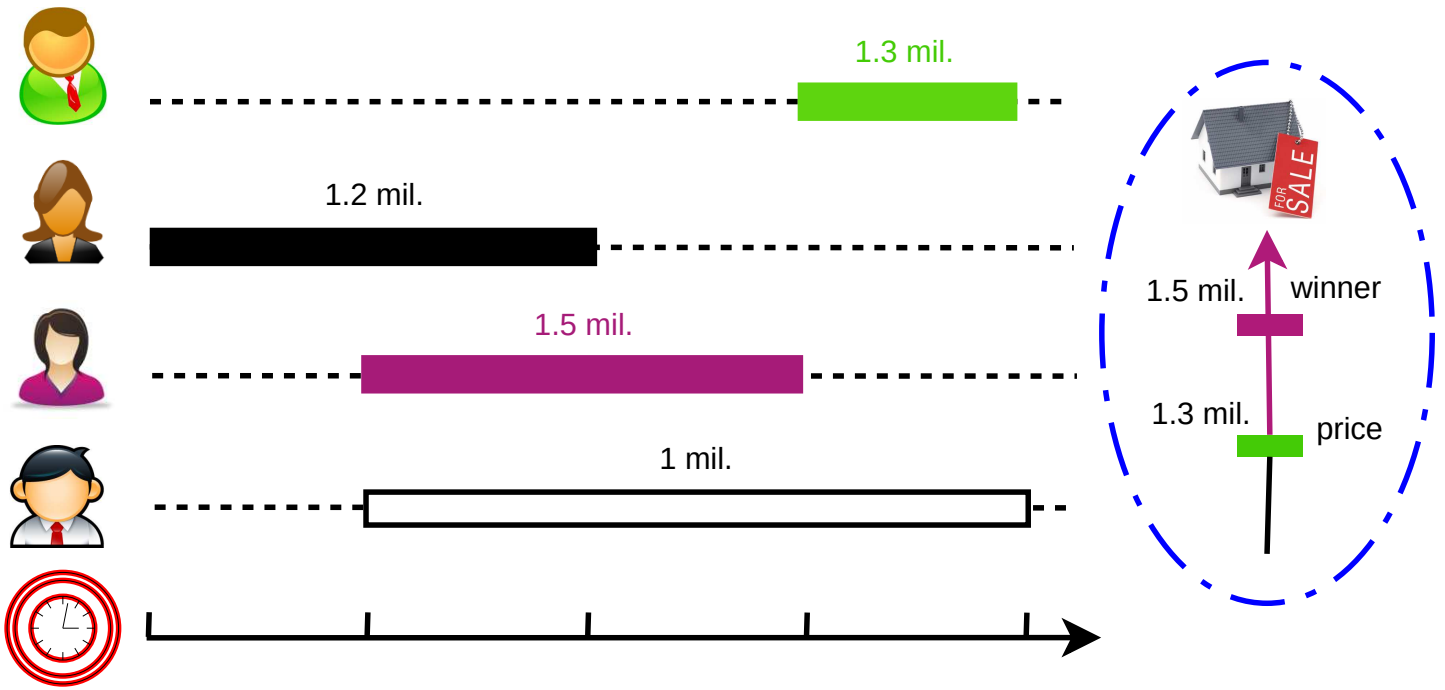
Example I (Dynamic Buyers)

Vickrey Auction (second-price sealed-bid)



Example I (Dynamic Buyers)

Online Vickrey Auction

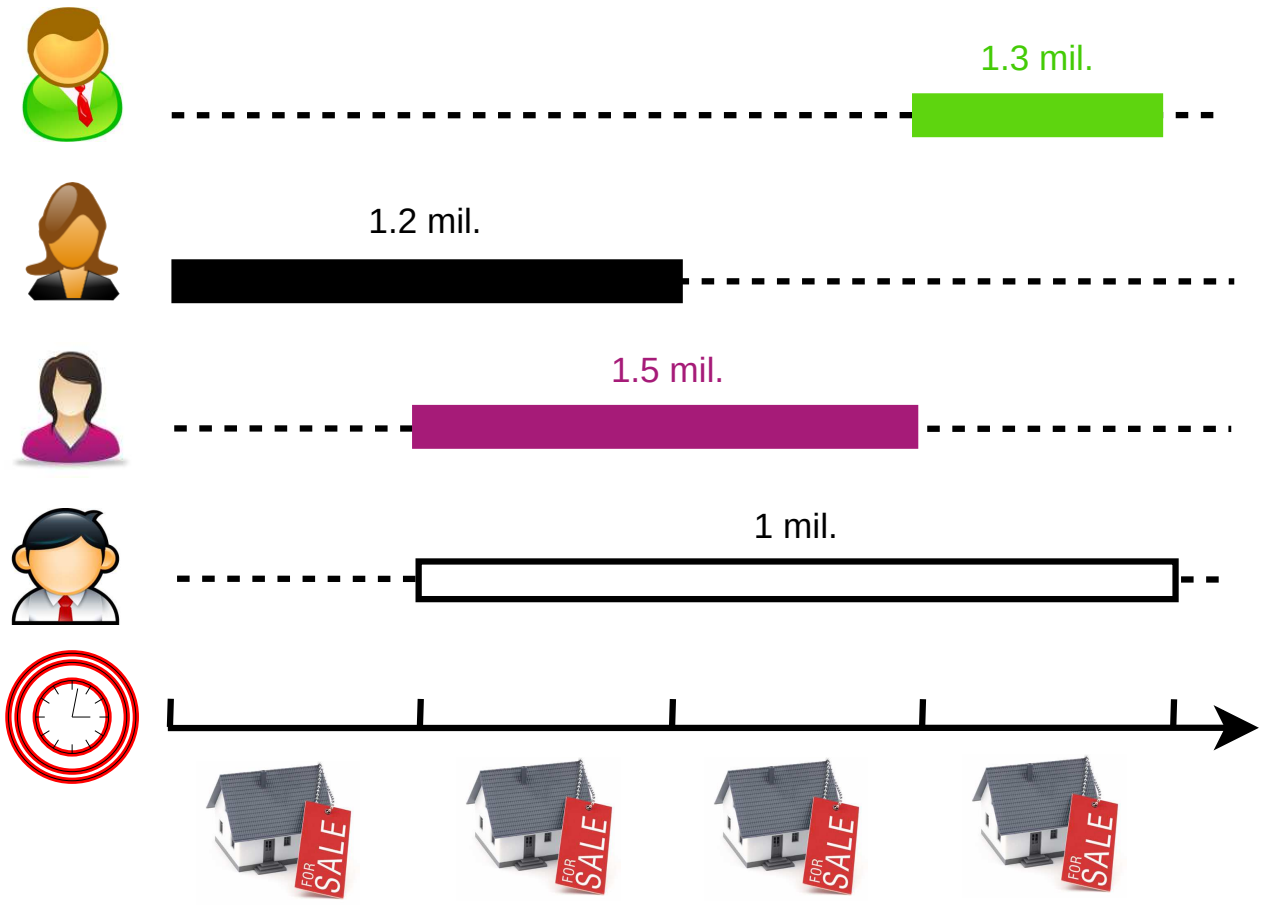


Example II (Dynamic Buyers)

Selling many identical houses (goods) in different time

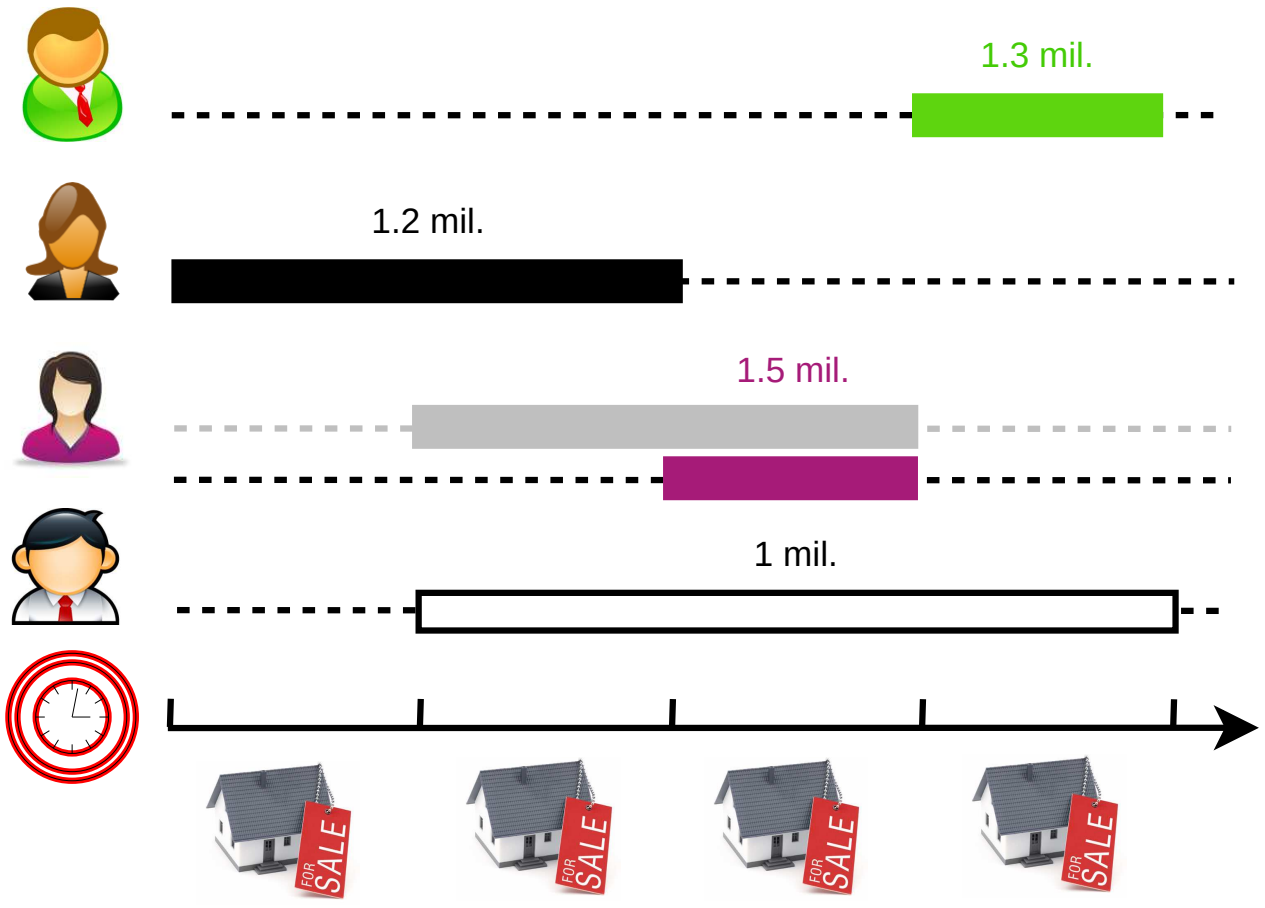
Example II (Dynamic Buyers)

Selling many identical houses (goods) in different time



Example II (Dynamic Buyers)

Selling many identical houses (goods) in different time



Example III (Dynamic Seller)

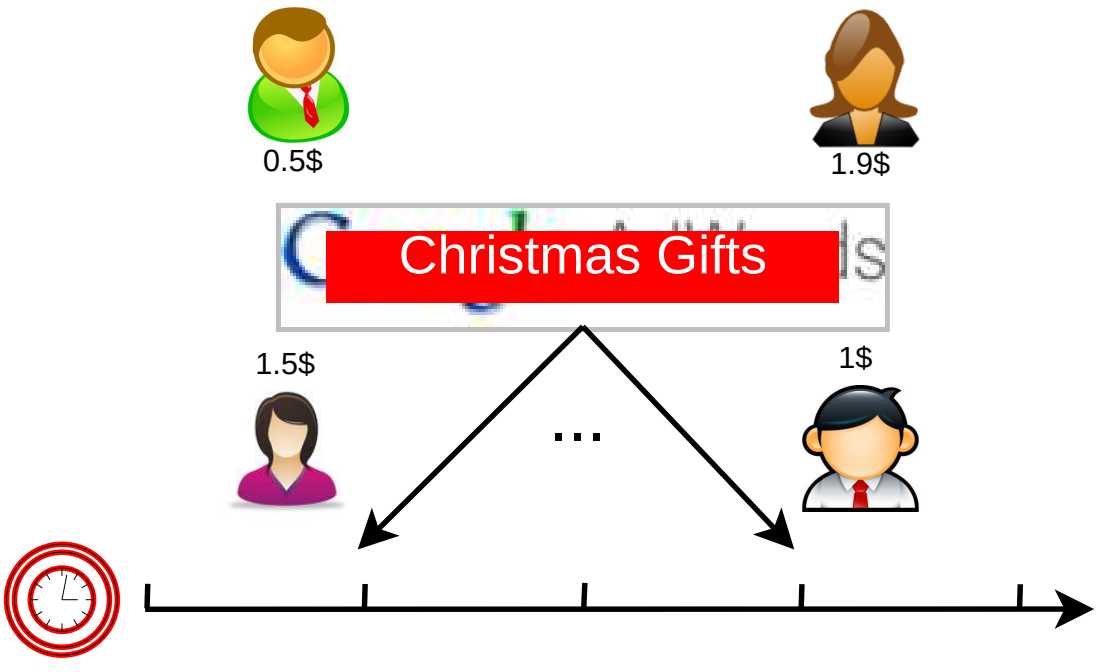
Ad Auction

Example III (Dynamic Seller)

Ad Auction: buyers bid for “Keyword”

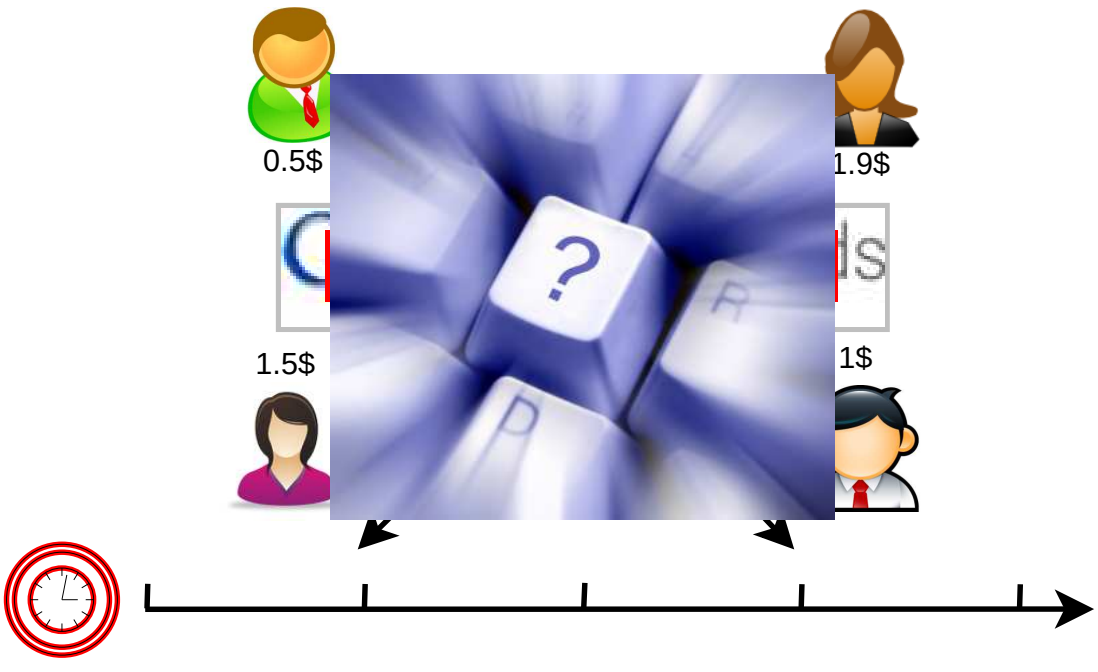
Example III (Dynamic Seller)

Ad Auction: dynamic arrival of “Keyword”



Example III (Dynamic Seller)

Ad Auction: how many “Keyword” will arrive?



Example IV (Dynamic Sellers and Buyers)

Exchanges: stock, currency, futures...

Example IV (Dynamic Sellers and Buyers)

Exchanges: stock, currency, futures...

Double Auction

Example IV (Dynamic Sellers and Buyers)

Exchanges: stock, currency, futures...

Online Double Auction

Summary

- Dynamics from buyers
  - online Vickrey auction (one goods to sell)
  - many goods to sell in a fixed schedule
- Dynamics from sellers
  - Ad auctions
- Dynamics from both
  - online double auction (exchanges)

Challenges

- dynamics provide new strategies for participants
- solutions of static mechanism design are insufficient

Static vs Online Mechanism Design

- (Static) Mechanism Design
  - well studied since 60s
  - got many nice results (e.g. Vickery auctions)
- Online Mechanism Design
  - just addressed (since 2000)
  - many real environments are dynamic, e.g. exchanges
  - new challenges (uncertainties of future)

What We Bring?

Economists:

- incentive constraint

Computer Scientists:

- computational constraint