Online Learning in Limit Order Book Trade Execution

Nima Akbarzadeh, *Student Member, IEEE*, Cem Tekin, *Member, IEEE*,
and Mihaela van der Schaar, *Fellow, IEEE*

***Abstract*—In this paper, we propose an online learning algorithm for optimal execution in the limit order book of a financial asset. Given a certain number of shares to sell and an allocated time window to complete the transaction, the proposed algorithm dynamically learns the optimal number of shares to sell via market orders at prespecified time slots within the allocated time interval. We model this problem as a Markov Decision Process (MDP), which is then solved by dynamic programming. First, we prove that the optimal policy has a specific form, which requires either selling no shares or the maximum allowed amount of shares at each time slot. Then, we consider the learning problem, in which the state transition probabilities are unknown and need to be learned on the fly. We propose a learning algorithm that exploits the form of the optimal policy when choosing the amount to trade. Interestingly, this algorithm achieves bounded regret with respect to the optimal policy computed based on the complete knowledge of the market dynamics. Our numerical results on several finance datasets show that the proposed algorithm performs significantly better than the traditional Q-learning algorithm by exploiting the structure of the problem.**

***Index Terms*—Limit order book, Markov decision process, online learning, dynamic programming, bounded regret.**

I. Introduction

OPTIMAL execution of trades is a problem of key importance for any investment activity [2]–[8]. Once the decision has been made to sell a certain number of shares the challenge often lies in how to optimally place this order in the market. In simple terms, we can formulate the objective as selling (buying) at the highest (lowest) price possible. Not only do we want to leave as little a foot-print in the market as possible, but also to sell (buy) at a price favorable to the order in question, while ensuring the trade actually gets fulfilled.

More formally we define the goal as to sell[1] a specific number of shares of a given stock during a fixed time period (round) in a way that maximizes the revenue, or equivalently, minimizes the accumulated cost of the trade. This problem is also called the optimal liquidation problem, and is performed over the limit order book (LOB) mechanism. In the LOB, the traders can specify the volume and a limit on the price of shares that they desire to sell/buy. The selling side is called the ask side and the buying side is called the bid side. An order in which both volume and price is defined is called a *limit order*. The limit orders may get executed after a while or get canceled by a *cancellation order* from the trader who submitted it. The order on one side of the LOB is executed only if the LOB can match the order with a previously submitted or a newly arrived order on the other side of the LOB. Another type of order is the *market order* where the trader only defines the volume, and then, the order is executed against the best available offers on the other side of the LOB. An LOB also lists the total selling and buying amounts and prices on bid and ask sizes, respectively. A detailed discussion of the LOB mechanism can be found in [9].

The optimal liquidation problem in the LOB is considered in numerous prior works. Among these, [7] and [10] solve this problem using static optimization approaches or dynamic programming, while several other works tackle this problem using a reinforcement learning approach. Reinforcement learning based methods consider various definitions of state, such as the remaining inventory, elapsed time, current spread, signed volume, etc. Actions are defined either as the volume to trade with a market order or as a limit order [8], [11], [12]. A hybrid method is proposed in [8]: firstly, an optimization problem is solved to define an upper bound on the volume to be traded in each time slot, using the Almgren Chriss (AC) model proposed in [7]. Then, a reinforcement learning approach is used to find the best action, i.e., the volume to trade with a market order, which is upper bounded by a relative value obtained in the optimization problem. Another prior work [12] implements the same approach with different action and state sets. In all of the above works, the authors used Q-learning to find the optimal action for a given state of the system. In [8] and [12] the learning problem is separated into training and test phases, where the Q-values are only updated in the training phase, and then, these Q-values are used in the test phase.

Unlike prior approaches, we use a model-based approach by considering the problem as an MDP, in which we develop a new market model, and then, learn the state transition dynamics of the model in an online manner through real-time execution of market orders. Specifically, we propose a new market state space

[1] This problem can generalized to buying problem as well.

Manuscript received December 16, 2017; revised May 15, 2018; accepted June 27, 2018. Date of publication July 20, 2018; date of current version August 2, 2018. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mark A. Davenport. The work of M. van der Schaar is supported by the National Science Foundation under NSF Award 1524417 and NSF Award 1462245. This work was presented in part at the Fifth IEEE Global Conference on Signal and Information Processing, Montreal, Quebec, November 2017. *(Corresponding author: Nima Akbarzadeh.)*

N. Akbarzadeh is with the Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada, and also with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey (e-mail: nima.akbarzadeh@mail.mcgill.ca).

C. Tekin is with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey (e-mail: cemtekin@ee.bilkent.edu.tr).

M. van der Schaar is with the Oxford Man Institute of Quantitative Finance, Oxford OX2 6ED, U.K. (e-mail: mihaela.vanderschaar@oxford-man.ox.ac.uk).

Digital Object Identifier 10.1109/TSP.2018.2858188

model, which can be decomposed into private and market variables. The private variable is the inventory level of the available shares to be sold during the remaining time slots in a round. The market variable is defined as the difference between the current bid price and the bid price at the beginning of the current round scaled down by volatility. Similar to [8], the action in a time slot is defined as the number of shares to be sold with a market order.

For this problem, we first deduce the form of the optimal policy using the mentioned decomposition of the state variables and dynamic programming. Essentially, we prove that in each time slot the optimal policy chooses an action from a candidate action set that contains only two actions. This result allows us to learn the optimal policy using the reduced action set, which speeds up both computation and learning. Then, we propose a learning algorithm, named *Greedy exploitation in Limit Order Book Execution* (GLOBE), that uses the estimated state transition probabilities and the form of the optimal policy to place orders at each round. To characterize how well GLOBE learns, we define the notion of regret, which measures the excess cost incurred by GLOBE compared to an oracle, which knows the true problem parameters and statistics of the order book, and computes the optimal policy at each round based on the market dynamics. Then, we show that the regret of GLOBE is bounded, which implies that GLOBE learns the optimal policy only after finitely many rounds. This is different from the results of prior works in online reinforcement learning, where the regret is shown to be $\mathcal{O}(\log T)$ [13]–[15]. This difference stems from the fact that GLOBE is able to learn the long-term impact of each action without the need for selecting that action due to the specific decomposition of the state space. Finally, we show the superiority of the proposed algorithm and its modifications over several variants of Q-learning based algorithms that exist in the literature.

The contributions of this paper can be summarized as follows:

- We propose a new model for LOB trade execution with private and market states, and show that the optimal policy has a special structure that allows efficient learning.
- We propose a new online learning algorithm called GLOBE that greedily exploits the estimated optimal policy in each round. Unlike other online reinforcement learning approaches [13]–[15], this algorithm does not require explorations to learn the state transition probabilities, and hence, its regret is bounded.
- We show that GLOBE provides significant performance improvement over other state-of-the-art learning algorithms designed for LOB in numerous finance datasets.

The rest of the paper is organized as follows. Related work is covered in Section II. The problem formulation is given in Section III. The form of the optimal policy is obtained in Section IV. GLOBE is introduced in Section V, and its regret analysis is carried out in Section VI. Section VII contains numerical results that involve GLOBE and several other state-of-the-art algorithms. The conclusion is given in Section VIII.

II. Related Work

A. Limit Order Book

Numerous works are dedicated to modeling the LOB dynamics [16]–[18], while others are concerned with learning to trade efficiently using either static optimization methods [7], [10] or reinforcement learning methods [8], [11], [12]. Apart from these, some other works aim to predict future parameters of the LOB [4], [19], which can help traders to optimize their trading strategies for maximizing the long-term gain.

Our work departs from the prior works related to LOB in two crucial aspects: (i) Similar to prior works, which model the LOB dynamics as a Markov process [16]–[18], we also model the LOB dynamics as a Markov process. However, our state space enjoys a very special decomposition, where each state is composed of a private state and a market state. This decomposition allows us to compute the form of the optimal policy analytically, and also serves as a basis for a computationally efficient and fast online learning algorithm that learns to trade optimally. Moreover, our market state model is novel in the sense that instead of taking the exact price as a state variable, we take the difference between the current bid price and the bid price at the beginning of the current round scaled down by the volatility as the state variable. As justified by our numerical findings in Section VII, this model stays accurate even when the price becomes much lower or higher than the usual range of the price observed in historical data. (ii) To the best of our knowledge, we are the first to define the notion of regret for LOB trade execution and prove that bounded regret is achievable. As opposed to the Q-learning based methods in prior works [8], [12] which only have asymptotic performance guarantees in terms of the average reward (or cost) under strict assumptions on the number of times each state-action pair is observed, our method comes with finite time performance guarantees on the cumulative reward (or cost). Note that bounded regret is a much stronger result than average reward optimality, since every policy with sublinear regret is average reward optimal [20].

B. Reinforcement Learning

Our work is also very closely related to the multi-armed bandit problem [21] and reinforcement learning problem in MDPs [14], [15]. Specifically, our model can be viewed as an episodic MDP, where each round is a new episode.

Numerous works develop reinforcement learning algorithms with regret bounds. For instance, in [14] and [15], the authors consider undiscounted reinforcement learning in ergodic MDPs with unknown state transition probabilities and develop algorithms with $\mathcal{O}(\log T)$ regret[2] with respect to the optimal policy. The authors of [22] consider online learning in an MDP with both Markov and uncontrolled dynamics, and design an algorithm that achieves $\mathcal{O}(T^{1/2} \log T)$ regret. Another Markov model in which the reward function is allowed to arbitrarily change in every time step is proposed in [23], and a policy that achieves $\mathcal{O}(T^{3/4})$ regret with respect to uniformly ergodic class of stationary policies is developed. In addition, an MDP with deterministic state transitions is studied in [24], and [25] and [26] consider episodic MDPs with fixed and variable lengths, respectively. Apart from these, several other works are concerned with model-free online learning methods [27], [28]. Another related work considers the risk-averse multi-armed bandit problem and provides regret bounds for the mean-variance performance measure [29].

Almost all of the works mentioned above that come with regret guarantees use the principle of *optimism under uncertainty* to choose an action or a policy in each round. This principle

[2] T is the time horizon.

TABLE I
COMPARISON OF OUR WORK WITH RELATED WORKS

	Our work	[14], [15], [30]	[7], [10]	[8], [11], [12]
Reinforcement learning	Yes	Yes	No	Yes
Learning technique	Model-based	Model-based	N/A	Model-free
Regret	Bounded	Logarithmic	N/A	No bound

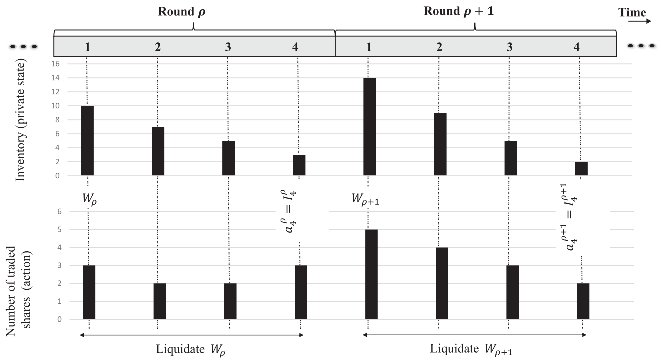

Fig. 1. Illustration of the trading activity. Each round lasts for 4 minutes and consists of 4 time slots. The trader receives the initial inventory W_ρ at the beginning of round ρ, which needs to be liquidated by the end of that round. Thus, at the end of round ρ, the trader always sells a_4^ρ shares, which is equal to the remaining inventory I_4^ρ.

explores rarely-selected actions to decrease the uncertainty over the long-term rewards of the policies that include these actions. Essentially, it serves to balance exploitation (selecting actions according to the estimated optimal policy) and exploration (selecting actions to reduce the uncertainty). As opposed to this approach, the state decomposition in our work enables us to decouple the state transition probabilities from the actions. This allows us to learn the optimal policy by pure exploitation. Since there is no exploration-exploitation tradeoff in our problem, we are able to achieve bounded regret. Apart from our work, there are numerous other settings in which bounded regret is achieved: (i) the multi-armed bandit problem where the expected rewards of the arms are related to each other through a global parameter [31], [32], (ii) a specific class MDPs in which each admissible policy selects every action with a positive probability [33], (iii) combinatorial multi-armed bandits with probabilistically triggered arms, where arm triggering probabilities are strictly positive [34]. A comparison of our work with the related works is given in Table I.

III. Problem Formulation

In this paper we consider the optimal liquidation problem in the LOB with unknown market dynamics. We consider an episodic setting, where at each round ρ the trader must sell a given number of shares over a fixed number of L time slots using market orders. In reality, depending on the trading application, time duration between time slots can be several seconds, minutes or hours. An illustration of how the trading activity takes places over time is given in Fig. 1. Since trading is done via market orders, the revenue from selling a_l^ρ shares at bid price $p_b(\rho, l)$ in time slot l of round ρ is $a_l^\rho p_b(\rho, l)$. Thus, at the end of round ρ, the trader receives as the revenue $\sum_{l=1}^{L} a_l^\rho p_b(\rho, l)$. The goal of the trader is to maximize the revenue incurred over rounds. However, the trader cannot compute the optimal trading strategy beforehand, since it does not know the market dynamics, and hence, the future distribution of the bid prices beforehand. Thus, it needs to maximize its revenue by learning the market dynamics over time.

In the remainder of this section, we give a formal description of the problem faced by the trader by defining states, actions, state transition dynamics, costs, the optimal policy and the regret of the trader.

A. Notation

We use $|\mathcal{A}|$ to denote the cardinality of a set $\mathcal{A}$. The system operates in rounds indexed by $\rho \in \{1, 2, \ldots\}$. Each round is composed of L time slots, where L denotes the *maximum execution time*. The set of time slots is denoted by $\mathcal{L} := \{1, \ldots, L\}$, and the time slots are indexed by $l \in \mathcal{L}$. The current round ends and a new round begins when the maximum execution time is reached.

B. States

The system is composed of a finite set of states denoted by $\mathcal{S} := \mathcal{I} \times \mathcal{M}$, where $\mathcal{I}$ denotes the set of *private states* and $\mathcal{M}$ denotes the set of *market states*. In our model, private states are related to the inventory of the trader, while the market states are related to the dynamics of the bid price.

1) Private State: $\mathcal{I} := \{0, \ldots, W_{\max}\}$ is the set of inventory levels, where $W_{\max}$ is an integer. In addition, the inventory level of shares at the beginning of each round is between $W_{\min}$ and $W_{\max}$, where $W_{\min}$ is an integer such that $0 < W_{\min} \leq W_{\max}$. The private state at time slot l of round ρ is denoted by I_l^ρ. We assume that $I_1^\rho = W_\rho$ where $W_\rho \in \mathcal{I}$ is the initial inventory level at round ρ.

2) Market State: The market states are a set of integers, denoted by $\mathcal{M}$, that are used to define the dynamics of the bid price. Let $M_l^\rho \in \mathcal{M}$ be the market state, and $p_b(\rho, l)$ be the bid price in time slot l of round ρ. It is assumed that the bid price in round ρ evolves according to the following rule: $p_b(\rho, l) = p_b(\rho, 1) + \sigma_\rho M_l^\rho$, where σ_ρ denotes the volatility (standard deviation) of the returns up to round ρ. Obviously, $M_1^\rho = 0 \in \mathcal{M}$, and all the states in $\mathcal{M}$ are assumed to be reachable from state 0 in at most $L - 1$ state transitions. Equivalently, we can define the market state as the difference between the bid prices normalized by the volatility:

$$M_l^\rho = \frac{p_b(\rho, l) - p_b(\rho, 1)}{\sigma_\rho}. \tag{1}$$

In order to define the return of round ρ, we also need $p_a(\rho, l)$, which is the ask price in time slot l of round ρ. Then, the return is $\mathrm{Ret}(\rho) := \log(p_m(\rho, L)/p_m(\rho, 1))$, where $p_m(\rho, l)$ is the mid price (the average of bid and ask prices) in time slot l of round ρ. Hence, the volatility of the returns up to round $\rho > 1$ is simply

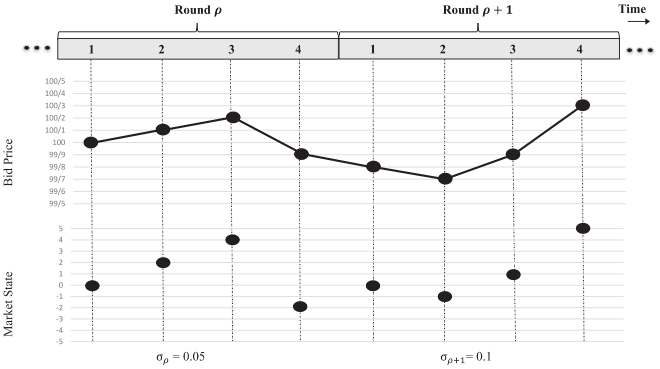

Fig. 2. Illustration of the relationship between the bid prices, the volatility and the market states. Each round lasts 4 minutes and consists of 4 time slots.

calculated as

$$\sigma_\rho = \left(\frac{\sum_{j=1}^{\rho-1} [\text{Ret}(j) - \mu_\rho]^2}{\rho - 1} \right)^{0.5} \tag{2}$$

where $\mu_\rho = \sum_{j=1}^{\rho-1} \text{Ret}(j)/(\rho - 1)$ is the mean of the returns up to round ρ. A sample plot that shows the relation between the market states and the bid prices is given in Fig. 2. Moreover, we define the *joint state* in time slot l of round ρ as $S_l^\rho := (I_l^\rho, M_l^\rho)$.

The intuition behind modeling the movement of the bid price over time as $p_b(\rho, l) = p_b(\rho, 1) + \sigma_\rho M_l^\rho$ is as follows. First of all, since movement of the bid price is defined with respect to the bid price at the beginning of the round, this allows the trader to use the knowledge from past observations in predicting the movement of the bid price even when the stock price enters to an interval which is not observed in the past. For instance, suppose that the past range of the stock price was $[100, 110]$, and the current range is $[120, 140]$. If the stochastic movement of the bid price was defined based on the absolute stock price, then the trader could not use its estimates from the training set to predict the price movement in the current range. Secondly, the amount of price change depends on the volatility (σ_ρ), which is very natural to assume, and has also been used in price models in previous works [7]. In addition, this market model can also be extended to the case when the bid price movement depends on the drift (trend of increase or decrease) as shown in Section VII. Finally, simulation results on real-world datasets given in Section VII show that GLOBE and its variants, which use this definition of the market state, outperform algorithms that use the state definitions proposed in earlier works [8], [12].

C. Actions

Similar to [8], our action set is constructed based on the AC model [7], which gives an optimal liquidation strategy by assuming that the stock price follows an arithmetic random walk with independent increments. In our case, the AC model defines the maximum number of shares that can be sold in a time slot. In the following, we provide a description of the AC model.

1) The AC Model: Suppose a trader wants to liquidate W[3] units of a security in L time slots.[4] Let w_l and A_l be the number of units planned to be hold and sold at time slot $l \in \mathcal{L}$, respectively. By definition, we have $w_1 = W$. The sequence of $\{w_1, w_2, \ldots, w_L\}$ is called the *trading trajectory*, and we have $w_l = W - \sum_{k=1}^{l-1} A_k$. Let ζ_l be independent samples drawn from a distribution with zero mean and unit variance, $g(A_l)$ and $h(A_l)$ be the permanent and temporary price impact functions, respectively.[5] The AC model assumes that the stock price[6] follows an arithmetic random walk with independent increments. Actually, the effective price per share at time slot $l \in \mathcal{L} - \{1\}$ is modelled as

$$p_b(l) = p_b(l-1) + \sigma\zeta_l - g(A_l) - h(A_l).$$

where effect of $h(\cdot)$ vanishes in the next time slot.

The cost of trading, called *the implementation shortfall* (IS) [35] is given as

$$\begin{aligned} \text{IS} &:= W p_b(1) - \sum_{l=1}^{L} p_b(l) A_l \\ &= \sum_{l=1}^{L} [g(A_l) w_l + h(A_l) A_l] - \sum_{l=1}^{L} \sigma \zeta_l w_l \end{aligned}$$

whose distribution is Gaussian if ζ_l are sampled from a Gaussian distribution. The expected value and the variance of the cost are

$$\mathbb{E}(\text{IS}) = \sum_{l=1}^{L} [g(A_l) w_l + h(A_l) A_l], \ \text{Var}(\text{IS}) = \sigma^2 \sum_{l=1}^{L} w_l^2.$$

The objective in the AC model is to minimize $\mathbb{E}(\text{IS}) + \lambda \text{Var}(\text{IS})$ given $\lambda \geq 0$. If $\lambda > 0$, then the optimal policy becomes risk-averse.[7]

Let A_l^* denote the optimal volume to be traded at time slot $l \in \mathcal{L} - \{L\}$. Then, the general solution when $g(A_l) = \gamma A_l$ and $h(A_l) = \eta A_l$ is

$$A_l^* = \frac{2 \sinh(\kappa/2)}{\sinh(\kappa L)} \cosh(\kappa(L - l + 0.5)) W \tag{3}$$

where

$$\kappa = \cosh^{-1}(0.5 \tilde{\kappa}^2 + 1), \ \tilde{\kappa}^2 = \frac{\lambda \sigma}{\eta - 0.5\gamma}.$$

In addition, the general solution under non-linear price impact functions has been considered in [36].

2) Action Set: The action set of our model is based on the AC model. We define actions as the amount of shares to be traded with a market order.[8] We assume that the action taken in time slot $l \in \mathcal{L} - \{L\}$ of round ρ cannot be larger than $A_l^\rho = A_l^*$ obtained in (3) for round ρ.[9] Thus, the set of possible actions

[3] The round index is dropped from all variables in this subsection for simplicity of notation.

[4] The model proposed in [7] is simplified by taking τ (the length of discrete time interval) as 1 and $t_l := l\tau$ as l.

[5] Temporary price impact causes temporary shift of the price from its equilibrium due to our trading strategy which vanishes in the next trading time slot. Permanent price impact refers to the shift in the equilibrium price due to our trading strategy which lasts at least up to the end of a round.

[6] As we consider the liquidation problem, our formulation is given in terms of the bid price.

[7] A policy is risk-averse if the trader would like to select actions such that the variance of the cost does not change much.

[8] A market order to sell is an order to execute a trade at whatever the best prevailing bid price which is a limit order with a price limit of zero at that time.

[9] This choice is made to roughly preserve the risk-awareness of the trader in the AC model. For instance, if $\lambda > 0$, then the strategy is risk-averse and risk-awareness is maintained if the actions are selected from A_l^ρ.

to take in time slot $l \in \mathcal{L} - \{L\}$ of round ρ is defined as $\mathcal{A}_l^\rho := \{0, \ldots, A_l^\rho\}$. Since A_l^ρ and $\mathcal{A}_l^\rho$ are fixed at the beginning of round ρ, when the round we refer to is clear from the context, we will drop superscript ρ, and simply use A_l and $\mathcal{A}_l$. Due to using the AC model, we also have $\sum_{l=1}^{L} A_l^\rho = W_\rho$.

In each round, a sequence of actions is selected with the aim of maximizing the revenue. Let a_l^ρ be the action taken at time slot l in round ρ. For $l = L$, the only possible action is to sell the remaining inventory since we require a complete liquidation at the end of a round. Therefore, we have $a_L^\rho = I_L^\rho$, $\forall \rho \geq 1$.

D. State Transitions

We impose the following assumption on the effect of actions to the market states.

Assumption 1: It is assumed that the order book is resilient to the trader's trading activities.

This assumption holds when the number of shares of a stock traded by the trader in each round forms only a small fraction of the total number of shares of the stock being traded in the market. This implies that the trader's actions do not influence the market states during a round, and is also assumed in other prior works [8], [12]. In practice this means that the market order should not be larger than the depth of the order book at the best bid. This is imposed, for instance, in [7] and [8], which effectively prevents taking large actions (large volume of transaction). Assumption 1 implies that the market state in a round evolves independently from the actions selected by the trader. Hence, the actions only affect the private state, and the market state is modeled as a Markov chain. Let $S' := (I', M')$ and $S := (I, M)$. Then, the state transition probability between time slots l and $l+1$ of round ρ can be written as

$$\mathbb{P}(S_{l+1}^\rho = S' | S_l^\rho = S, a_l^\rho = a) = P(M, M')\mathbb{I}(I' = I - a),$$
$$\forall S \in \mathcal{S}, \forall S' \in \mathcal{S}, \forall a \in \mathcal{A}_l^\rho, \forall l \in \mathcal{L} - \{L\}, \forall \rho \geq 1$$

where $\mathbb{I}(a = b)$ is the indicator function which is zero when $a \neq b$ and one when $a = b$, and $P(M, M')$ denotes the probability that the market state transitions from M to M'. Also, let $\mathbf{P} = \{P(M, M')\}_{M \in \mathcal{M}^r, M' \in \mathcal{M}}$ denote the set of state transition probabilities, where $\mathcal{M}^r$ is the set of states that are reachable from state 0 in at most $L - 2$ state transitions ($\mathcal{M}^r \subset \mathcal{M}$).

E. Cost Function

Similar to [8], we calculate implementation shortfall in round ρ as:

$$\text{IS}_\rho := \frac{W_\rho p_r(\rho) - \sum_{l=1}^{L} a_l^\rho p_b(\rho, l)}{W_\rho p_r(\rho)} \tag{4}$$

for a sequence of market states $(M_1^\rho, \ldots, M_L^\rho)$, a sequence of actions $(a_1^\rho, \ldots, a_L^\rho)$, an inventory level W_ρ such that $\sum_{l=1}^{L} a_l^\rho = W_\rho$, and a reference price $p_r(\rho)$, where the reference price is set as $p_r(\rho) := p_m(\rho, 1)$. The objective is to minimize the accumulated cost in a round, which is equivalent to maximizing the revenue from the trade in that round. The normalization is beneficial as the cost value depends on the ratio of volume being traded at each time slot to the initial inventory level and the ratio of bid price at each time slot to the reference price. Hence, if these ratios remain the same, the cost would be the same regardless of the exact values of these parameters. This allows us to fairly compare performances of different algorithms in different markets and different rounds, even when the reference prices and the initial inventories are different.

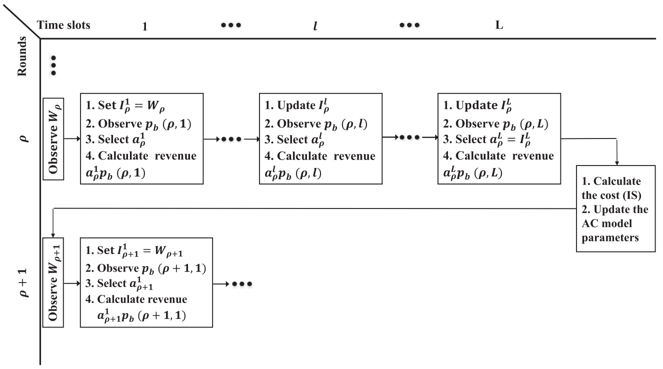

Fig. 3. In each round ρ, the trader first observes the trade vector X_ρ. Then, at each time slot $l \in \mathcal{L}$, it updates the private state I_l^ρ, observes the market state M_l^ρ, and selects a_l^ρ based on this observation. Finally, at the end of each round, the trader calculates the implementation shortfall and updates its trading strategy.

Next, we decompose the implementation shortfall over time slots in a round. For this, we first define the bid-ask spread[10] in time slot l of round ρ as $B_l^\rho := p_a(\rho, l) - p_b(\rho, l)$, and the trade vector of round ρ as $X_\rho := (W_\rho, p_r(\rho), \sigma_\rho, B_1^\rho)$. We assume that X_ρ takes values in a finite set $\mathcal{X}$.[11] By using the state definition and B_1^ρ, (4) can be re-written as

$$\begin{aligned}
\text{IS}_\rho &= \frac{1}{W_\rho p_r(\rho)} \left[\sum_{l=1}^{L} a_l^\rho \left(p_r(\rho) - p_b(\rho, l) \right) \right] \\
&= \frac{1}{W_\rho p_r(\rho)} \left[\sum_{l=1}^{L} a_l^\rho \left(\frac{B_1^\rho}{2} - M_l^\rho \sigma_\rho \right) \right] \\
&= \sum_{l=1}^{L} C_{X_\rho}(M_l^\rho, a_l^\rho)
\end{aligned}$$

where

$$C_{X_\rho}(M_l^\rho, a_l^\rho) := \frac{1}{W_\rho p_r(\rho)} \left[a_l^\rho \left(\frac{B_1^\rho}{2} - M_l^\rho \sigma_\rho \right) \right]$$

is the immediate cost incurred at time slot l of round ρ. Note that our market state definition allows us to decompose the implementation shortfall as a function of the market state.

Finally, the observations and the decisions of the trader at each time slot of a round is shown in Fig. 3.

F. Value Functions and the Optimal Policy

If the state transition probabilities were known in advance, then, the optimal policy can be computed by dynamic programming. In this subsection, we consider this case to gain insight on the form of the optimal policy.

A deterministic Markov policy with time budget L specifies the actions to be taken for each state and trade vector at each time slot. Let $\boldsymbol{\pi} := (\pi_1, \pi_2, \ldots, \pi_L)$ denote such a policy, where

[10] We always have $p_a(\rho, l) \geq p_b(\rho, l)$.

[11] $\mathcal{X}$ can be taken as finite by quantizing the possible values for the reference price, volatility and the bid-ask spread.

$\pi_l : \mathcal{S} \times \mathcal{X} \to \mathcal{A}_l$. We use $\pi_l(M_l, I_l, X)$ to denote the action selected by policy $\boldsymbol{\pi}$ in time slot l when the joint state is (M_l, I_l) in time slot l and the trade vector is X where M_l and I_l represent the market and private variables, respectively. When clear from the context, we will drop the arguments, and represent the action selected by the policy in time slot l by π_l. Moreover, we replace M_l^ρ and I_l^ρ with M_l and I_l when the round is clear from the context. We also let Π denote the set of all deterministic Markov policies with time budget L. The cost incurred by following policy $\boldsymbol{\pi}$ given trade vector $X \in \mathcal{X}$ is given as

$$C_X^\pi := \sum_{l=1}^{L} C_X(M_l, \pi_l(M_l, I_l, X)).$$

The optimal policy is the one that minimizes the expected cost (or equivalently, maximizes the expected revenue), and is given as

$$\boldsymbol{\pi}^*(X) := \arg\min_{\boldsymbol{\pi} \in \Pi} \mathbb{E}[C_X^{\boldsymbol{\pi}}]$$

where the expectation is taken over the randomness of the market states. The expected cost of the optimal policy given the trade vector X is denoted by $\mu^*(X)$.

Let $V_l^*(M, I, X)$ denote the expected cost (V-value) of policy $\boldsymbol{\pi}^*(X)$ starting from joint state (M, I) at time slot l given X. The Bellman optimality equations [37], [38] are given below: $\forall M \in \mathcal{M}^r$, $\forall I \in \mathcal{I}$, $\forall X \in \mathcal{X}$, $\forall l \in \mathcal{L} - \{L\}$,

$$\begin{aligned} &Q_l^*(M, I, X, a) \\ &:= C_X(M, a) + \mathbb{E}[V_{l+1}^*(M', I - a, X) | M] \\ &= C_X(M, a) + \sum_{M' \in \mathcal{M}} P(M, M') V_{l+1}^*(M', I - a, X) \end{aligned} \tag{5}$$

$V_l^*(M, I, X) = \min_{a \in \mathcal{A}_l} Q_l^*(M, I, X, a)$, $\forall l \in \mathcal{L} - \{L\}$. Then, the optimal actions can be computed as $\pi_l^*(M, I, X) = \arg\min_{a \in \mathcal{A}_l} Q_l^*(M, I, X, a)$, $\forall l \in \mathcal{L} - \{L\}$. Note that the Q-values are not defined when $l = L$, and we have $V_L^*(M, I, X) = C_X(M, I)$ and $\pi_L^*(M, I, X) = I$. The optimal policy can be computed by solving these equations backward from time slot L down to 1. In addition, for all policy $\boldsymbol{\pi} \in \Pi$, we use $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ and $V_{l+1}^{\boldsymbol{\pi}}(M, I, X)$ to denote the Q-value and the V-value of the policy given the trade vector X and the joint state (M, I), respectively. Hence, we have

$$\begin{aligned} &V_l^{\boldsymbol{\pi}}(M_l, I_l, X) \\ &:= \mathbb{E}\left[\sum_{k=0}^{L-l} C_X(M_{l+k}, \pi_{l+k}(M_{l+k}, I_{l+k}, X)) \middle| M_l \right] \end{aligned} \tag{6}$$

which is the value of policy $\boldsymbol{\pi}$ at time slot l given the triplet (M_l, I_l, X).

G. Learning and the Regret

We assume that the trader does not know the state transition probabilities of the market Markov chain. Hence, these parameters should be learned and updated online. In round ρ, the trader selects actions based on the estimated optimal policy, denoted by $\hat{\boldsymbol{\pi}}_\rho$, which is calculated based on the estimated transition probabilities, the estimated value functions and the trade vector (denoted by X_ρ) at the beginning of the round. The loss of the trader in terms of the total expected cost with respect to an *oracle*, who knows the state transition probabilities and acts optimally at every round, is defined as the *regret*. The regret by round R given a sequence of trade vectors $(X_1, \ldots, X_R)$ is defined as

$$\mathrm{Reg}(R) := \sum_{\rho=1}^{R} \left(\mathbb{E}\left[C_{X_\rho}^{\hat{\boldsymbol{\pi}}_\rho} \right] - \mu^*(X_\rho) \right). \tag{7}$$

When the regret grows sublinearly over rounds, the average performance of the trader converges to the performance of the optimal policy as $R \to \infty$. Moreover, when the regret is bounded, then, one can show that the trader only takes a finite number of suboptimal actions as $R \to \infty$. Therefore, in the latter case, the trader places all of the market orders optimally only after finitely many rounds. In Section VI, we prove that the expected regret of GLOBE is bounded.

IV. Form of the Optimal Policy

In this section, we show that the optimal policy takes a simple form, which reduces the set of candidates for the optimal action in each time slot to two. Before we discuss the theorem which gives the form of the optimal policy, we decompose the cost function as follows: $C_X(M, a) = a g_X(M)$ where $g_X(M) := (B/2 - M\sigma)/(p_r W)$ for $X = (W, p_r, \sigma, B)$.

Theorem 1: Given the LOB model defined in Section III, the optimal action at each time slot is

$$\pi_l^* = \begin{cases} 0 & \text{if } g_X(M_l) > \mathbb{E}[g_X(M_L)|M_l] \\ A_l & \text{if } g_X(M_l) \le \mathbb{E}[g_X(M_L)|M_l] \end{cases}, \forall l \in \mathcal{L} - \{L\}$$

and $\pi_L^* = I_L$.

Proof: See Appendix A. ■

The theorem above shows that the optimal action at each time slot depends on the current market state and the distribution of the market state at the final time slot given the current state. The trader may decide to sell all of the available limit at the current time slot or hold the shares up to the final time slot. The intuitive reason behind this result is that we have a linear cost function in a and $g_X(M)$. If the expected market state in the final time slot is greater than the current market state, we desire to wait and sell the maximum allowed amount of shares to sell in the current time slot in the final time slot. The reason for this is that, the final time slot is the only time slot where we can sell more than the pre-defined limit. Thus, the set of candidate optimal actions is given as $\mathcal{A}_l^* := \{0, A_l\}$, $\forall l \in \mathcal{L} - \{L\}$. Therefore, the learning problem reduces to learning the best of these two actions in each time slot. This reduces the number of candidate optimal policies from $|\mathcal{A}_1| \times \cdots \times |\mathcal{A}_{L-1}|$ to 2^{L-1}. We denote the set of all candidate optimal policies by Π^{opt}. Finally, it is important to note that Π^{opt} can differ between rounds.

V. The Learning Algorithm

In this section, we propose the learning algorithm for the trader that selects actions by learning the state transition probabilities and exploiting the form of the optimal policy given in the previous section. This algorithm is named as *Greedy exploitation in Limit Order Book Execution* (GLOBE) and its pseudo-code is given in Algorithm 1.

Algorithm 1: GLOBE.

```
Input: $L$, $\mathcal{M}$, $\mathcal{M}^r$
Initialize: $\rho = 1$, $N(M) = 0$, $N(M, M') = 0, \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M}$
while $\rho > 1$ do
    $\hat{P}_\rho(M, M') = \frac{N(M,M') + \mathbb{I}(N(M)=0)}{N(M) + |\mathcal{M}|\mathbb{I}(N(M)=0)}$
    Update $\sigma_\rho$ in the AC model based on the past observations
    Observe $X_\rho = (W_\rho, p_r(\rho), \sigma_\rho, B_1^\rho)$
    Compute $A_l$ based on the AC model [7], $\forall l \in \mathcal{L}$
    Compute the estimated optimal policy by dynamic programming using the action set $\mathcal{A}_l^*, \forall l \in \mathcal{L} - \{L\}$ and $\hat{P}_\rho(M, M'), \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M}$
    $I_1^\rho = W_\rho, l = 1$
    while $l < L$ do
        Observe $M_l^\rho$, sell $a_l^\rho \in \mathcal{A}_l^*$ using the estimated optimal policy
        Calculate $C_{X_\rho}(M_l^\rho, a_l^\rho)$
        $I_{l+1}^\rho = I_l^\rho - a_l^\rho$
        $l = l + 1$
    end while
    $a_L^\rho = I_L^\rho$
    $\rho = \rho + 1$
    Update $N(M, M')$ and $N(M), \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M}$ according to (8) and (9)
end while
```

GLOBE takes as input L, $\mathcal{M}$ and $\mathcal{M}^r$.[12] In addition, it keeps the following counters: $N(M, M')$, which denotes the number of occurrences of a state transition from market state M to M', and $N(M)$, which denotes the number of times market state M is visited before the final time slot by the beginning of the current round. We use $N_\rho(M, M')$ and $N_\rho(M)$ to denote the values of these counters at the beginning of round ρ. Using these, GLOBE calculates an estimate of the transition probability from state M to M' used in round ρ, which is denoted by $\hat{P}_\rho(M, M')$. Thus, we have $\forall M \in \mathcal{M}^r$ and $\forall M' \in \mathcal{M}$

$$N_\rho(M, M') = \sum_{\rho'=1}^{\rho-1} \sum_{l=1}^{L-1} \mathbb{I}(M_{\rho'}^l = M)\mathbb{I}(M_{\rho'}^{l+1} = M'), \quad (8)$$

$$N_\rho(M) = \sum_{M' \in \mathcal{M}} N_\rho(M, M'),$$

$$\hat{P}_\rho(M, M') = \frac{N_\rho(M, M') + \mathbb{I}(N_\rho(M) = 0)}{N_\rho(M) + |\mathcal{M}|\mathbb{I}(N_\rho(M) = 0)}. \quad (9)$$

At the beginning of round ρ GLOBE first updates the volatility σ_ρ based on (2). Alternatively, one may also use the techniques proposed in [39], [40] for volatility estimation. Then, it uses the AC model to obtain the maximum number of shares to sell in each time slot, i.e., A_l, and implements dynamic programming with action set $\mathcal{A}_l^*$, obtained from Theorem 1, instead of $\mathcal{A}_l$, $l \in \mathcal{L} - \{L\}$ and set of transition probabilities $\{\hat{P}_\rho(M, M')\}_{M \in \mathcal{M}^r, M' \in \mathcal{M}}$ to find the estimated optimal policy, and follows that policy during the round to sell the shares. The above procedure repeats in each round.

As an alternative, GLOBE can also use the rule given in Theorem 1 to decide on whether to sell A_l or 0 in time slot l of round ρ, by finding the expected market state in the final time slot of that round using $\hat{P}_\rho(M, M')$ values, and then comparing $g_{X_\rho}(M_l^\rho)$ and $\mathbb{E}_{\hat{P}_\rho}[g_{X_\rho}(M_L)|M_l^\rho]$. However, the computational complexity of this method is greater than that of dynamic programming that uses the reduced action set.

Remark 1: The number of multiplications for calculating the expectation given in Theorem 1 for all time slots is $O(L|\mathcal{M}|^{2.374})$ using algorithms optimized for matrix multiplication [41]. On the other hand, dynamic programming with reduced action set requires $O(L|\mathcal{M}|^2)$ multiplications when computing the optimal policy.

VI. Regret Analysis

In this section, we upper bound the regret of GLOBE defined in (7). Before delving into the details, we state a lemma, which gives an explicit formula for the maximum possible estimation error in the state transition probabilities (denoted by δ) such that the estimated optimal policy is the same as the true optimal policy.

Recall that we have

$$Q_l^{\boldsymbol{\pi}}(M, I, X, a) = C_X(M, a) + \sum_{M' \in \mathcal{M}} P(M, M') V_{l+1}^{\boldsymbol{\pi}}(M', I - a, X)$$

$\forall M \in \mathcal{M}^r$, $\forall I \in \mathcal{I}$, $\forall X \in \mathcal{X}$, $\forall a \in \mathcal{A}_l$, $\forall l \in \mathcal{L} - \{L\}$, $\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}$. We do not need to calculate $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ for $M \notin \mathcal{M}^r$ as these states are never observed in the first $L - 1$ time slots of a round.

The estimate of $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ in round ρ is given as

$$\hat{Q}_{l,\rho}^{\boldsymbol{\pi}}(M, I, X, a) := C_X(M, a) + \sum_{M' \in \mathcal{M}} \hat{P}_\rho(M, M') \hat{V}_{l+1,\rho}^{\boldsymbol{\pi}}(M', I - a, X)$$

$\forall M \in \mathcal{M}^r$, $\forall I \in \mathcal{I}$, $\forall X \in \mathcal{X}$, $\forall a \in \mathcal{A}_l$, $\forall l \in \mathcal{L} - \{L\}$, $\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}$ where $\hat{V}_{l,\rho}^{\boldsymbol{\pi}}(M, I, X)$ is the estimated V-value of policy $\boldsymbol{\pi}$ in joint state (M, I) given the trade vector X in time slot l of round ρ. In order to bound the regret, we need to analyze the distance between $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ and $\hat{Q}_{l,\rho}^{\boldsymbol{\pi}}(M, I, X, a)$. As a first step, we derive the form of $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ as a function of the state transition probabilities. In order to simplify the notation, in the next two lemmas, we use $\hat{P}(M, M')$ instead of $\hat{P}_\rho(M, M')$, when the round is clear from the context.

Let $POL(Z^{0:y}, x)$ be a yth order polynomial function of the variable x with the coefficients $Z^{0:y} := (Z^y, Z^{y-1}, \ldots, Z^0)$ where Z^i, $i \in \{0, \ldots, y\}$ is the coefficient that multiplies x^i.

Lemma 1: For all $\boldsymbol{\pi} \in \Pi^{\text{opt}}$, $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$ is a polynomial function of $P(\tilde{M}, \tilde{M}')$, $\forall \tilde{M} \in \mathcal{M}^r$ and $\forall \tilde{M}' \in \mathcal{M}$ where the order is at most $L - l$. Let $Z_{\tilde{M},\tilde{M}'}^{\boldsymbol{\pi},i}(M, I, X, a, l)$ and $Z_{\tilde{M},\tilde{M}'}^{\boldsymbol{\pi},0:L-l}(M, I, X, a, l)$ be the coefficient of $(P(\tilde{M}, \tilde{M}'))^i$ and the set of all coefficients of the powers of $P(\tilde{M}, \tilde{M}')$ in $Q_l^{\boldsymbol{\pi}}(M, I, X, a)$, respectively. Hence, we have

$$Q_l^{\boldsymbol{\pi}}(M, I, X, a) = POL(Z_{\tilde{M},\tilde{M}'}^{\boldsymbol{\pi},0:L-l}(M, I, X, a, l), P(\tilde{M}, \tilde{M}'))$$

[12] In practice, $\mathcal{M}$ and $\mathcal{M}^r$ can be easily computed from historical data since the market state measures price movement relative to the reference price at the beginning of a round. Therefore, in the numerical analysis in Section VII, GLOBE runs by using estimated $\mathcal{M}$ and $\mathcal{M}^r$ sets computed using historical data.

$\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}$, $\forall M \in \mathcal{M}^r$, $\forall I \in \mathcal{I}$, $\forall X \in \mathcal{X}$, $\forall a \in \mathcal{A}_l$, $\forall l \in \mathcal{L} - \{L\}$, $\forall \tilde{M} \in \mathcal{M}^r$ and $\forall \tilde{M}' \in \mathcal{M}$.

Proof: See Appendix B. ■

In the following lemma, we derive the relation between the error in the estimated transition probabilities and the error in the estimated Q-values. Given a set of feasible actions $\mathcal{A}_1, \ldots, \mathcal{A}_{L-1}$, let $\hat{Q}_l^{\boldsymbol{\pi}}(M_l, I_l, X, a_l)$ be the estimated value of $Q_l^{\boldsymbol{\pi}}(M_l, I_l, X, a_l)$ when $\hat{P}(M, M')$ has been used instead of $P(M, M')$, $\forall M \in \mathcal{M}^r$ and $\forall M' \in \mathcal{M}$.

Lemma 2: Consider a set of feasible actions $\mathcal{A}_1, \ldots, \mathcal{A}_{L-1}$ computed by the AC model. Let $N^r := |\mathcal{M}||\mathcal{M}^r|$, λ be any value in $(0, 1)$,

$$H := \sup_{\boldsymbol{\pi} \in \Pi^{\text{opt}}, I, X, a, l, \tilde{M} \in \mathcal{M}^r, M \in \mathcal{M}^r, M' \in \mathcal{M}, \mathbf{P} \in \mathcal{P}} \sum_{i=1}^{L} i |Z_{M,M'}^{\boldsymbol{\pi}, i}(\tilde{M}, I, X, a, l)|$$

where $\mathcal{P}$ denotes the set of all state transition probability matrices where only the state transition probabilities from $\mathcal{M}^r$ to $\mathcal{M}$ are allowed to be non-zero and $\gamma := \lambda/(N^r H)$. If

$$|P(M, M') - \hat{P}(M, M')| \leq \gamma, \ \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M}$$

then we have

$$|Q_l^{\boldsymbol{\pi}}(M_l, I_l, X, a_l) - \hat{Q}_l^{\boldsymbol{\pi}}(M_l, I_l, X, a_l)| \leq \lambda$$

$\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}, \forall M_l \in \mathcal{M}^r, \forall I_l \in \mathcal{I}, \forall X \in \mathcal{X}, \forall a_l \in \mathcal{A}_l, \forall l \in \mathcal{L} - \{L\}$.

Proof: See Appendix C. ■

Let $\Delta_X^{\boldsymbol{\pi}} := \mathbb{E}\left[C_X^{\boldsymbol{\pi}}\right] - \mu^*(X)$ denote the suboptimality gap of policy $\boldsymbol{\pi}$ given trader vector X, and

$$\Pi_X^{\text{sub}} := \{\boldsymbol{\pi} \in \Pi^{\text{opt}} : \Delta_X^{\boldsymbol{\pi}} > 0\}$$

denote the set of suboptimal policies (among the set of candidate optimal policies) given trade vector X. Let

$$\Delta_{\max} := \max_{X \in \mathcal{X}, \boldsymbol{\pi} \in \Pi_X^{\text{sub}}} \Delta_X^{\boldsymbol{\pi}}$$

$$\Delta_{\min} := \min_{X \in \mathcal{X}, \boldsymbol{\pi} \in \Pi_X^{\text{sub}}} \Delta_X^{\boldsymbol{\pi}}$$

be the maximum and the minimum suboptimality gaps, which upper bound and lower bound the expected regret of a round in which a suboptimal policy is chosen, respectively.

From Lemma 2 and the fact that $V_l^{\boldsymbol{\pi}}(M_l^\rho, I_l^\rho, X_\rho) = Q_l^{\boldsymbol{\pi}}(M_l^\rho, I_l^\rho, X_\rho, \pi_l(M_l^\rho, I_l^\rho, X_\rho))$, it is straightforward to see that if $|P(M, M') - \hat{P}_\rho(M, M')| \leq \lambda/(N^r H)$ for all $M \in \mathcal{M}^r$ and all $M' \in \mathcal{M}$, then we have $\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}, \forall M_l^\rho \in \mathcal{M}^r, \forall I_l^\rho \in \mathcal{I}, \forall X_\rho \in \mathcal{X}, \forall l \in \mathcal{L} - \{L\}, \forall \rho \geq 1$:

$$|V_l^{\boldsymbol{\pi}}(M_l^\rho, I_l^\rho, X_\rho) - \hat{V}_{l,\rho}^{\boldsymbol{\pi}}(M_l^\rho, I_l^\rho, X_\rho)| \leq \lambda. \tag{10}$$

Moreover, if (10) holds, then by the result in Appendix D, we have

$$|V_l^*(M_l^\rho, I_l^\rho, X_\rho) - V_{l,\rho}^{\hat{\boldsymbol{\pi}}_\rho}(M_l^\rho, I_l^\rho, X_\rho)| \leq 2\lambda.$$

From (6) we know that

$$\mathbb{E}\left[C_{X_\rho}^{\boldsymbol{\pi}}\right] = \mathbb{E}\left[\sum_{l=1}^{L} C_{X_\rho}(M_l^\rho, \pi_l(M_l^\rho, I_l^\rho, X_\rho))\right] = V_{1,\rho}^{\boldsymbol{\pi}}(M_1^\rho, I_1^\rho, X_\rho)$$

where $M_1^\rho = 0$ and $I_1^\rho = W_\rho$. Hence, if (10) holds, then we have

$$|\mu^*(X_\rho) - \mathbb{E}\left[C_{X_\rho}^{\hat{\boldsymbol{\pi}}_\rho}\right]| \leq 2\lambda.$$

Let $\delta := \Delta_{\min}/(3N^r H)$. Based on the discussion above, if

$$|P(M, M') - \hat{P}_\rho(M, M')| \leq \delta, \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M} \tag{11}$$

then we have

$$|\mu^*(X_\rho) - \mathbb{E}\left[C_{X_\rho}^{\hat{\boldsymbol{\pi}}_\rho}\right]| < \Delta_{\min}.$$

Let

$$O_\rho := \{\hat{\boldsymbol{\pi}}_\rho \notin \Pi_{X_\rho}^{\text{sub}}\}$$

be the event of selecting an optimal policy in round ρ. Thus, if (11) holds, we have $\hat{\boldsymbol{\pi}}_\rho \notin \Pi_{X_\rho}^{\text{sub}}$. This implies that

$$\bigcap_{M \in \mathcal{M}^r, M' \in \mathcal{M}} \{|P(M, M') - \hat{P}_\rho(M, M')| \leq \delta\} \subset O_\rho.$$

Using the statement above, we also obtain

$$O_\rho^C \subset \bigcup_{M \in \mathcal{M}^r, M' \in \mathcal{M}} \{|P(M, M') - \hat{P}_\rho(M, M')| \geq \delta\}.$$

Let $\mathcal{M}_0 := \{(M, M') : M \in \mathcal{M}^r, M' \in \mathcal{M}, P(M, M') = 0\}$. For all $(M, M') \in \mathcal{M}_0$, if $N_\rho(M) > 0$, then we have $\hat{P}_\rho(M, M') = P_\rho(M, M') = 0$ which means that the estimation error is zero. Then, by using the union bound and the definition of O_ρ, we can divide the sum in (7) into two parts as follows:

$$\begin{aligned}
\mathbb{E}[\text{Reg}(R)] &\leq \mathbb{E}\left[\sum_{\rho=1}^{R} \mathbb{I}(O_\rho^C)\right] \Delta_{\max} = \sum_{\rho=1}^{R} \mathbb{E}\left[\mathbb{I}(O_\rho^C)\right] \Delta_{\max} \\
&\leq \sum_{\rho=1}^{R} \sum_{(M,M') \notin \mathcal{M}_0} \mathbb{P}\left(|P(M, M') - \hat{P}_\rho(M, M')| \geq \delta\right) \Delta_{\max} \\
&+ \sum_{\rho=1}^{R} \sum_{(M,M') \in \mathcal{M}_0} \mathbb{P}\left(|P(M, M') - \hat{P}_\rho(M, M')| \geq \delta\right) \Delta_{\max}.
\end{aligned} \tag{12}$$

Thus, all we need is to bound the convergence rate of the estimated state transition probabilities to the true values.

We proceed by showing that for $M \in \mathcal{M}^r$, $N_\rho(M)$ is not smaller than a linear function of ρ, with a very high probability for large ρ. To show this, let

$$\epsilon := \min_{M \in \mathcal{M}^r} \mathbb{P}(\cup_{l=1}^{L-1}\{M_l = M\})$$

be a lower bound on the probability that state M is observed in the first $L - 1$ time slots of a round. Since $M \in \mathcal{M}^r$, we have $\epsilon > 0$.

Lemma 3: For all $M \in \mathcal{M}^r$

$$\mathbb{P}\left(N_{\rho+1}(M) \leq 0.5\epsilon\rho\right) \leq \frac{1}{\rho^2} \text{ for } \rho \geq \rho'$$

where ρ' is the smallest integer such that $\log \rho/\rho \leq 0.25\epsilon^2$ for all $\rho \geq \rho'$.

Proof: See Appendix E. ■

Lemma 3 will be used to show that the estimated state transition probabilities for $(M, M') \notin \mathcal{M}_0$ are close to their true values for $\rho \geq \rho'$. Below, we povide an upper bound on ρ' in terms of ϵ.

Lemma 4: $\rho' \leq 1 + (12/(\epsilon e))^3$.

Proof: Let $B_1(\rho) := 0.5\epsilon\rho - \sqrt{\rho \log \rho}$, $B_2(\rho) := 0.5\epsilon\rho - \sqrt{\rho}\log\rho$. By definition, $\rho' > 1$ is the smallest round in which $B_1(\rho) \geq 0$ for all $\rho \geq \rho'$. Let ρ^* be the largest solution of $B_2(\rho) = 0$. Next, we show that $\rho^* + 1 \geq \rho'$. For $\rho \geq 3$, we have $\sqrt{\log \rho} \leq \log \rho$, and hence, $\sqrt{\rho \log \rho} \leq \sqrt{\rho}\log\rho$, which implies that $B_1(\rho) \geq B_2(\rho)$.

Since ρ^* is the largest solution to $B_2(\rho) = 0$, we have $B_2(\rho) > 0$ for $\rho > \rho^*$, which also implies that $B_1(\rho) > 0$ for $\rho > \rho^*$. Therefore, we must have $\rho^* + 1 \geq \rho'$. Next, we will find an upper bound on ρ^*. $B_2(\rho^*) = 0$ implies that $0.5\epsilon\rho^* = \sqrt{\rho^*}\log\rho^* \Rightarrow \rho^* = (2/\epsilon)\sqrt{\rho^*}\log\rho^*$. Then, according to Part B of Appendix G (by setting $x = \rho^*$, $v = 0.5$ and $a_3 = 2/\epsilon$), we obtain $\rho^* \leq (12/(\epsilon e))^3$. Thus, we have $\rho' \leq 1 + (12/(\epsilon e))^3$. ■

The following lemma bounds the deviation of the estimated state transition probabilities from the true state transition probabilities for $(M, M') \in \mathcal{M}_0$.

Lemma 5: For all $(M, M') \in \mathcal{M}_0$

$$\mathbb{P}(|\hat{P}_\rho(M, M') - P(M, M')| \geq \delta) \leq (1 - \epsilon)^{\rho - 1}.$$

Proof: For all $(M, M') \in \mathcal{M}_0$, we have

$$\begin{aligned}&\mathbb{P}(|\hat{P}_\rho(M, M') - P(M, M')| \geq \delta)\\ &\leq \mathbb{P}(|\hat{P}_\rho(M, M') - P(M, M')| > 0)\\ &= \mathbb{P}(N_\rho(M) = 0) \leq (1 - \epsilon)^{\rho - 1}.\end{aligned}$$

■

Finally, we combine the results of (12), Lemmas 3, 4 and 5 to bound the expected regret in the following theorem.

Theorem 2: The expected regret of GLOBE by round R is bounded by

$$\begin{aligned}\mathbb{E}[\text{Reg}(R)] &\leq \left(\rho' + \frac{9N^r}{\epsilon\delta^2}\right)\Delta_{\max}\\ &\leq \left(1 + \left(\frac{12}{\epsilon e}\right)^3 + \frac{9N^r}{\epsilon\delta^2}\right)\Delta_{\max}\end{aligned}$$

where ρ' is the smallest value such that $\log\rho/\rho \leq 0.25\epsilon^2$ for all $\rho \geq \rho'$, and the second inequality follows from Lemma 4.

Proof: See Appendix F. ■

Theorem 2 shows that the expected regret of GLOBE is bounded, i.e., $\mathbb{E}[\text{Reg}(R)] = \mathcal{O}(1)$. Moreover, the expected regret is inversely proportional to ϵ and δ, since GLOBE needs more accurate estimations of state transition probabilities in order to select the optimal policy when ϵ or δ is small.

Since regret of GLOBE is bounded, GLOBE selects a suboptimal action or policy only in finitely many rounds with probability one.

Corollary 1: $\mathbb{P}(O_\rho^C \text{ occurs infinitely often}) = 0$.

Proof: From Theorem 2, we have $\sum_{\rho=1}^{\infty} \mathbb{P}(O_\rho^C) < \infty$ The result follows from the Borel-Cantelli lemma. ■

VII. Illustrative Results

A. Simulation Setup

We consider six order book datasets: Apple, Amazon, Google, Intel-com and Microsoft shares traded in NASDAQ, which are abbreviated as AAPL, AMZN, GOOG, INTC and MSFT, respectively[13] and Eurodollar short term interest rate (STIR) future contract traded in Chicago Mercantile Exchange (CME). The time horizon of the first five datasets is approximately 6 hours and 30 minutes and for the last dataset, it is approximately 90 hours and 45 minutes. Among all information available in the datasets, we use the market bid/ask prices and bid/ask volumes over time. More information on these datasets is given in Appendix H.

The trader wants to sell W_ρ number of shares in round ρ at the best price using market orders. We obtain the market state from the real-world bid price as follows. By using (1), we find the market state in time slot l of round ρ as M, where M satisfies

$$\left| M - \frac{p_b(\rho, l) - p_b(\rho, 1)}{\sigma_\rho} \right| \leq 0.5.$$

The number of states varies from dataset to dataset based on the volatility scale. To reduce the number of market states, we use a scale factor K, where instead of σ_ρ we use $K\sigma_\rho$ in the market state definition and the above inequalities. Tuning of the hyper-parameter K as well as tuning of the other hyper-parameters are done via validation (see Section VII-D).

We define $\mathcal{M}^r(\rho)$ and $\mathcal{M}(\rho)$ as the set of states which belong to $\mathcal{M}^r$ and $\mathcal{M}$ and have been observed by the beginning of round ρ, respectively. After each round, these sets are updated. For instance, let $\mathcal{M}_\rho$ be the set of states observed in round ρ. Then, we have $\mathcal{M}(\rho + 1) = \mathcal{M}(\rho) \cup \mathcal{M}_\rho$. A similar update rule also applies to $\mathcal{M}^r(\rho)$. Note that $\mathcal{M}^r(\rho)$ and $\mathcal{M}(\rho)$ will converge to $\mathcal{M}^r$ and $\mathcal{M}$ as the number of rounds increase.

Next, we continue by explaining the remaining simulation parameters. Each data instance for each time slot is created by taking the average of the mid/bid/ask prices for every 10 second interval. Then, the dataset is divided into rounds, where each round consists of $L = 4$ consecutive time slots. The initial inventory level of each round is drawn uniformly at random from $[10, 50]$. The volatility parameter used in the AC model is updated online.

Furthermore, similar to [8], the permanent price impact parameter is set to 0, and the temporary price impact parameter is updated online according to [7]. Although one can specify a fixed value of λ in the AC model, we decided to tune this parameter for each algorithm separately.

B. Algorithms

Next, we describe the algorithms that we compare GLOBE against:[14]

1) EQ: In this method, the shares are equally[15] divided among the time slots. Hence, at each time slot of round ρ, the trader sells $\lfloor W_\rho / L \rfloor$ [10], except the final time slot where the remaining inventory is sold. EQ does not have any hyper-parameter.

2) AC: This policy is defined in [7] and discussed in Section III-C1. Different from [7], the volatility and temporary price impact parameters are updated after each round. In addition, the suggested number of shares to be sold in each time slot is rounded to an integer value. The remaining inventory is sold in the final time slot. For AC, the penalty (λ) is the hyper-parameter.

[13] See https://lobsterdata.com/info/DataSamples.php.

[14] The results of all of the Q-learning based methods are averaged over 50 runs.

[15] The abbreviation EQ comes from EQual.

3) Q-Exp: This is a Q-learning method, which uses the state set defined in [8] and the action set defined in our paper. It uses the ϵ-greedy policy [28] which explores with probability p_{exp} and exploits with probability $1 - p_{\text{exp}}$. In this method, the set of market states is the combination of bid-ask spread and bid volumes as proposed by [8]. The number of market states, denoted by N_Q, p_{exp} and λ are the hyper-parameters of Q-Exp.

4) Q-Mat: This is a variant of the method proposed in [8], which uses the state set defined in [8] and the action set defined in our paper. This method uses a training set to calculate the Q-values, and builds a Q-matrix for all combinations of market states (bid-ask spread and traded volume), inventory states, actions and time slots. Then, it uses this Q-matrix on the test set. The hyper-parameters of Q-Mat are N_Q and λ.

5) GLOBE: GLOBE is given in Algorithm 1. The hyper-parameters of GLOBE are K and λ.

6) C-GLOBE: This is the contextual version of GLOBE. C-GLOBE takes the drift (trend of increase or decrease in the bid price) as the context. The drift in round ρ is denoted by d_ρ, and is calculated based on a window of past instances K_w as follows for $\rho > 1$:

$$\mu_\rho^d = \frac{\sum_{j=\max\{1,\rho-K_w\}}^{\rho-1} \text{Ret}(j)}{\min\{K_w, \rho-1\}},$$

$$d_\rho = \left(\frac{\sum_{j=\max\{1,\rho-K_w\}}^{\rho-1} [\text{Ret}(j) - \mu_\rho^d]^2}{\min\{K_w, \rho-1\}} \right)^{0.5}.$$

The context set is divided into two parts: a part in which the drift is negative and another part in which the drift is nonnegative. Then, two different instances of GLOBE are run for the two parts of the context set. First, C-GLOBE calculates the drift in the current round and determines whether it is negative or not. Then, it chooses the instance of GLOBE to run based on the value of the drift. This way, the algorithm keeps two different sets of state transition probability estimates: one for negative drift and one for nonnegative drift. The hyper-parameters of C-GLOBE are K_w, λ and K.

7) Extended Versions: Here, we introduce Q-Exp+, Q-Mat+, GLOBE+, C-GLOBE+, which are variants of Q-Exp, Q-Mat, GLOBE and C-GLOBE respectively, whose action sets in time slot l are $\{0, 1, \dots, 2A_l\}$ instead of $\{0, 1, \dots, A_l\}$. Such a modification allows exploration of a larger set actions, and is adopted from [8]. Since the action sets of GLOBE+ and C-GLOBE+ are different from GLOBE and C-GLOBE, Theorem 1 does not hold, and thus, we use dynamic programming with action set $\{0, 1, \dots, 2A_l\}$ for these algorithms. The hyper-parameters of the extended versions are the same as the hyper-parameters of the original algorithms.

C. Performance Measure

For each method, we calculate the Averaged Cost Per Round (ACPR), which is given as $\text{ACPR}_R = \frac{1}{R}\sum_{\rho=1}^{R} \text{IS}_\rho$ to measure the performance for R rounds. Then, we compare ACPR_R of each method (alg) against AC starting from the first round in the test set, using a performance metric similar to the one used in [19], which we call the Relative Improvement per round (RI), given as

$$\text{RI}_R(alg) := \frac{\text{ACPR}_R(AC) - \text{ACPR}_R(alg)}{|\text{ACPR}_R(AC)|} \times 100.$$

TABLE II
SET OF HYPER-PARAMETERS

	λ	K	K_w	N_Q	p_{exp}
First choice	0.01	10	50	25 (5×5)	0.05
Second choice	0.1	20	100	100 (10×10)	0.1

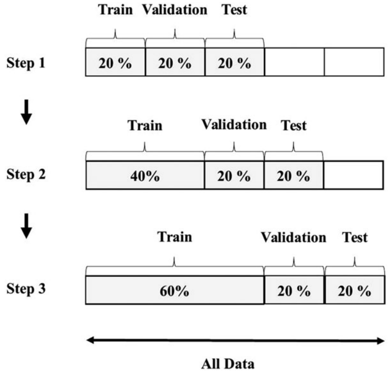

Fig. 4. Steps of the validation procedure.

If an algorithm outperforms (under-performs) AC, then its RI is positive (negative). The reason behind comparing with AC arises from the fact that the action set of all algorithms (except EQ) are built based on the AC model.

D. Hyper-Parameter Selection via Validation

In order to tune the hyper-parameters of the algorithms, we divide each dataset into three blocks without disrupting chronological order of the events: A training block that contains either 20%, 40% or 60% of the samples, a validation block that contains 20% of the samples and a test block that contains 20% of the samples, respectively. Then, the algorithms are trained on the training block for all hyper-parameter values listed in Table II, and the best hyper-parameter values are chosen as the ones which give the lowest ACPR on the validation block. Then, the performances are reported on the test block. This procedure is done three times for three different test blocks as illustrated in Fig. 4. The size of training block increases in each step as more samples are observed. In addition, the hyper-parameters are adjusted dynamically at each step, which is consistent with the online nature of the data. We would also like to note that unlike Q-MAT and Q-MAT+, which learn only over the training set, all versions of GLOBE and Q-EXP continue learning over the validation and test sets.

E. Simulation Results

In Table III we report the average RI of all algorithms on the three test blocks in Fig. 4. The ACPR of the algorithms are reported in Table V in Appendix I. We observe that GLOBE and its variants (C-GLOBE, GLOBE+, C-GLOBE+) outperform Q-learning-based methods (Q-MAT, Q-EXP, Q-MAT+, Q-EXP+) in all of the datasets. Specifically, GLOBE (GLOBE+) and C-GLOBE (C-GLOBE+) have better performance than Q-MAT

TABLE III
RI AND SD OF IMPLEMENTATION SHORTFALL OF ALL ALGORITHMS AT THE END OF THE TIME HORIZON WITH RESPECT TO THE AC MODEL CALCULATED OVER THE TEST SETS. ALL SD VALUES ARE MULTIPLIED BY 10^4. THE BEST TWO ARE SHOWN IN BOLD

Algorithms / Dataset	AAPL (RI)	AMZN (RI)	GOOG (RI)	INTC (RI)	MSFT (RI)	EDC (RI)	AAPL (SD)	AMZN (SD)	GOOG (SD)	INTC (SD)	MSFT (SD)	EDC (SD)
AC	**0.00**	0.00	0.00	0.00	0.00	0.00	**1.33**	**2.33**	1.58	1.48	1.55	**0.25**
EQ	-1.55	0.3	-0.17	-3.72	-0.79	0.01	1.44	2.38	1.62	2.01	2.19	**0.25**
Q-MAT	-7.99	-5.33	-7.69	-13.74	-10.16	0.22	2.31	3.96	2.85	3.65	3.69	0.41
Q-EXP	-6.08	-2.93	-5.97	-9.82	-5.36	0.17	2.05	3.50	2.40	3.04	3.08	0.38
GLOBE	-1.43	0.26	-0.76	2.02	1.00	1.19	1.54	2.72	1.86	1.55	1.41	0.28
C-GLOBE	-2.49	-1.42	-0.77	2.08	3.50	0.89	1.51	3.94	1.85	1.52	1.54	0.34
Q-MAT+	-6.48	-5.04	-4.71	-12.89	-11.12	0.05	2.17	3.68	2.61	3.56	3.52	0.41
Q-EXP+	-1.37	-0.31	-1.06	-3.42	-0.82	0.21	1.52	2.51	1.76	2.15	1.84	0.36
GLOBE+	**4.38**	**3.93**	**6.23**	**7.96**	**5.81**	**1.53**	**0.99**	**2.35**	**1.14**	**0.92**	**0.68**	**0.24**
C-GLOBE+	-5.67	**1.29**	**1.43**	**6.91**	**5.58**	**1.49**	1.80	3.63	**1.44**	**1.06**	**1.25**	0.33

(Q-MAT+) and Q-EXP (Q-EXP+) in general. We think that the market state model proposed in this paper allows GLOBE and its variants to learn faster than the Q-learning based methods.

In addition, the standard deviation (SD) of the implementation shortfall calculated over the test sets for AC algorithm is usually among the best ones. This is expected since the penalty term of the AC model is tuned to be positive, which makes it risk-averse. Also, the standard deviations of the cost incurred during the test rounds for GLOBE (GLOBE+) and C-GLOBE (C-GLOBE+) are almost always better than Q-EXP (Q-EXP+) and Q-MAT (Q-MAT+). The RI of the Q-learning based methods are better in EDC dataset than the other datasets due to the fact that this dataset contains a higher number of samples than the others. However, GLOBE+ and C-GLOBE+ still outperform Q-MAT+ and Q-EXP+. In essence, based on both RI and SD, under the same action set, GLOBE and its variants perform better than the Q-learning based methods. This result shows that the good performance of our proposed methods hold with a higher confidence (lower risk) than the other learning methods.

VIII. CONCLUSION

In this paper, we proposed an online learning algorithm for trade execution in LOB. We modeled this problem as an MDP using a novel market state definition, and derived the form of the optimal policy for this MDP. Then, we developed a learning algorithm that learns to trade optimally using the state transition probability estimates, and proved that it achieves bounded regret. We also showed that our method outperforms its competitors in numerous finance datasets. As a future work, we will investigate the performance of GLOBE on other datasets with other types of stocks.

APPENDIX A
PROOF OF THEOREM 1

Using the tower property of the conditional expectation and (6), we obtain

$$\begin{aligned}
&\mathbb{E}[V_l^{\boldsymbol{\pi}}(M_l, I_l, X)|M_{l-1}]\\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_{k=0}^{L-l} C_X(M_{l+k}, \pi_{l+k}(M_{l+k}, I_{l+k}, X))\middle| M_l\right]\middle| M_{l-1}\right]\\
&= \sum_{k=0}^{L-l} \mathbb{E}\left[\mathbb{E}\left[C_X(M_{l+k}, \pi_{l+k}(M_{l+k}, I_{l+k}, X))\middle| M_l\right]\middle| M_{l-1}\right]\\
&= \sum_{k=0}^{L-l} \mathbb{E}[C_X(M_{l+k}, \pi_{l+k}(M_{l+k}, I_{l+k}, X))|M_{l-1}].
\end{aligned}$$

We use backward induction to prove the theorem. Induction basis consists of time slots L, $L-1$ and $L-2$.

Induction basis:

Since all shares must be sold by the end of a round, in time slot L, the trader must sell all remaining shares I_L. Hence, we have $\pi_L^* = I_L$. We also have $\mathbb{E}\left[V_L^*(M_L, I_L, X)|M_{L-1}\right] = \mathbb{E}\left[C_X(M_L, I_L)|M_{L-1}\right] = \mathbb{E}\left[I_L g_X(M_L)|M_{L-1}\right]$. Thus, for π_{L-1}^*, we have

$$\begin{aligned}
&\pi_{L-1}^*(M_{L-1}, I_{L-1}, X)\\
&= \underset{a\in\mathcal{A}_{L-1}}{\arg\min}\,\{C_X(M_{L-1}, a)\\
&\qquad + \mathbb{E}[V_L^*(M_L, I_{L-1}-a, X)|M_{L-1}]\}\\
&= \underset{a\in\mathcal{A}_{L-1}}{\arg\min}\{C_X(M_{L-1}, a) + \mathbb{E}\left[C_X(M_L, I_{L-1}-a)|M_{L-1}\right]\}\\
&= \underset{a\in\mathcal{A}_{L-1}}{\arg\min}\{a g_X(M_{L-1}) + \mathbb{E}[(I_{L-1}-a) g_X(M_L)|M_{L-1}]\}\\
&= \underset{a\in\mathcal{A}_{L-1}}{\arg\min}\{a g_X(M_{L-1}) + \mathbb{E}[-a g_X(M_L)|M_{L-1}]\}\\
&= \underset{a\in\mathcal{A}_{L-1}}{\arg\min}\{a\left(g_X(M_{L-1}) - \mathbb{E}[g_X(M_L)|M_{L-1}]\right)\}.
\end{aligned}$$

Hence,

$$\begin{aligned}
&\begin{cases} g_X(M_{L-1}) > \mathbb{E}[g_X(M_L)|M_{L-1}] \Rightarrow \pi_{L-1}^* = 0\\ g_X(M_{L-1}) \le \mathbb{E}[g_X(M_L)|M_{L-1}] \Rightarrow \pi_{L-1}^* = A_{L-1}\end{cases}\\
&\Rightarrow \pi_{L-1}^* \in \{0, A_{L-1}\}.
\end{aligned} \tag{13}$$

We also have

$$\begin{aligned}
&\pi^*_{L-2}(M_{L-2}, I_{L-2}, X)\\
&= \arg\min_{a\in\mathcal{A}_{L-2}} \{C_X(M_{L-2}, a)\\
&\qquad + \mathbb{E}[V^*_{L-1}(M_{L-1}, I_{L-2}-a, X)|M_{L-2}]\}\\
&= \arg\min_{a\in\mathcal{A}_{L-2}} \{C_X(M_{L-2}, a)\\
&\qquad + \mathbb{E}[\pi^*_{L-1}(M_{L-1}, I_{L-2}-a, X)g_X(M_{L-1})|M_{L-2}]\\
&\qquad + \mathbb{E}[\pi^*_L g_X(M_L)|M_{L-2}]\}\\
&= \arg\min_{a\in\mathcal{A}_{L-2}} \{C_X(M_{L-2}, a)\\
&+ \mathbb{E}[\pi^*_{L-1}(M_{L-1}, I_{L-2}-a, X)g_X(M_{L-1})|M_{L-2}]\\
&+ \mathbb{E}[(I_{L-2}-a-\pi^*_{L-1}(M_{L-1}, I_{L-2}-a, X))g_X(M_L)|M_{L-2}]\}.
\end{aligned}$$

From (13), we know that π^*_{L-1} only depends on the market statistics. It does not depend on the inventory level, and hence, the action selected in time slot $L-2$. Therefore, we have

$$\begin{aligned}
&\pi^*_{L-2}(M_{L-2}, I_{L-2}, X)\\
&\quad= \arg\min_{a\in\mathcal{A}_{L-2}}\{a g_X(M_{L-2}) + \mathbb{E}[-a g_X(M_L)|M_{L-2}]\}\\
&\quad= \arg\min_{a\in\mathcal{A}_{L-2}}\{a(g_X(M_{L-2}) - \mathbb{E}[g_X(M_L)|M_{L-2}])\}.
\end{aligned}$$

Thus,

$$\begin{aligned}
&\begin{cases} g_X(M_{L-2}) > \mathbb{E}[g_X(M_L)|M_{L-2}] \Rightarrow \pi^*_{L-2} = 0\\ g_X(M_{L-2}) \le \mathbb{E}[g_X(M_L)|M_{L-2}] \Rightarrow \pi^*_{L-2} = A_{L-2}\end{cases}\\
&\Rightarrow \pi^*_{L-2} \in \{0, A_{L-2}\}.
\end{aligned}$$

Induction step:

Fix $l\in\{1,\ldots,L-3\}$. We will prove that if $\pi^*_{l+k}\in\{0, A_{l+k}\}$, $\forall k\in\{1,\ldots,L-l-1\}$, where π^*_{l+k}'s only depend on the market statistics, then $\pi^*_l\in\{0, A_l\}$. We have

$$\begin{aligned}
&\pi^*_l(M_l, I_l, X)\\
&\quad= \arg\min_{a\in\mathcal{A}_l}\{C_X(M_l, a) + \mathbb{E}[V^*_{l+1}(M_{l+1}, I_{l+1}, X)|M_l]\}\\
&\quad= \arg\min_{a\in\mathcal{A}_l}\Bigg\{C_X(M_l, a)\\
&\qquad\quad + \sum_{k=1}^{L-l}\mathbb{E}[C_X(M_{l+k}, \pi^*_{l+k}(M_{l+k}, I_{l+k}, X))|M_l]\Bigg\}\\
&\quad= \arg\min_{a\in\mathcal{A}_l}\Bigg\{a g_X(M_l) + \sum_{k=1}^{L-l-1}\mathbb{E}[\pi^*_{l+k} g_X(M_{l+k})|M_l]\\
&\qquad\quad + \mathbb{E}\left[\left(I_l - a - \sum_{k=1}^{L-l-1}\pi^*_{l+k}\right) g_X(M_L)\middle| M_l\right]\Bigg\}.
\end{aligned}$$

Since by the induction assumption π^*_{l+k}, $k\in\{1,\ldots,L-l-1\}$ only depends on the market statistics, they are all independent from a. Therefore, we have

$$\begin{aligned}
\pi^*_l(M_l, I_l, X) &= \arg\min_{a\in\mathcal{A}_l}\{a g_X(M_l) + \mathbb{E}[-a g_X(M_L)|M_l]\}\\
&= \arg\min_{a\in\mathcal{A}_l}\{a\,(g_X(M_l) - \mathbb{E}[g_X(M_L)|M_l])\}
\end{aligned}$$

from which we obtain

$$\begin{aligned}
&\begin{cases} g_X(M_l) > \mathbb{E}[g_X(M_L)|M_l] \Rightarrow \pi^*_l = 0\\ g_X(M_l) \le \mathbb{E}[g_X(M_L)|M_l] \Rightarrow \pi^*_l = A_l\end{cases}\\
&\Rightarrow \pi^*_l \in \{0, A_l\}.
\end{aligned}$$

This proves that $\pi^*_l\in\{0, A_l\}, \forall l\in\{1,\ldots,L-1\}$.

Appendix B
Proof of Lemma 1

The proof is done by induction. In the proof, we use the trivial fact that the implementation shortfall is finite.

Induction Basis:

$$\begin{aligned}
&Q^{\boldsymbol{\pi}}_{L-1}(M_{L-1}, I_{L-1}, X, a_{L-1}) = C_X(M_{L-1}, a_{L-1})\\
&\quad+ \sum_{M_L\in\mathcal{M}} P(M_{L-1}, M_L) C_X(M_L, I_{L-1}-a_{L-1})
\end{aligned}$$

which is a polynomial function of $P(M, M')$ with order at most 1, $\forall\boldsymbol{\pi}\in\Pi^{\text{opt}}$, $\forall M_{L-1}\in\mathcal{M}^r$, $\forall I_{L-1}\in\mathcal{I}$, $\forall X\in\mathcal{X}$, $\forall a_{L-1}\in\mathcal{A}_{L-1}$ and $\forall M\in\mathcal{M}^r$, $\forall M'\in\mathcal{M}$.

Induction Step:

Assume that $Q^{\boldsymbol{\pi}}_{l'}(M_{l'}, I_{l'}, X, a_{l'})$ is a polynomial function of $P(M, M')$ whose order is at most $L-l'$, $\forall\boldsymbol{\pi}\in\Pi^{\text{opt}}$, $\forall M_{l'}\in\mathcal{M}^r$, $\forall I_{l'}\in\mathcal{I}$, $\forall X\in\mathcal{X}$, $\forall a_{l'}\in\mathcal{A}_{l'}$, $\forall M\in\mathcal{M}^r$ and $\forall M'\in\mathcal{M}$, for all $l'\in\{l,\ldots,L-1\}$. Then, we show that $Q^{\boldsymbol{\pi}}_{l-1}(M_{l-1}, I_{l-1}, X, a_{l-1})$, $\forall\boldsymbol{\pi}\in\Pi^{\text{opt}}$, $\forall M_{l-1}\in\mathcal{M}^r$, $\forall I_{l-1}\in\mathcal{I}$, $\forall X\in\mathcal{X}$ and $\forall a_{l-1}\in\mathcal{A}_{l-1}$ is a polynomial function of $P(M, M')$, $\forall M\in\mathcal{M}^r$, $\forall M'\in\mathcal{M}$ where the order is at most $L-l+1$.

To see this, observe from (5) that

$$\begin{aligned}
&Q^{\boldsymbol{\pi}}_{l-1}(M_{l-1}, I_{l-1}, X, a_{l-1})\\
&\quad= C_X(M_{l-1}, a_{l-1}) + \sum_{M_l\in\mathcal{M}} P(M_{l-1}, M_l) V^{\boldsymbol{\pi}}_l(M_l, I_l, X).
\end{aligned}$$

Since $V^{\boldsymbol{\pi}}_l(M_l, I_l, X) = Q^{\boldsymbol{\pi}}_l(M_l, I_l, X, \pi_l(M_l, I_l, X))$, $V^{\boldsymbol{\pi}}_l(M_l, I_l, X)$ is a polynomial function of $P(M, M')$ where the order is at most $L-l$. This completes the proof.

Appendix C
Proof of Lemma 2

Let $f(x_1, x_2, \ldots, x_n)$ be a polynomial function of the variables $\{x_1, x_2, \ldots, x_n\}$. We are interested in upper bounding $|f(x_1, x_2, \ldots, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)|$ where $\hat{x}_i$ is the estimated value of the ith variable. We can rewrite this difference as sum of the differences of functions that differ only in one variable as follows:

$$\begin{aligned}
&|f(x_1, x_2, \ldots, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)| \\
&= |(f(x_1, x_2, \ldots, x_n) - f(\hat{x}_1, x_2, \ldots, x_n)) \\
&\quad + (f(\hat{x}_1, x_2, \ldots, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, x_n)) \\
&\quad + \ldots \\
&\quad + (f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{n-1}, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n))|.
\end{aligned}$$

Then, by using the triangle inequality we get

$$\begin{aligned}
&|f(x_1, x_2, \ldots, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)| \\
&\leq |(f(x_1, x_2, \ldots, x_n) - f(\hat{x}_1, x_2, \ldots, x_n))| \\
&\quad + |(f(\hat{x}_1, x_2, \ldots, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, x_n))| \\
&\quad + \ldots \\
&\quad + |(f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{n-1}, x_n) - f(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n))|. \quad (14)
\end{aligned}$$

If each absolute difference on the right-hand side of (14) is smaller than λ/n, then, the left-hand side is smaller than λ.

The variables of the polynomial that we consider in our problem are the state transition probabilities. First, we sort $P(M, M')$, $\forall M \in \mathcal{M}^r$, $\forall M' \in \mathcal{M}$ in an ascending way, and re-index them such that κ corresponds to the state-next state pair with the κth lowest $P(M, M')$ value,[16] where $\kappa \in \{1, \ldots, N^r\}$. We define J_κ as the state transition probability that corresponds to the κth state-next state pair. Let $\mathcal{J} := (J_1, \ldots, J_{N^r})$ denote the set of all J_κs, $\hat{J}_\kappa$ denote the estimate of J_κ used by GLOBE, and $\mathcal{J}_m := \{\hat{J}_1, \ldots, \hat{J}_m, J_{m+1}, \ldots, J_{N^r}\}$.

Let $\tilde{Q}^{\boldsymbol{\pi}}_{l,m}(M, I, X, a)$ be the estimate of $Q^{\boldsymbol{\pi}}_l(M, I, X, a)$ computed based on the set of state transition probabilities $\mathcal{J}_m$. Note that if $m = 0$, then $\tilde{Q}^{\boldsymbol{\pi}}_{l,m}(M, I, X, a) = Q^{\boldsymbol{\pi}}_l(M, I, X, a)$ and if $m = N^r$, then $\tilde{Q}^{\boldsymbol{\pi}}_{l,m}(M, I, X, a) = \hat{Q}^{\boldsymbol{\pi}}_l(M, I, X, a)$.

Next, we take $\tilde{Q}^{\boldsymbol{\pi}}_{l,m}(M, I, X, a)$ and write it as a function of the kth term in $\mathcal{J}_m$. Let $Z^{\boldsymbol{\pi},i,m}_\kappa(M, I, X, a, l)$ and $Z^{\boldsymbol{\pi},0:L-l,m}_\kappa(M, I, X, a, l)$ be the coefficient of the ith power of the κth term in $\mathcal{J}_m$ and the set of all coefficients of all the powers of the κth term in $\mathcal{J}_m$, respectively. Given that $|P(M, M') - \hat{P}(M, M')| \leq \gamma, \forall M \in \mathcal{M}^r, \forall M' \in \mathcal{M}$, we obtain the following for all $m \in \{1, \ldots, N^r\}$ using the result of Lemma 1:

$$\begin{aligned}
&|Q^{\boldsymbol{\pi}}_l(M_l, I_l, X, a_l) - \hat{Q}^{\boldsymbol{\pi}}_l(M_l, I_l, X, a_l)| \\
&\leq \sum_{m=1}^{N^r} |\tilde{Q}^{\boldsymbol{\pi}}_{l,m}(M_l, I_l, X, a_l) - \tilde{Q}^{\boldsymbol{\pi}}_{l,m-1}(M_l, I_l, X, a_l)| \\
&= \sum_{m=1}^{N^r} \left| \sum_{i=1}^{L-l} Z^{\boldsymbol{\pi},i,m}_m(M_l, I_l, X, a_l, l)((\hat{J}_m)^i - (J_m)^i) \right| \\
&= \sum_{m=1}^{N^r} \left| \sum_{i=1}^{L-l} \left(Z^{\boldsymbol{\pi},i,m}_m(M_l, I_l, X, a_l, l)(\hat{J}_m - J_m) \right.\right. \\
&\qquad \left.\left. \times \sum_{j=0}^{i-1} (\hat{J}_m)^j (J_m)^{i-j-1} \right) \right| \\
&\leq \sum_{m=1}^{N^r} \sum_{i=1}^{L-l} \left| Z^{\boldsymbol{\pi},i,m}_m(M_l, I_l, X, a_l, l)(\hat{J}_m - J_m) \right| \\
&\qquad \times \left(\sum_{j=0}^{i-1} (\hat{J}_m)^j (J_m)^{i-j-1} \right) \\
&\leq \sum_{m=1}^{N^r} \sum_{i=1}^{L-l} |Z^{\boldsymbol{\pi},i,m}_m(M_l, I_l, X, a_l, l)| \gamma i \\
&\leq \sum_{m=1}^{N^r} \gamma H = N^r H \gamma \qquad (15)
\end{aligned}$$

where we used $(\sum_{j=0}^{i-1} (\hat{J}_m)^j (J_m)^{i-j-1}) \leq i$.

According to (15), this implies that γ should be set to $\lambda/(N^r H)$ such that

$$|Q^{\boldsymbol{\pi}}_l(M_l, I_l, X, a_l) - \hat{Q}^{\boldsymbol{\pi}}_l(M_l, I_l, X, a_l)| \leq \lambda,$$

$\forall \boldsymbol{\pi} \in \Pi^{\text{opt}}, \forall X \in \mathcal{X}, \forall M_l \in \mathcal{M}^r, \forall I_l \in \mathcal{I}, \forall a_l \in \mathcal{A}_l, \forall l \in \mathcal{L} - \{L\}, \forall M \in \mathcal{M}^r$ and $\forall M' \in \mathcal{M}$.

Appendix D
Sub-Optimality Gap Bound

In order to simplify the notation, for a given (M, I, X), we use $\mu_{\boldsymbol{\pi}}$ and $\hat{\mu}_{\boldsymbol{\pi}}$ as the true and estimated V-value of policy $\boldsymbol{\pi} \in \Pi^{\text{opt}}$, respectively. Let $\boldsymbol{\pi}^* = \arg\min_{\boldsymbol{\pi} \in \Pi^{\text{opt}}} \mu_{\boldsymbol{\pi}}$ and $\hat{\boldsymbol{\pi}} = \arg\min_{\boldsymbol{\pi} \in \Pi^{\text{opt}}} \hat{\mu}_{\boldsymbol{\pi}}$. The trader selects the policy $\hat{\boldsymbol{\pi}}$ among the set of policies based on the estimated values. The suboptimality gap of the selected policy is bounded as follows:

$$\begin{aligned}
|\mu_{\boldsymbol{\pi}^*} - \mu_{\hat{\boldsymbol{\pi}}}| &= |\mu_{\boldsymbol{\pi}^*} - \mu_{\hat{\boldsymbol{\pi}}} - \hat{\mu}_{\hat{\boldsymbol{\pi}}} + \hat{\mu}_{\hat{\boldsymbol{\pi}}}| \leq |\mu_{\boldsymbol{\pi}^*} - \hat{\mu}_{\hat{\boldsymbol{\pi}}}| \\
&\quad + |\mu_{\hat{\boldsymbol{\pi}}} - \hat{\mu}_{\hat{\boldsymbol{\pi}}}|.
\end{aligned}$$

Next, assume that $|\mu_{\boldsymbol{\pi}} - \hat{\mu}_{\boldsymbol{\pi}}| \leq \lambda, \forall \boldsymbol{\pi} \in \Pi^{\text{opt}}$. Then, we have

$$\begin{aligned}
&\begin{cases} \mu_{\boldsymbol{\pi}^*} \leq \mu_{\hat{\boldsymbol{\pi}}} \Rightarrow \mu_{\boldsymbol{\pi}^*} - \hat{\mu}_{\hat{\boldsymbol{\pi}}} \leq \mu_{\hat{\boldsymbol{\pi}}} - \hat{\mu}_{\hat{\boldsymbol{\pi}}} \leq \lambda \\ \hat{\mu}_{\boldsymbol{\pi}^*} \geq \hat{\mu}_{\hat{\boldsymbol{\pi}}} \Rightarrow \mu_{\boldsymbol{\pi}^*} - \hat{\mu}_{\hat{\boldsymbol{\pi}}} \geq \mu_{\boldsymbol{\pi}^*} - \hat{\mu}_{\boldsymbol{\pi}^*} \geq -\lambda \end{cases} \\
&\Rightarrow |\mu_{\boldsymbol{\pi}^*} - \hat{\mu}_{\hat{\boldsymbol{\pi}}}| \leq \lambda \Rightarrow |\mu_{\boldsymbol{\pi}^*} - \mu_{\hat{\boldsymbol{\pi}}}| \leq 2\lambda.
\end{aligned}$$

Appendix E
Proof of Lemma 3

Let $m_\rho(M)$ be the indicator function of the event that state M is observed at least once in the first $L - 1$ time slots of round ρ, and $N'_{\rho+1}(M) := \sum_{i=1}^{\rho} m_i(M)$. Due to the definition of $m_\rho(M)$ and ϵ, for all $M \in \mathcal{M}^r$ we have

$$\mathbb{E}[N'_{\rho+1}(M)] = \mathbb{E}\left[\sum_{i=1}^{\rho} m_i(M)\right] = \sum_{i=1}^{\rho} \mathbb{E}\left[m_i(M)\right] \geq \epsilon\rho.$$

Using Hoeffding's inequality, we obtain

$$\begin{aligned}
&\mathbb{P}\left(N'_{\rho+1}(M) - \mathbb{E}[N'_{\rho+1}(M)] \leq -z\right) \leq e^{-2z^2/\rho} \\
&\Rightarrow \mathbb{P}\left(N'_{\rho+1}(M) \leq \epsilon\rho - z\right) \leq e^{-2z^2/\rho}.
\end{aligned}$$

We set $z = \sqrt{\rho \log \rho}$ and we obtain

$$\mathbb{P}\left(N'_{\rho+1}(M) \leq \epsilon\rho - \sqrt{\rho \log \rho}\right) \leq \frac{1}{\rho^2}.$$

[16] Ties can be broken arbitrarily.

For all $\rho \geq \rho'$, we have $\log \rho \leq 0.25\epsilon^2 \rho$ which results in $0.5\epsilon\rho \leq \epsilon\rho - \sqrt{\rho \log \rho}$. Therefore,

$$\mathbb{P}\left(N'_{\rho+1}(M) \leq 0.5\epsilon\rho\right) \leq \frac{1}{\rho^2} \text{ for } \rho \geq \rho' \Rightarrow$$

$$\mathbb{P}\left(N_{\rho+1}(M) \leq 0.5\epsilon\rho\right) \leq \frac{1}{\rho^2} \text{ for } \rho \geq \rho' \tag{16}$$

where in (16) we used the fact that $N_\rho(M) \geq N'_\rho(M)$.

Appendix F
Proof of Theorem 2

$$\mathbb{E}[\text{Reg}(R)] \leq \rho' \Delta_{\max}$$

$$+ \sum_{\rho=\rho'}^{R-1} \sum_{(M,M') \notin \mathcal{M}_0} \mathbb{P}\Big(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\Big) \Delta_{\max} \tag{17}$$

$$+ \sum_{\rho=\rho'}^{R-1} \sum_{(M,M') \in \mathcal{M}_0} \mathbb{P}\Big(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\Big) \Delta_{\max}. \tag{18}$$

We start by bounding (17). Let $f(\rho) := 0.5\epsilon\rho$. For all $\rho \geq \rho'$ and all $(M, M') \notin \mathcal{M}_0$, we have

$$\begin{aligned}
&\mathbb{P}\left(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\right) \\
&= \sum_{n=0}^{\infty} \mathbb{P}(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta | N_{\rho+1}(M) = n) \\
&\quad \times \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\leq \mathbb{P}\left(N_{\rho+1}(M) = 0\right) + \sum_{n=1}^{\infty} 2e^{-2n\delta^2} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\leq \sum_{n=0}^{\infty} 2e^{-2n\delta^2} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&= \sum_{n=0}^{\lceil f(\rho) \rceil - 1} 2e^{-2n\delta^2} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\quad + \sum_{n=\lceil f(\rho) \rceil}^{\infty} 2e^{-2n\delta^2} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\leq 2 \sum_{n=0}^{\lceil f(\rho) \rceil - 1} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\quad + 2e^{-2f(\rho)\delta^2} \sum_{n=\lceil f(\rho) \rceil}^{\infty} \mathbb{P}\left(N_{\rho+1}(M) = n\right) \\
&\leq 2\mathbb{P}\left(N_{\rho+1}(M) \leq f(\rho)\right) + 2e^{-\epsilon\delta^2\rho} \\
&\leq \frac{2}{\rho^2} + 2e^{-\epsilon\delta^2\rho}
\end{aligned} \tag{19}$$

where the first inequality results from Hoeffding's inequality and (19) results from Lemma 3. Using (19), we upper bound (17) as

$$\begin{aligned}
&\sum_{\rho=\rho'}^{R-1} \sum_{(M,M') \notin \mathcal{M}_0} \mathbb{P}\left(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\right) \Delta_{\max} \\
&\leq N^r \Delta_{\max} \left(\sum_{\rho=\rho'}^{R-1} \left(\frac{2}{\rho^2} + 2e^{-\epsilon\delta^2\rho} \right) \right) \\
&\leq N^r \Delta_{\max} \left(\sum_{\rho=1}^{\infty} \frac{2}{\rho^2} + \sum_{\rho=0}^{\infty} 2e^{-\epsilon\delta^2\rho} \right) \\
&\leq N^r \Delta_{\max} \left(\frac{\pi^2}{3} + \frac{2}{1 - e^{-\epsilon\delta^2}} \right) \\
&\leq N^r \Delta_{\max} \left(6 + \frac{2}{\epsilon\delta^2} \right)
\end{aligned} \tag{20}$$

where (20) follows from

$$1 - e^{-\epsilon\delta^2} \geq \frac{\epsilon\delta^2}{\epsilon\delta^2 + 1} \Rightarrow \frac{1}{1 - e^{-\epsilon\delta^2}} \leq 1 + \frac{1}{\epsilon\delta^2}$$

since $e^{-x} \leq 1/(1+x)$ for $x > 0$, and $2 + \pi^2/3 \leq 6$.

Next we bound (18). By Lemma 5, we have for all $(M, M') \in \mathcal{M}_0$

$$\begin{aligned}
&\sum_{\rho=\rho'}^{R-1} \mathbb{P}\left(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\right) \\
&\leq \sum_{\rho=\rho'}^{R-1} (1-\epsilon)^\rho \leq \sum_{\rho=0}^{\infty} (1-\epsilon)^\rho = \frac{1}{\epsilon}.
\end{aligned}$$

Hence, we obtain

$$\begin{aligned}
&\sum_{\rho=\rho'}^{R-1} \sum_{(M,M') \in \mathcal{M}_0} \mathbb{P}\left(|P(M,M') - \hat{P}_{\rho+1}(M,M')| \geq \delta\right) \Delta_{\max} \\
&\leq \frac{N^r}{\epsilon} \Delta_{\max}.
\end{aligned} \tag{21}$$

Finally, we combine the results in (20) and (21), and use the fact that $\epsilon \leq 1$ and $\delta \leq 1$ to get

$$\begin{aligned}
\mathbb{E}[\text{Reg}(R)] &\leq \left(\rho' + N^r \left(6 + \frac{2}{\epsilon\delta^2} \right) + \frac{N^r}{\epsilon} \right) \Delta_{\max} \\
&= \left(\rho' + N^r \left(\frac{6\epsilon\delta^2}{\epsilon\delta^2} + \frac{2}{\epsilon\delta^2} + \frac{\delta^2}{\epsilon\delta^2} \right) \right) \Delta_{\max} \\
&\leq \left(\rho' + N^r \left(\frac{9}{\epsilon\delta^2} \right) \right) \Delta_{\max}.
\end{aligned}$$

Appendix G
Linear-Exponential Equation

A) From the result in [42] we have

$$x = a_1 \log x + a_2 \Rightarrow x = e^{(x-a_2)/a_1} \Rightarrow e^{a_2/a_1} x = (e^{1/a_1})^x \Rightarrow$$

$$x = \frac{1}{\log e^{1/a_1}} \text{glog}\left(\frac{e^{a_2/a_1}}{\log e^{1/a_1}} \right) = a_1 \text{glog}\left(a_1 e^{a_2/a_1} \right). \tag{22}$$

According to the definition of $\text{glog}(y)$, given $y \geq e$, two solution exist for $\text{glog}(y)$. The larger one is called $\text{glog}_+(y)$ and

TABLE IV
STATISTICS OF THE DATASETS. THE ABBREVIATIONS ARE DEFINED AS FOLLOWS, SD: STARTING DATE, ST: STARTING TIME, TH: TIME HORIZON IN HOURS, NS: NUMBER OF SAMPLES, ATD: AVERAGE TIME DIFFERENCE BETWEEN SAMPLES IN SECONDS, MID: MID PRICE, BAS: BID-ASK SPREAD, VOL: TRADING VOLUME IN THE MARKET (SUM OF BID AND ASK VOLUMES)

Info / Dataset	AAPL	AMZN	GOOG
SD	2012/06/21	2012/06/21	2012/06/21
ST	9:30 AM	9:30 AM	9:30 AM
TH	6.5	6.5	6.5
NS	118497	57515	49482
ATD	0.19	0.41	0.47
mean MID	583.14	222.70	570.64
std MID	2.99	1.36	4.28
mean BAS	0.08	0.07	0.16
std BAS	0.04	0.03	0.09
mean VOL	374.39	394.71	307.09
std VOL	891.78	218.95	293.02

Info / Dataset	INTC	MSFT	EDC
SD	2012/06/21	2012/06/21	2008/10/26
ST	9:30 AM	9:30 AM	10:00 PM
TH	6.5	6.5	90.75
NS	404986	411409	1048282
ATD	0.06	0.06	0.31
mean MID	27.05	30.55	97.34
std MID	0.27	0.29	0.24
mean BAS	0.01	0.01	0.003
std BAS	0.002	0.002	0.001
mean VOL	31554.99	30761.64	432.44
std VOL	21356.89	18375.83	323.01

the smaller one is called glog$_-(y)$. As we are interested in upper bounding x in (22), we use the bound given in [42] for the larger value which holds for $e \leq y$ and $k \geq 1$: glog$_+(y)$ $\leq (y(\frac{k}{e})^k)^{\frac{1}{k-1}}$. We set $k = 3$ and $y = a_1 e^{a_2/a_1}$. Then, we get

$$x \leq a_1 \left(a_1 e^{a_2/a_1} \left(\frac{3}{e}\right)^3 \right)^{0.5} = a_1^{1.5} e^{a_2/2a_1} \left(\frac{3}{e}\right)^{1.5}$$
$$= \left(\frac{3a_1}{e}\right)^{1.5} e^{a_2/2a_1}.$$

B) Let $v \in (0,1)$ and $u := x^v$. Then, for $x = a_3 x^{1-v} \log x$, we have

$$x = a_3 x^{1-v} \log x \Rightarrow x^v = \frac{a_3}{v} \log x^v \Rightarrow u = \frac{a_3}{v} \log u.$$

Then according to part (A), we obtain

$$x^v = u \leq \left(\frac{3a_3}{ev}\right)^{1.5} \Rightarrow x \leq \left(\frac{3a_3}{ev}\right)^{1.5/v}.$$

APPENDIX H
INFORMATION AND STATISTICS OF THE STOCKS

Table IV illustrates some properties of the stocks used in Section VII.

APPENDIX I
ACPR OF THE ALGORITHMS

Table V illustrates the ACPR of the algorithms calculated over the test sets discussed in Section VII. Substituting these values in the formula given for RI may not give the exact values reported in Table III since the values given in Table V are truncated.

TABLE V
ACPR OF ALL ALGORITHMS CALCULATED OVER THE TEST SETS. ALL VALUES ARE MULTIPLIED BY 10

Algorithms / Dataset	AAPL	AMZN	GOOG	INTC	MSFT	EDC
AC	**1.308**	2.710	2.325	2.146	1.924	0.307
EQ	1.328	2.702	2.329	2.226	1.939	0.307
Q-MAT	1.412	2.855	2.504	2.441	2.119	0.306
Q-EXP	1.387	2.790	2.464	2.357	2.027	0.307
GLOBE	1.326	2.703	2.343	2.103	1.904	0.303
C-GLOBE	1.340	2.749	2.343	2.102	1.856	0.304
Q-MAT+	1.392	2.847	2.435	2.423	2.138	0.307
Q-EXP+	1.326	2.719	2.350	2.220	1.939	0.306
GLOBE+	**1.250**	**2.604**	**2.180**	**1.976**	**1.812**	**0.302**
C-GLOBE+	1.382	**2.675**	**2.292**	**1.998**	**1.816**	**0.302**

ACKNOWLEDGMENT

The authors would like to thank T. Flury from MAN AHL for his guidance and feedback, and MAN AHL for sharing the EDC dataset.

REFERENCES

[1] N. Akbarzadeh, C. Tekin, and M. van der Schaar, “Online learning in limit order book trade execution,” in *Proc. IEEE Global Conf. Signal Inf. Process.*, 2017, pp. 898–902.

[2] Y. Feng and D. P. Palomar, *A Signal Processing Perspective on Financial Engineering* (Foundations and Trends in Signal Processing, vol. 9. Breda, The Netherlands: Now Publishers, 2016.

[3] Y. Feng, D. P. Palomar, and F. Rubio, “Robust optimization of order execution,” *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 907–920, Feb. 2015.

[4] D. Palguna and I. Pollak, “Non-parametric prediction in a limit order book,” in *Proc. IEEE Global Conf. Signal Inf. Process.*, 2013, pp. 1139–1139.

[5] A. N. Akansu, D. Malioutov, D. P. Palomar, E. Jay, and D. P. Mandic, “Introduction to the issue on financial signal processing and machine learning for electronic trading,” *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 979–981, Sep. 2016.

[6] G. Rosenberg, P. Haghnegahdar, P. Goddard, P. Carr, K. Wu, and M. L. de Prado, “Solving the optimal trading trajectory problem using a quantum annealer,” *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 1053–1060, Sep. 2016.

[7] R. Almgren and N. Chriss, “Optimal execution of portfolio transactions,” *J. Risk*, vol. 3, pp. 5–40, 2001.

[8] D. Hendricks and D. Wilcox, “A reinforcement learning extension to the Almgren-Chriss framework for optimal trade execution,” in *Proc. IEEE Conf. Comput. Intell. for Financial Eng. Econ.*, 2014, pp. 457–464.

[9] M. D. Gould, M. A. Porter, S. Williams, M. McDonald, D. J Fenn, and S. D. Howison, “Limit order books,” *Quantitative Finance*, vol. 13, no. 11, pp. 1709–1742, 2013.

[10] D. Bertsimas and A. W. Lo, “Optimal control of execution costs,” *J. Financial Markets*, vol. 1, no. 1, pp. 1–50, 1998.

[11] A. A. Sherstov and P. Stone, “Three automated stock-trading agents: A comparative study,” in *Proc. Int. Workshop Agent-Mediated Electron. Commerce*, 2004, pp. 173–187.

[12] Y. Nevmyvaka, Y. Feng, and M. Kearns, “Reinforcement learning for optimized trade execution,” in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 673–680.

[13] P. Auer and R. Ortner, “Logarithmic online regret bounds for undiscounted reinforcement learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 49–56.

[14] A. Tewari and P. Bartlett, “Optimistic linear programming gives logarithmic regret for irreducible MDPs,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1505–1512.

[15] P. Auer, T. Jaksch, and R. Ortner, "Near-optimal regret bounds for reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 89–96.

[16] R. Cont and A. De Larrard, "Price dynamics in a Markovian limit order market," *SIAM J. Financial Math.*, vol. 4, no. 1, pp. 1–25, 2013.

[17] R. Cont, S. Stoikov, and R. Talreja, "A stochastic model for order book dynamics," *Operations Res.*, vol. 58, no. 3, pp. 549–563, 2010.

[18] W. Huang, C. A. Lehalle, and M. Rosenbaum, "Simulating and analyzing order book data: The queue-reactive model," *J. Amer. Statist. Assoc.*, vol. 110, no. 509, pp. 107–122, 2015.

[19] D. Palguna and I. Pollak, "Mid-price prediction in a limit order book," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 1083–1092, Sep. 2016.

[20] C. Tekin and M. van der Schaar, "RELEAF: An algorithm for learning and exploiting relevance," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 716–727, Jun. 2015.

[21] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.

[22] G. Neu, A. Gyorgy, C. Szepesvari, and A. Antos, "Online Markov decision processes under bandit feedback," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 676–691, Mar. 2014.

[23] P. Guan, M. Raginsky, and R. M. Willett, "Online Markov decision processes with Kullback–Leibler control cost," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1423–1438, Jun. 2014.

[24] O. Dekel and E. Hazan, "Better rates for any adversarial deterministic MDP," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, vol. 28, pp. 675–683.

[25] A. Zimin and G. Neu, "Online learning in episodic Markovian decision processes by relative entropy policy search," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 1583–1591, 2013.

[26] C. Tekin and M. van der Schaar, "Episodic multi-armed bandits," arXiv preprint arXiv:1508.00641, 2015.

[27] V. F. Farias, C. C. Moallemi, B. Van Roy, and T. Weissman, "Universal reinforcement learning," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2441–2454, May 2010.

[28] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, vol. 1. Cambridge, U.K.: MIT Press, 1998.

[29] S. Vakili and Q. Zhao, "Risk-averse multi-armed bandit problems under mean-variance measure," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 1093–1111, Sep. 2016.

[30] R. Ortner, "Online regret bounds for Markov decision processes with deterministic transitions," in *Proc. 19th Int. Conf. Algorithmic Learn. Theory*, 2008, pp. 123–137.

[31] A. J. Mersereau, P. Rusmevichientong, and J. N. Tsitsiklis, "A structured multiarmed bandit problem and the greedy policy," *IEEE Trans. Autom. Control*, vol. 54, no. 12, pp. 2787–2802, Dec. 2009.

[32] O. Atan, C. Tekin, and M. van der Schaar, "Global multi-armed bandits with Hölder continuity," in *Proc. 18th Int. Conf. Artif. Intell. Statist.*, 2015, pp. 28–36.

[33] N. Akbarzadeh and C. Tekin, "Gambler's ruin bandit problem," in *Proc. 54th Ann. Allerton Conf. Commun., Control, Comput.*, 2016, pp. 1236–1243.

[34] A. O. Saritac and C. Tekin, "Combinatorial multi-armed bandits with probabilistically triggered arms: A case with bounded regret," in *Proc. 5th IEEE Global Conf. Signal Inf. Process.*, 2017, pp. 111–115.

[35] A. F. Perold, "The implementation shortfall: Paper versus reality," *J. Portfolio Manage.*, vol. 14, no. 3, pp. 4–9, 1988.

[36] R. F. Almgren, "Optimal execution with nonlinear impact functions and trading-enhanced risk," *Appl. Math. Finance*, vol. 10, no. 1, pp. 1–18, 2003.

[37] R. Bellman, "Dynamic programming and stochastic control processes," *Inf. Control*, vol. 1, no. 3, pp. 228–239, 1958.

[38] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1. Belmont, MA, USA: Athena Scientific, 1995.

[39] F. A. Tobar, M. E. Orchard, D. P. Mandic, and A. G. Constantinides, "Estimation of financial indices volatility using a model with time-varying parameters," in *Proc. IEEE Conf. Comput. Intell. Financial Eng. Econ.*, 2014, pp. 318–324.

[40] J. Han, X. P. Zhang, and F. Wang, "Gaussian process regression stochastic volatility model for financial time series," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 6, pp. 1015–1028, Sep. 2016.

[41] A. M. Davie and A. J. Stothers, "Improved bound for complexity of matrix multiplication," *Proc. Roy. Soc. Edinburgh Sec. A: Math.*, vol. 143, no. 2, pp. 351–369, 2013.

[42] D. Kalman, "A generalized logarithm for exponential-linear equations," *The College Math. J.*, vol. 32, pp. 2–14, 2001.

Nima Akbarzadeh (S'17) received the B.Sc. degree in electrical and computer engineering from Shiraz University, Shiraz, Iran, in 2014, the M.Sc. in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2017. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, McGill University, Montreal, QC, Canada. His research interests include reinforcement learning, recommender systems, planning, and stochastic optimization. He is a member of the McGill Center of Intelligent Machines (CIM).

Cem Tekin (M'13) received the B.Sc. degree in electrical and electronics engineering from the Middle East Technical University, Ankara, Turkey, in 2008, and the M.S.E. degree in electrical engineering: systems, the M.S. degree in mathematics, and the Ph.D. degree in electrical engineering: systems from the University of Michigan, Ann Arbor, MI, USA, in 2010, 2011, and 2013, respectively. He is currently an Assistant Professor with the Electrical and Electronics Engineering Department, Bilkent University, Ankara, Turkey. From February 2013 to January 2015, he was a Postdoctoral Scholar with the University of California, Los Angeles. His research interests include reinforcement learning, multi-armed bandit problems, data mining, multi-agent systems, cognitive radio networks, and smart healthcare. He received the University of Michigan Electrical Engineering Departmental Fellowship in 2008, and the Fred W. Ellersick award for the best paper in MILCOM 2009.

Mihaela van der Schaar (F'10) is Man Professor with the University of Oxford, a Fellow of Christ Church College, and Turing Faculty Fellow of The Alan Turing Institute. She holds 33 granted USA patents. Her current research focuses on machine learning and data science. She has also been the recipient of an NSF Career Award, three IBM Faculty Awards, the IBM Exploratory Stream Analytics Innovation Award, the Philips Make a Difference Award, and several best paper awards, including the IEEE Darlington Award.