Data-Driven Market-Making via Model-Free Learning

Yueyang Zhong
The University of Chicago Booth School of Business

Paper Link: https://www.ijcai.org/Proceedings/2020/615

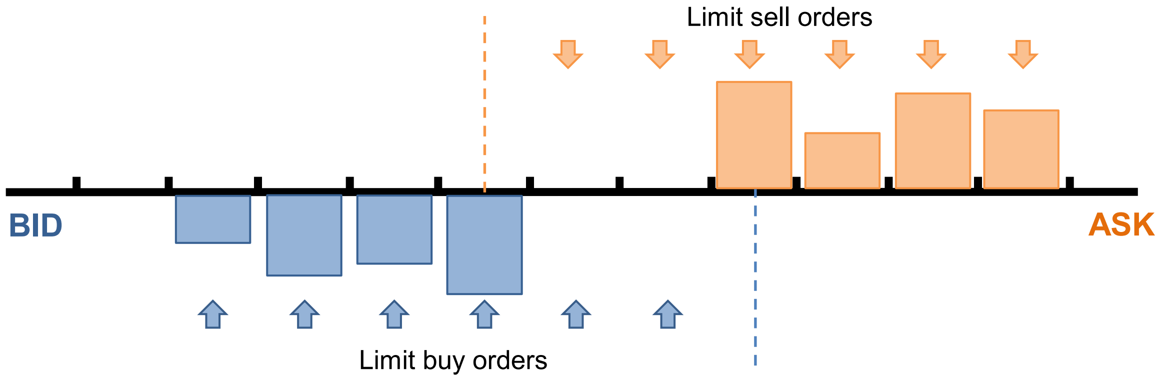
Amy R. Ward
The University of Chicago Booth School of Business

YeeMan Bergstrom
Proprietary Trading, Chicago

Jan 7-15, 2020

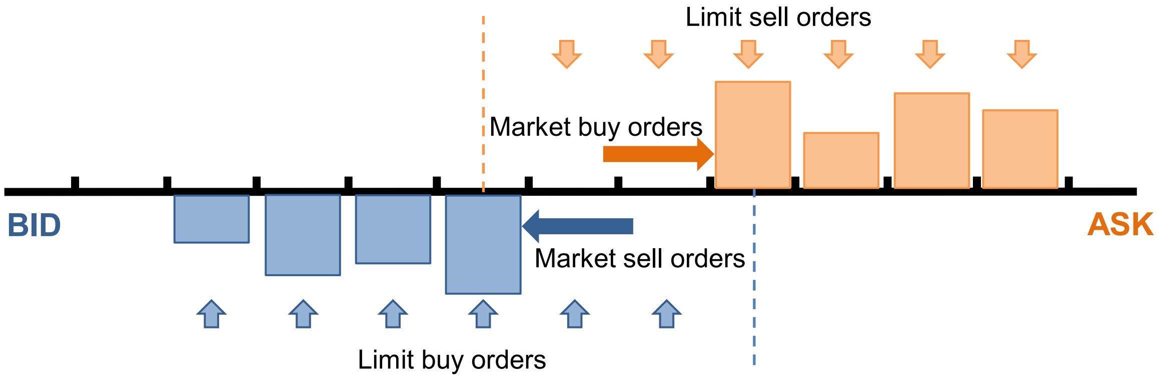
Background

- Modern U.S. equity markets: electronic exchanges (~70%)
- Limit order: buy and sell (at a specified price - bid price, ask price)
- Limit order book (LOB): a record of outstanding limit orders



Background

- Market order: buy and sell (from the best available market price to the 2^{nd} best price and so on)
- Within each price level: FCFS
- Market-making firm profit: bid-ask spread, when a limit buy order and a limit sell order get executed



Motivation

- Consider a market-making firm

- Challenge: hard to guarantee being on both sides of the trade due to **stochastic**

  (1) market order arrivals

  (2) limit order arrivals and cancellations from other participants

⇩

Unpredictable market price movements

Objective

- To provide real-time guidance for how to manage the firm's portfolio of limit buy and sell orders on the LOB so as to maximize the expected net profit with
  - limited mismatch between the amounts bought and sold;
  - sufficiently high Sharpe ratio.

    Measures the return of an investment compared to its risk
    (acceptable: >1; very good: >2; excellent: >3)

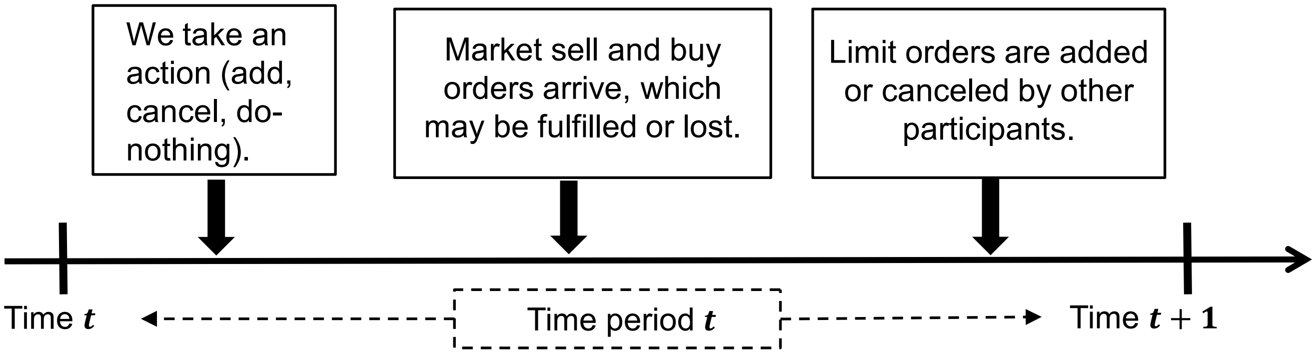
- Specifically, to provide the best action at each (discrete) decision epoch.

Outline

- Markov decision process (MDP) formulation
- Model-free Q-learning with state aggregation
- Performance evaluation using real data

Model

- A finite-horizon discrete-time MDP
- Assumption: at most one buy and one sell order can rest on the best bid price and the best ask price, respectively.
  - Convention: backtest using the simplest strategy
- Timing of LOB events

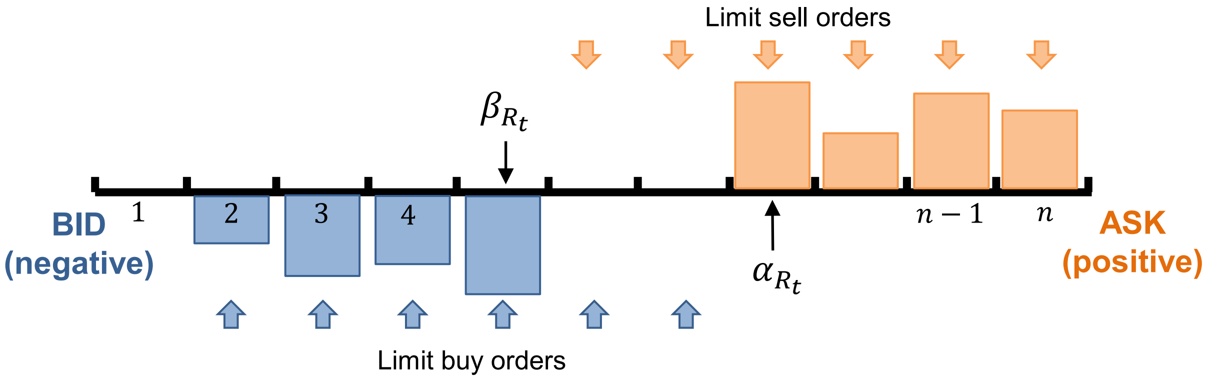


Model: State Variable

- Price levels: $\mathcal{P} := \{1,2,\ldots,n\}$
- Our limit orders: $|R^1_{tp}| \in \{0,1\}$ (conservatively assume resting at the back of the queue)

  Other participants' limit orders: $|R^2_{tp}| \in \{0,1,2,\ldots\}$

  LOB state variable: $R_t = \left(R^1_{tp}, R^2_{tp}\right)_{p\in\mathcal{P}}$
- Best bid and ask prices: β_{R_t}, α_{R_t}



Model: Decision Variable

- Allowable actions:

  Having or not having one buy (sell) order at the best bid (ask) price

  1 0

- Action space: $A_t = (A_{t1}, A_{t2}) \in \mathcal{A} := \{(0,0),(0,1),(1,0),(1,1)\}$

  Bid side Ask side

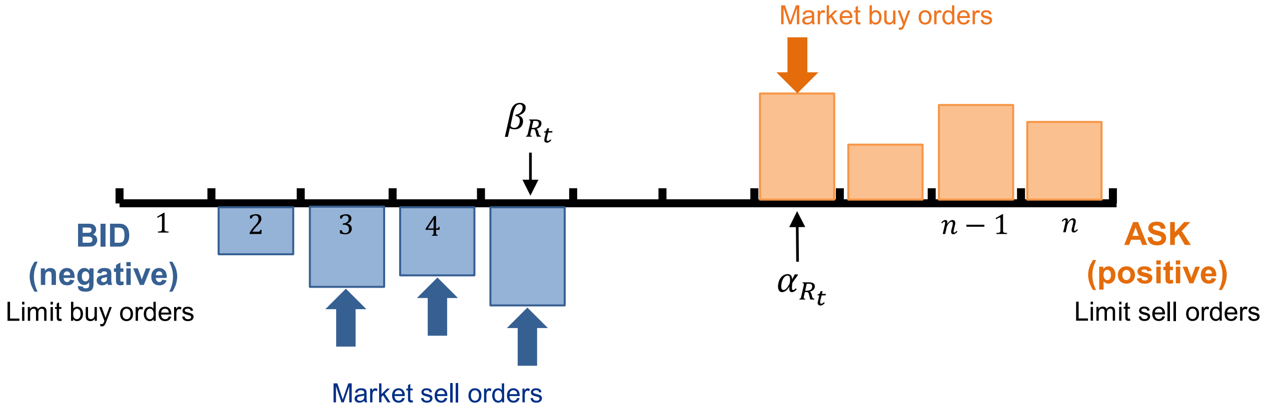
- Post-decision state: $R_{tp}^{a2} = R_{tp}^{2},\ R_{tp}^{a1} = \begin{cases} 1, & \text{if } A_{t1} = 1, p = \beta_{R_t} \text{ or } A_{t2} = 1, p = \alpha_{R_t} \\ 0, & \text{otherwise} \end{cases}.$

Model: Exogenous Information

- Market buy and sell orders: $\hat{D}_t^{MB}$, $\hat{D}_t^{MS}$ $\Longrightarrow$ $R_{tp}^{m} = \left(R_{tp}^{m1}, R_{tp}^{m2}\right)_{p\in\mathcal{P}}$

- Orders and cancellations from other participants:
  $\hat{O}_t = \left(\hat{O}_{tp}\right)_{p\in\mathcal{P}}$, $\hat{C}_t = \left(\hat{C}_{tp}\right)_{p\in\mathcal{P}}$ $\Longrightarrow$ $R_{tp}^{o} = \left(R_{tp}^{o1}, R_{tp}^{o2}\right)_{p\in\mathcal{P}}$

- Pre-decision state for the next decision epoch: $R_{t+1} = \left(R_{tp}^{o1}, R_{tp}^{o2}\right)_{p\in\mathcal{P}}$



Model: Objective

$m_{R_t} \coloneqq (\alpha_{R_t} + \beta_{R_t})/2$

- Objective function: profit and loss (PnL) relative to the mid price + penalty of mid price movement

$$V\left(R_t, A_t, \widehat{D}_t^{MB}, \widehat{D}_t^{MS}, inv_t\right) \coloneqq PnL\left(R_t, A_t, \widehat{D}_t^{MB}, \widehat{D}_t^{MS}\right) + inv_t \cdot \Delta_{m_t}$$

- Profit and loss (PnL):

Mid price

$$PnL\left(R_t, A_t, \widehat{D}_t^{MB}, \widehat{D}_t^{MS}\right) \coloneqq E^{\beta} \cdot \left(m_{R_t} - \beta_{R_t}\right) + E^{\alpha} \cdot \left(\alpha_{R_t} - m_{R_t}\right)$$

$\{0,1\}$ if one of our resting order get executed on the bid side

$\{0,1\}$ if one of our resting order get executed on the ask side

- Inventory level (open position): inv_t = cumulative amount bought – cumulative amount sold

Model: Challenges

- Challenges in solving the MDP:
  (1) difficulty in estimating the transition probabilities (sizes and arrivals cannot fit any distribution);
  (2) a very large state space.

- Example: 20 price levels, maximum queue length=1000
  LOB state space size: $1000^{20} = 1 \times 10^{60}$!!!

$\Rightarrow$ stochastic approximation method + state aggregation

Q-learning Model: Aggregation

- Q-learning algorithm only works well in small state and action spaces [Powell, 2007]

- State aggregation [Pepyne et al., 1996]

- Five attributes

From data
(1) bidSpeed: $BS \in \{0,1\}$, if the market sell orders exceed the book size at the best bid price;
(2) askSpeed: $AS \in \{0,1\}$, if the market buy orders exceed the book size at the best ask price;
(3) avgmidChangeFrac: $MF \in \{0, \pm 1, \pm 2\}$, the relative change in the average mid price $\boldsymbol{f} \in [\mathbf{0}, \mathbf{1}]$

History-dependent
(4) invSign: $IS \in \{0, \pm 1, \pm 2\}$, the side and magnitude of inv_t $\boldsymbol{I} \in [\mathbf{0}, \infty)$
(5) cumPnL: $cumPnL \in \{0,1\}$, if the cumulative PnL is large or small $\boldsymbol{P} \in (-\infty, \infty)$

- State aggregation function:

$$G(R_t, inv_t, pnl_t) \coloneqq (BS_t, AS_t, MF_t, IS_t, CP_t)$$

Aggregated state space size: $2 \times 2 \times 5 \times 5 \times 2 = 200$ ☺

Q-learning Model: Algorithm

For each iteration $n \in [\overline{N}]$:

Randomly select some "aggregated state-action" pairs to update;

For each selected "aggregated state-action" pair (s, a):

1. Randomly select a sample path ω with initial aggregated state s, and associated full state by R_n^s
2. Update the full state and aggregate to $\overline{s} = G(R_{n+1}^s, inv_{n+1}^s, pnl_{n+1}^s)$;
3. Update the Q factor:

   $Q_{n+1}(s,a) = \left(1 - \alpha_n(s,a)\right) \cdot Q_n(s,a) + \alpha_n(s,a) \cdot \left(V(\omega) + \gamma \max_{v \in \mathcal{A}_s} Q_n(\overline{s}, v)\right)$, where $V(\omega)$ is the "PnL + penalty term" obtained from sample path ω, $\alpha_n(s,a) := \frac{\alpha_0}{\#\, updates\ of\ (s,a)}$ is the learning rate.

- For any full state R_t with inventory and PnL at inv and pnl, the optimal action is:

$$\operatorname*{argmax}_a Q_{\overline{N}}\left(G(R_t, inv, pnl), a\right)$$

Dataset

- Product: an asset traded on the Chicago Mercantile Exchange (CME)
- Data: event-by-event (i.e., add, cancel, execution) tick data (including quantity and price level) from 9:00 a.m. – 14:30 p.m. (microsecond precision) in 2019

Performance Evaluation

- Backtest in our partner firm:

In-sample	Out-of-sample
June 2019	July 2019
Train	Backtest

- In-sample experiments: set and fix algorithm parameters
  (1) Thresholds $f = 0.5, I = 20, P = 450$

- Out-of sample test: using the fixed algorithm parameters

Resulting Q Table

- We trained six Q tables for each hour from 9:00 a.m. – 14:30 p.m. (9:00-10:00, 10:00-11:00, 11:00-12:00, 12:00-13:00, 13:00-14:00, 14:00-14:30)

Aggregated book state					Suggested action	
bidSpeed	askSpeed	avgmidChangeFrac	avgSign	CumPnL	Action_bid	Action_ask
0	0	-2	-2	0	0	0
0	0	-2	-2	1	1	0
0	0	-2	-1	0	0	1
0	0	-2	-1	1	0	1
...	...	...	...	...	...	...

Resulting Q Factors

The trading policies learned from the resulting Q table:

- It is profitable to place limit orders on the more active side; e.g., add limit buy orders on a sell-heavy market;
- Market-making is not directional [Menkveld, 2013];
- The optimal strategy keeps inventory near zero [Guilbaud and Pham, 2013];
- The optimal strategy cancels all orders when cumulative PnL is low.

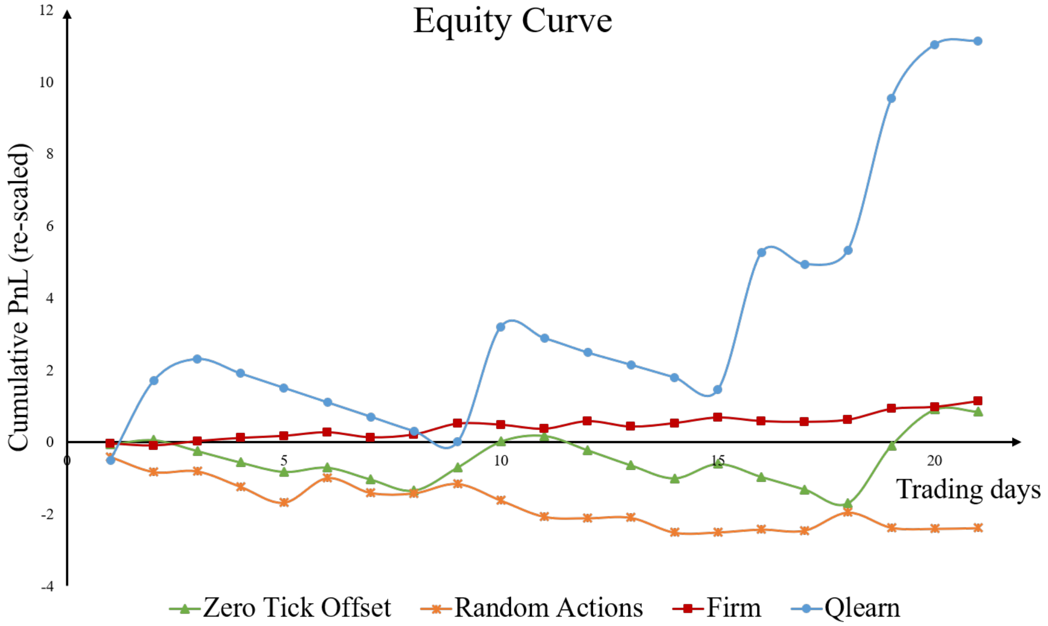
Performance Evaluation

- Common benchmarks [Spooner et al., 2018; Lim and Gorse, 2018; Doloc, 2019]

  (1) Fixed spread-based strategy: having limit orders at the best bid and ask prices at all times;

  (2) Random strategy: having limit orders at the best bid and ask prices by flipping an unbiased coin;

  (3) Partner firm's implemented trading strategy.



The out-of-sample performance: an average daily PnL over 1000, and a Sharpe ratio above 3.

Future Directions

- Smooth out the resulting equity curve: avoid losing money in a row of days
- Develop an algorithmic approach to decide when to close down trading, resulting in the length of the finite time horizon being a random variable
- …

Paper Link: https://www.ijcai.org/Proceedings/2020/615

THANK YOU

Paper Link: https://www.ijcai.org/Proceedings/2020/615