Competitive Algorithms for VWAP and Limit Order Trading

Sham M. Kakade
Computer and Information Science
University of Pennsylvania
kakade@linc.cis.upenn.edu

Michael Kearns
Computer and Information Science
University of Pennsylvania
mkearns@cis.upenn.edu

Yishay Mansour
Computer Science
Tel Aviv University
mansour@post.tau.ac.il

Luis E. Ortiz
Computer and Information Science
University of Pennsylvania
leortiz@linc.cis.upenn.edu

ABSTRACT

We introduce new online models for two important aspects of modern financial markets: *Volume Weighted Average Price* trading and *limit order books*. We provide an extensive study of competitive algorithms in these models and relate them to earlier online algorithms for stock trading.

Categories and Subject Descriptors

F.2 [**Analysis of Algorithms and Problem Complexity**]: Miscellaneous; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Algorithms, Economics

Keywords

Online Trading, Competitive Analysis, VWAP

1. INTRODUCTION

While popular images of Wall Street often depict swashbuckling traders boldly making large gambles on just their market intuitions, the vast majority of trading is actually considerably more technical and constrained. The constraints often derive from a complex combination of business, regulatory and institutional issues, and result in certain kinds of “standard” trading strategies or criteria that invite algorithmic analysis.

One of the most common activities in modern financial markets is known as *Volume Weighted Average Price*, or VWAP, trading. Informally, the VWAP of a stock over a specified market period is simply the average price paid *per share* during that period, so the price of each transaction in the market is weighted by its volume. In VWAP trading, one attempts to buy or sell a fixed number of shares at a price that closely tracks the VWAP.

Very large institutional trades constitute one of the main motivations behind VWAP activity. A typical scenario goes as follows. Suppose a very large mutual fund holds 3% of the outstanding shares of a large, publicly traded company — a huge fraction of the shares — and that this fund's manager decides he would like to reduce this holding to 2% over a 1-month period. (Such a decision might be forced by the fund's own regulations or other considerations.) Typically, such a fund manager would be unqualified to sell such a large number of shares in the open market — it requires a professional broker to intelligently break the trade up over time, and possibly over multiple exchanges, in order to minimize the market impact of such a sizable transaction. Thus, the fund manager would approach brokerages for help in selling the 1%.

The brokerage will typically alleviate the fund manager's problem immediately by simply buying the shares directly from the fund manager, and then selling them off later — but what price should the brokerage pay the fund manager? Paying the price on the day of the sale is too risky for the brokerage, as they need to sell the shares themselves over an extended period, and events beyond their control (such as wars) could cause the price to fall dramatically. The usual answer is that the brokerage offers to buy the shares from the fund manager at a per-share price tied to the VWAP over some future period — in our example, the brokerage might offer to buy the 1% at a per-share price of the coming month's VWAP minus 1 cent. The brokerage now has a very clean challenge: by selling the shares themselves over the next month in a way that exactly matches the VWAP, a penny per share is earned in profits. If they can beat the VWAP by a penny, they make two cents per share. Such small-margin, high-volume profits can be extremely lucrative for a large brokerage. The importance of the VWAP has led to many automated VWAP trading algorithms — indeed, every major brokerage has at least one ”VWAP box”,

EC'04, May 17–20, 2004, New York, New York, USA.

	Price Volume Model	Order Book Model	Macroscopic Distribution Model
OWT	$\Theta(\log(R))$ (From[3])	$O(\log(R)\log(N))$	$2\mathcal{E}(P^{\text{bins}}_{maxprice})$ $2(1+\epsilon)\mathcal{E}(P^{\text{bins}}_{maxprice})$ for ϵ-approx of $P^{\text{bins}}_{maxprice}$
VWAP	$\Theta(\log(Q))$ $\Theta(\log(R))$ $\Omega(Q)$ fixed schedule 1 for volume in $[N, QN]$	$O(\log(R)\log(N))$ (from above) $O(\log(Q))$ for large N	(same as above plus...) $2\mathcal{E}(P^{\text{bins}}_{vol})$ $(1+\epsilon)2\mathcal{E}(P^{\text{bins}}_{vol})$ for ϵ-approx. of P^{bins}_{vol}

Figure 1: The table summarizes the results presented in this paper. The rows represent results for either the OWT or VWAP criterion. The columns represent which model we are working in. The entry in the table is the competitive ratio between our algorithm and an optimal algorithm, and the closer the ratio is to 1 the better. The parameter R represents a bound on the maximum to the minimum price fluctuation and the parameter Q represents a bound on the maximum to minimum volume fluctuation in the respective model. (See Section 4 for a description of the Macroscopic Distribution Model.) All the results for the OWT trading criterion (which is a stronger criterion) directly translate to the VWAP criterion. However, in the VWAP setting, considering a restriction on the maximum to the minimum volume fluctuation Q, leads to an additional class of results which depends on Q.

and some small companies focus exclusively on proprietary VWAP trading technology.

In this paper, we provide the first study of VWAP trading algorithms in an online, competitive ratio setting. We first formalize the VWAP trading problem in a basic online model we call the *price-volume* model, which can be viewed as a generalization of previous theoretical online trading models incorporating market volume information. In this model, we provide VWAP algorithms and competitive ratios, and compare this setting with the *one-way trading (OWT)* problem studied in [3].

Our most interesting results, however, examine the VWAP trading problem in a new online trading model capturing the important recent phenomenon of *limit order books* in financial markets. Briefly, a limit buy or sell order specifies both the number of shares *and* the desired price, and will only be executed if there is a matching party on the opposing side, according to a well-defined matching procedure used by all the major exchanges. While limit order books (the list of limit orders awaiting possible future execution) have existed since the dawn of equity exchanges, only very recently have these books become *visible to traders* in real time, thus opening the way to trading algorithms of all varieties that attempt to exploit this rich *market microstructure* data. Such data and algorithms are a topic of great current interest on Wall Street [4].

We thus introduce a new online trading model incorporating limit order books, and examine both the one-way and VWAP trading problems in it. Our results are summarized in Figure 1 (see the caption for a summary).

2. THE PRICE-VOLUME TRADING MODEL

We now present a trading model which includes both price and volume information about the sequence of trades. While this model is a generalization of previous formalisms for online trading, it makes an *infinite liquidity* assumption which fails to model the negative *market impact* that trading a large number of shares typically has. This will be addressed in the order book model studied in the next section.

A note on terminology: throughout the paper (unless otherwise specified), we shall use the term "market" to describe all activity or orders *other* than those of the algorithm under consideration. The setting we consider can be viewed as a game between our algorithm and the market.

2.1 The Model

In the *price-volume trading model*, we assume that the *intraday* trading activity in a given stock is summarized by a discrete sequence of price and volume pairs (p_t, v_t) for $t = 1, \ldots, T$. Here $t = 0$ corresponds to the day's market open, and $t = T$ to the close. While there is nothing technically special about the time horizon of a single day, it is particularly consistent with limit order book trading on Wall Street. The pair (p_t, v_t) represents the fact that a total of v_t shares were traded at an (average) price per share p_t in the market between time $t-1$ and t. Realistically, we should imagine the number of intervals T being reasonably large, so that it is sensible to assign a common approximate price to all shares traded within an interval.

In the price-volume model, we shall make an *infinite liquidity* assumption for our trading algorithms. More precisely, in this online model, we see the price-volume sequence one pair at a time. Following the observation of (p_t, v_t), we are permitted to sell any (possibly fractional) number of shares n_t *at the price* p_t. Let us assume that our goal is to *sell* N shares over the course of the day. Hence, at each time, we must select a (possibly fractional) number of shares n_t to sell at price p_t, subject to the global constraint $\sum_{t=1}^{T} n_t = N$. It is thus assumed that if we have "left over" shares to sell after time $T-1$, we are forced to sell them at the closing price of the market — that is, $n_T = N - \sum_{t=1}^{T-1} n_t$ is sold at p_T. In this way we are certain to sell exactly N shares over the course of the day; the only thing an algorithm must do is determine the schedule of selling based on the incoming market price-volume stream.

Any algorithm which sells fractional volumes can be converted to a randomized algorithm which only sells integral volumes with the same *expected* number of shares sold. If we keep the hard constraint of selling exactly N shares, we might incur an additional slight loss in the conversion. (Note that we only allow fractional volumes in the price-volume model, where liquidity is not an issue. In the order book model to follow, we do not allow fractional volumes.)

In VWAP trading, the goal of an online algorithm A which sells exactly N shares is not to maximize profits per se, but to track the market VWAP. The market VWAP for an intraday trading sequence $S = (p_1, v_1), \ldots, (p_T, v_T)$ is simply the average price paid *per share* over the course of the trading day, *ie*

$$\mathrm{VWAP}_M(S) = \left(\sum_{t=1}^{T} p_t v_t\right)/V$$

where V is the total daily volume, i.e., $V = \sum_{t=1}^{T} v_t$. If on the sequence S, the algorithm A sells its N stocks using the volume sequence $n_1, \ldots n_T$, then we analogously define the VWAP of A on market sequence S by

$$\mathrm{VWAP}_A(S) = \left(\sum_{t=1}^{T} p_t n_t\right)/N\,.$$

Note that the market VWAP does not include the shares that the algorithm sells.

The *VWAP competitive ratio of* A with respect to a set of sequences Σ is then

$$R_{\mathrm{VWAP}}(A) = \max_{S\in\Sigma}\{\mathrm{VWAP}_M(S)/\mathrm{VWAP}_A(S)\}$$

In the case that A is randomized, we generalize the definition above by taking an expectation over $\mathrm{VWAP}_A(S)$ inside the max. We note that unlike on Wall Street, our definition of VWAP_M does not take our own trading into account. It is easy to see that this makes it a more challenging criterion to track.

In contrast to the VWAP, another common measure of the performance of an online selling algorithm would be its *one-way trading (OWT) competitive ratio* [3] with respect to a set of sequences Σ:

$$R_{\mathrm{OWT}}(A) = \max_{S\in\Sigma}\max_{1\le t\le T}\{p_t/\mathrm{VWAP}_A(S)\}$$

where the algorithms performance is compared to the largest individual *price* appearing in the sequence S.

In both VWAP and OWT, we are comparing the average price per share received by a selling algorithm to some measure of market performance. In the case of OWT, we compare to the rather ambitious benchmark of the high price of the day, ignoring volumes entirely. In VWAP trading, we have the more modest goal of comparing favorably to the overall market average of the day. As we shall see, there are some important commonalities and differences to these two approaches. For now we note one simple fact: on any *specific* sequence S, $\mathrm{VWAP}_A(S)$ may be *larger* that $\mathrm{VWAP}_M(S)$. However, $R_{\mathrm{VWAP}}(A)$ cannot be smaller than 1, since on any sequence S in which all price p_t are identical, it is impossible to get a better average share per price. Thus, for all algorithms A, both $R_{\mathrm{VWAP}}(A)$ and $R_{\mathrm{OWT}}(A)$ are larger than 1, and the closer to 1 they are, the better A is tracking its respective performance measure.

2.2 VWAP Results in the Price-Volume Model

As in previous work on online trading, it is generally not possible to obtain finite bounds on competitive ratios with absolutely no assumptions on the set of sequences Σ — bounds on the maximum variation in price *or* volume are required, depending on the exact setting. We thus introduce the following two assumptions.

2.2.0.1 Volume Variability Assumption..

Let $0 < V_{min} \le V_{max}$ be known positive constants, and define $Q = V_{max}/V_{min}$. For all intraday trading sequences $S \in \Sigma$, the total daily volume $V \in [V_{min}, V_{max}]$.

2.2.0.2 Price Variability Assumption..

Let $0 < p_{\min} \le p_{\max}$ be known positive constants, and define $R = p_{\max}/p_{\min}$. For all intraday trading sequences $S \in \Sigma$, the prices satisfy $p_t \in [p_{\min}, p_{\max}]$, for all $t = 1, \ldots, T$.

Competitive ratios are generally taken over all sets Σ consistent with at least one of these assumptions. To gain some intuition consider the two trivial cases of $R = 1$ and $Q = 1$. In the case of $R = 1$ (where there is no fluctuation in price), any schedule is optimal. In the case of $Q = 1$ (where the total volume V over the trading period is known), we can gain a competitive ratio of 1 by selling $\frac{v_t}{V}N$ shares after each time period.

For the OWT problem in the price-volume model, volumes are irrelevant for the performance criterion, but for the VWAP criterion they are central. For the OWT problem under the price variability assumption, the results of [3] established that the optimal competitive ratio was $\Theta(\log(R))$. Our first result establishes that the optimal competitive ratio for VWAP under the volume variability assumption is $\Theta(\log(Q))$ and is achieved by an algorithm that ignores the price data.

THEOREM 1. *In the price-volume model under the volume variability assumption, there exists an online algorithm A for selling N shares achieving competitive ratio $R_{\mathrm{VWAP}}(A) \le 2\log(Q)$. In addition, if only the volume variability (and not the price variability) assumption holds, any online algorithm A for selling N shares has $R_{\mathrm{VWAP}}(A) = \Omega(\log(Q))$.*

PROOF. (Sketch) For the upper bound, the idea is similar to the *price reservation* algorithm of [3] for the OWT problem, and similar in spirit to the general technique of *classify and select* [1]. Consider algorithms which use a parameter $\hat{V}$, which is interpreted as an estimate for the total volume for the day. Then at each time t, if the market price and volume is (p_t, v_t), the algorithm sells a fraction $v_t/\hat{V}$ of its shares. We consider a family of $\log(Q)$ such algorithms, where algorithm A_i uses $\hat{V} = V_{min}2^{i-1}$. Clearly, one of the A_i has a competitive ratio of 2. We can derive an $O(\log(Q))$ VWAP competitive ratio by running these algorithms in parallel, and letting each algorithm sell $N/\log(Q)$ shares. (Alternatively, we can randomly select one A_i and guarantee the same expected competitive ratio.)

We now sketch the proof of the lower bound, which relates performance in the VWAP and OWT problems. Any algorithm that is c-competitive in the VWAP setting (under fixed Q) is $3c$-competitive in the OWT setting with $R = Q/2$. To show this, we take any sequence S of prices for the OWT problem, and convert it into a price-volume sequence for the VWAP problem. The prices in the VWAP sequence are the same as in S. To construct the volumes in the VWAP sequence, we segment the prices in S into $\log(R)$ intervals $[2^{i-1}p_{min}, 2^i p_{min})$. Suppose $p_t \in [2^{i-1}p_{min}, 2^i p_{min})$, and this is the first time in S that a price has fallen in this interval. Then in the VWAP sequence we set the volume $v_t = 2^{i-1}$. If this is not the first visit to the interval containing p_t, we set $v_t = 0$. Assume that the maximum price in S is p_{max}. The VWAP of our sequence is at least $p_{max}/3$. Since we had a c competitive algorithm, its average sell is at least $p_{max}/3c$. The lower bound now follows using the lower bound in [3]. □

An alternative approach to VWAP is to ignore the volumes in favor of prices, and apply an algorithm for the OWT problem. Note that the lower bound in this theorem, unlike in the previous one, only assumes a price variation bound.

Theorem 2. *In the price-volume model under the price variability assumption, there exists an online algorithm A for selling N shares achieving competitive ratio $R_{\mathrm{VWAP}}(A) = O(\log(R))$. In addition, if only the price variability (and not the volume variability) assumption holds, any online A for selling N shares has $R_{\mathrm{VWAP}}(A) = \Omega(\log(R))$.*

Proof. (Sketch) Follows immediately from the results of [3] for OWT: the upper bound from the simple fact that for any sequence S, $\mathrm{VWAP}_A(S)$ is less than $\max_{1 \le t \le T}\{p_t\}$, and the lower bound from a reduction to OWT. $\square$

Theorems 1 and 2 demonstrate that one can achieve logarithmic VWAP competitive ratios under the assumption of *either* bounded variability of total volume or bounded variability of maximum price. If both assumptions hold, it is possible to give an algorithm accomplishing the minimum of $\log(Q)$ and $\log(R)$. This "flexibility" of approach derives from the fact that the VWAP is a quantity in which both prices and volumes matter, as opposed to OWT.

2.3 Related Results in the Price-Volume Model

All of the VWAP algorithms we have discussed so far make some use of the daily data (p_t, v_t) as it unfolds, using either the price or volume information. In contrast, a *fixed schedule* VWAP algorithm has a predetermined distribution $\{f_1, f_2, \ldots f_T\}$, and simply sells $f_t N$ shares at time t, independent of (p_t, v_t). Fixed schedule VWAP algorithms, or slight variants of them, are surprisingly common on Wall Street, and the schedule is usually derived from historical intraday volume data. Our next result demonstrates that such algorithms can perform considerably worse than dynamically adaptive algorithms in terms of the worst case competitive ratio.

Theorem 3. *In the price-volume model under both the volume and price variability assumptions, any fixed schedule VWAP algorithm A for selling N shares has sell VWAP competitive ratio $R_{\mathrm{VWAP}}(A) = \Omega(\min(T, R))$.*

The proofs of all the results in this subsection are in the Appendix.

So far our emphasis has been on VWAP algorithms that must sell *exactly* N shares. In many realistic circumstances, however, there is actually some flexibility in the precise number of shares to be sold. For instance, this is true at large brokerages, where many separate VWAP trades may be pooled and executed by a common algorithm, and the firm would be quite willing to carry a small position of unsold shares overnight if it resulted in better execution prices. The following theorem (which interestingly has no analogue for the OWT problem) demonstrates that this trade-off in shares sold and performance can be realized dramatically in our model. It states that if we are willing to let the number of shares sold vary with Q, we can in fact achieve a VWAP competitive ratio of 1.

Theorem 4. *In the price-volume model under the volume variability assumption, there exists an algorithm A that always sells between N and QN shares and that the average price per sold share is exactly $\mathrm{VWAP}_M(S)$.*

In many online problems, there is a clear distinction between "benefit" problems and "cost" problems [2]. In the VWAP setting, selling shares is a benefit problem, and buying shares is a cost problem. The definitions of the competitive ratios, $R^{buy}_{\mathrm{VWAP}}(A)$ and $R^{buy}_{\mathrm{OWT}}(A)$, for algorithms which

LAST MATCH		TODAY'S ACTIVITY	
Price	24.0700	Orders	52,983
Time	14:57:07.72	Volume	10,243,212

BUY ORDERS		SELL ORDERS	
SHARES	PRICE	SHARES	PRICE
500	24.0620	500	24.0690
6,000	24.0610	500	24.0690
5,000	24.0600	500	24.0700
100	24.0600	200	24.0800
1,100	24.0550	1,981	24.0900
100	24.0500	412	24.0900
5,000	24.0500	3,000	24.0980
200	24.0500	500	24.1000
3,294	24.0500	100	24.1200
1,000	24.0500	2,800	24.1400
3,000	24.0430	5,000	24.1400
100	24.0400	1,000	24.1400
5,503	24.0400	5,000	24.1500
2,100	24.0300	400	24.1600
2,800	24.0300	1,000	24.1700
(412 more)		(694 more)	

As of 14:57:16.178

Figure 2: Sample Island order books for MSFT.

buy exactly N shares are $\max_{S \in \Sigma}\{\mathrm{VWAP}_A(S)/\mathrm{VWAP}_M(S)\}$ and $\max_{S \in \Sigma} \max_t\{\mathrm{VWAP}_A(S)/p_t\}$ respectively. Eventhough Theorem 4 also holds for buying, in general, the competitive ratio of the buy (cost) problem is much higher, as stated in the following theorem.

Theorem 5. *In the price-volume model under the volume and price variability assumptions, there exists an online algorithm A for buying N shares achieving buy VWAP competitive ratio $R^{buy}_{\mathrm{VWAP}}(A) = O(\min\{Q, \sqrt{R}\})$. In addition any online algorithm A for buying N shares has buy VWAP competitive ratio $R^{buy}_{\mathrm{VWAP}}(A) = \Omega(\min\{Q, \sqrt{R}\})$.*

3. A LIMIT ORDER BOOK TRADING MODEL

Before we can define our online trading model based on limit order books, we give some necessary background on the detailed mechanics of financial markets, which are sometimes referred to as *market microstructure*. We then provide results and algorithms for both the OWT and VWAP problems.

3.1 Background on Limit Order Books and Market Microstructure

A fundamental distinction in stock trading is that between a *limit order* and a *market order*. Suppose we wish to purchase 1000 shares of Microsoft (MSFT) stock. In a limit order, we specify not only the desired volume (1000 shares), but also the desired price. Suppose that MSFT is currently trading at roughly $24.07 a share (see Figure 2, which shows an actual snapshot of a recent MSFT order book on Island (www.island.com), a well-known electronic exchange for NASDAQ stocks), but we are only willing to buy the 1000 shares at $24.04 a share or lower. We can choose to submit a limit order with this specification, and our order will be placed in a queue called the *buy order book*, which is ordered by price, with the highest offered unexecuted buy price at the top (often referred to as the *bid*). If there are multiple limit orders at the same price, they are ordered by time of arrival (with older orders higher in the book). In the example provided by Figure 2, our order would be placed immediately after the extant order for 5,503 shares at $24.04; though we offer the same price, this order has arrived before ours. Similarly, a sell order book for sell limit orders (for instance, we might want to sell 500 shares of MSFT at $24.10 or higher) is maintained, this time with the lowest sell price offered (often referred to as the *ask*).

Thus, the order books are sorted from the most competitive limit orders at the top (high buy prices and low sell prices) down to less competitive limit orders. The bid and ask prices (which again, are simply the prices in the limit orders at the top of the buy and sell books, respectively) together are sometimes referred to as the *inside market*, and the difference between them as the *spread*. By definition, the order books always consist exclusively of *unexecuted* orders — they are queues of orders hopefully waiting for the price to move in their direction.

How then do orders get executed? There are two methods. First, any time a market order arrives, it is immediately matched with the most competitive limit orders on the opposing book. Thus, a market order to buy 2000 shares is matched with enough volume on the sell order book to fill the 2000 shares. For instance, in the example of Figure 2, such an order would be filled by the two limit sell orders for 500 shares at $24.069, the 500 shares at $24.07, the 200 shares at $24.08, and then 300 of the 1981 shares at $24.09. The remaining 1681 shares of this last limit order would remain as the new top of the sell limit order book. Second, if a buy (sell, respectively) limit order comes in above the ask (below the bid, respectively) price, then the order is matched with orders on the opposing books. It is important to note that *the prices of executions are the prices specified in the limit orders already in the books*, not the prices of the incoming order that is immediately executed.

Every market or limit order arrives atomically and instantaneously — there is a strict temporal sequence in which orders arrive, and two orders can never arrive simultaneously. This gives rise to the definition of the *last price* of the exchange, which is simply the last price at which the exchange executed an order. It is this quantity that is usually meant when people casually refer to the (ticker) price of a stock.

Note that a limit buy (sell, respectively) order with a price of infinity (0, respectively) is effectively a market order. We shall thus assume without loss of generality that all orders are placed as limit order. Although limit orders which are unexecuted may be removed by the party which placed them, for simplicity, we assume that limit orders are never removed from the books.

We refer the reader to [4] for further discussion of modern electronic exchanges and market microstructure.

3.2 The Model

The online *order book trading model* is intended to capture the realistic details of market microstructure just discussed in a competitive ratio setting. In this refined model, a day's market activity is described by a sequence of *limit orders* (p_t, v_t, b_t). Here b_t is a bit indicating whether the order is a buy or sell order, while p_t is the limit order price and v_t the number of shares desired. Following the arrival of each such limit order, an online trading algorithm is permitted to place its own limit order. These two interleaved sources (market and algorithm) of limit orders are then simply operated on according to the matching process described in Section 3.1. Any limit order that is not immediately executable according to this process is placed in the appropriate (buy or sell) book for possible future execution; arriving orders that can be partially or fully executed are so executed, with any residual shares remaining on the respective book.

The goal of a VWAP or OWT selling algorithm is essentially the same as in the price-volume model, but the context has changed in the following two fundamental ways. First, the assumption of infinite liquidity in the price-volume model is eliminated entirely. The number of shares available at any given price is restricted to the total volume of limit orders offering that price. Second, all incoming orders, and therefore the complete limit order books, are assumed to be visible to the algorithm. This is consistent with modern electronic financial exchanges, and indeed is the source of much current interest on Wall Street [4].

In general, the definition of competitive ratios in the order book model is complicated by the fact that now our algorithm's activity influences the sequence of executed prices and volumes. We thus first define the *execution sequence* determined by a limit order sequence (placed by the market and our algorithm). Let $S = (p_1, v_1, b_1), \ldots, (p_T, v_T, b_T)$ be a limit order sequence placed by the market, and let $S' = (p'_1, v'_1, b'_1), \ldots, (p'_T, v'_T, b'_T)$ be a limit order sequence placed by our algorithm (unless otherwise specified, all b'_t are of the sell type). Let $merge(S, S')$ be the *merged sequence* $(p_1, v_1, b_1), (p'_1, v'_1, b'_1), \ldots, (p_T, v_T, b_T), (p'_T, v'_T, b'_T)$, which is the time sequence of orders placed by the market and algorithm. Note that the algorithm has the option of not placing an order, which we can view as a zero volume order.

If we conducted the order book maintenance and order execution process described in Section 3.1 on the sequence $merge(S, S')$, at irregular intervals a trade occurs for some number of shares and some price. In each executed trade, the selling party is either the market or the algorithm. Let $exec_M(S, S') = (q_1, w_1), \ldots, (q_{T'}, w_{T'})$ be the sequence of executions where the market (that is, a party other than the algorithm) was the selling party, where the q_t are the execution prices and w_t the execution volumes. Similarly, we define $exec_A(S, S') = (r_1, x_1), \ldots, (r_{T''}, x_{T''})$ to be the sequence of executions in which the algorithm was the selling party. Thus, $exec_A(S, S') \cup exec_M(S, S')$ is the set of all executions. We generally expect T'' to be (possibly much) smaller than T'.

The revenue of the algorithm and the market are defined as:

$$\mathrm{REV}_M(S,S') \equiv \sum_{t=1}^{T'} q_t w_t\,,\ \ \mathrm{REV}_A(S,S') \equiv \sum_{t=1}^{T''} r_t x_t$$

Note that both these quantities are solely determined by the execution sequences $exec_M(S,S')$ and $exec_A(S,S')$, respectively.

For an algorithm A which is constrained to sell exactly N shares, we define the OWT competitive ratio of A, $R_{\mathrm{OWT}}(A)$, as the maximum ratio (under any $S \in \Sigma$) of the revenue obtained by A, as compared to the revenue obtained by an optimal offline algorithm A^*. More formally, for A^* which is constrained to sell exactly N shares, we define

$$R_{\mathrm{OWT}}(A) = \max_{S\in\Sigma}\max_{A^*} \frac{\mathrm{REV}_{A^*}(S\,S^*)}{\mathrm{REV}_A(S,S')}$$

where S^* is the limit order sequence placed by A^* on S. If the algorithm A is randomized then we take the appropriate expectation with respect to $S' \sim A$.

We define the VWAP competitive ratio, $R_{\mathrm{VWAP}}(A)$, as the maximum ratio (under any $S \in \Sigma$) between the market and algorithm VWAPs. More formally, define $\mathrm{VWAP}_M(S,S')$ as $\mathrm{REV}_M(S,S')/\sum_{t=1}^{T'} w_t$, where the denominator is just the total executed volume of orders placed by the market. Similarly, we define $\mathrm{VWAP}_A(S,S')$ as $\mathrm{REV}_A(S,S')/N$, since we assume the algorithm sells *no more* than N shares (this definition implicitly assumes that A gets a 0 price for unsold shares). The VWAP competitive ratio of A is then:

$$R_{\mathrm{VWAP}}(A) = \max_{S\in\Sigma}\{\mathrm{VWAP}_M(S,S')/\mathrm{VWAP}_A(S,S')\}$$

where S' is the online sequence of limit orders generated by A in response to the sequence S.

3.3 OWT Results in the Order Book Model

For the OWT problem in the order book model, we introduce a more subtle version of the price variability assumption. This is due to the fact that our algorithm's trading can impact the high and low prices of the day. For the assumption below, note that $exec_M(S,\emptyset)$ is the sequence of executions *without* the interaction of our algorithm.

3.3.0.3 Order Book Price Variability Assumption..

Let $0 < p_{\min} \le p_{\max}$ be known positive constants, and define $R = p_{\max}/p_{\min}$. For all intraday trading sequences $S \in \Sigma$, the prices p_t in the sequence $exec_M(S,\emptyset)$ satisfy $p_t \in [p_{\min}, p_{\max}]$, for all $t = 1,\ldots,T$.

Note that this assumption does *not* imply that the ratios of high to low prices under the sequences $exec_M(S,S')$ or $exec_A(S,S')$ are bounded by R. In fact, the ratio in the sequence $exec_A(S,S')$ could be infinite if the algorithm ends up selling some stocks at a 0 price.

THEOREM 6. *In the order book model under the order book price variability assumption, there exists an online algorithm A for selling N shares achieving sell OWT competitive ratio $R_{\mathrm{OWT}}(A) = 2\log(R)\log(N)$.*

PROOF. The algorithm A works by guessing a price p in the set $\{p_{min}2^i : 1 \le i \le \log(R)\}$ and placing a sell limit order for all N shares at the price p at the beginning of the day. (Alternatively, algorithm A can place $\log(R)$ sell limit orders, where the i-th one has price $2^i p_{min}$ and volume $N/\log(R)$.) By placing an order at the beginning of the day, the algorithm undercuts all sell orders that will be placed during the day for a price of p or higher, meaning the algorithm's N shares must be filled first at this price. Hence, if there were k shares that would have been sold at price p or higher without our activity, then A would sell at least kp shares.

We define $\{p_j\}$ to be the multiset of prices of individual shares that are either executed or are buy limit order shares that remained unexecuted, *excluding the activity of our algorithm* (that is, assuming our algorithm places no orders). Assume without loss of generality that $p_1 \ge p_2 \ge \ldots$. Consider guessing the kth highest such price, p_k. If an order for N shares is placed at the day's start at price p_k, then we are guaranteed to obtain a return of kp_k. Let $k^* = \mathrm{argmax}_k\{kp_k\}$. We can view our algorithm as attempting to guess p_{k^*}, and succeeding if the guess p satisfies $p \in [p_{k^*}/2, p_{k^*}]$. Hence, we are $2\log(R)$ competitive with the quantity $\max_{1\le k\le N} kp_k$. Note that

$$\begin{aligned}\rho &\equiv \sum_{i=1}^{N} p_i\\ &= \sum_{i=1}^{N}\frac{1}{i}ip_i\\ &\le \max_{1\le k\le N} kp_k \sum_{i=1}^{N}\frac{1}{i}\\ &\le \log(N)\max_{1\le k\le N} kp_k\end{aligned}$$

where ρ is defined as the sum of the top N prices p_i without A's involvement.

Similarly, let $\{p'_j\}$ be the multiset of prices of individual executed shares, or the prices of unexecuted buy order shares, but now *including* the orders placed by some selling algorithm A'. We now wish to show that for all algorithms A' which sell N shares, $\mathrm{REV}_{A'} \le \sum_{i=1}^{N} p'_i \le \rho$. Essentially, this inequality states the intuitive idea that a selling algorithm can only lower executed or unmatched buy order share prices. To prove this, we use induction to show that the removal of the activity of a selling algorithm causes these prices to increase. First, remove the last share in the last sell order placed by either A' or the market on an arbitrary sequence $merge(S,S')$ — by this we mean, take the last sell order placed by A' or the market and decrease its volume by one share. After this modification, the top N prices $p'_1 \ldots p'_N$ will not decrease. This is because either this sell order share was not executed, in which case the claim is trivially true, or, if it was executed, the removal of this sell order share leaves an additional unexecuted buy order share of equal or higher price. For induction, assume that if we remove a share from any sell order that was placed, by A' or the market, at or after time t then the top N prices do not decrease. We now show that if we remove a share from the last sell order that was placed by A' or the market before time t, then the top N prices do not decrease. If this sell order share was not executed, then the claim is trivially true. Else, if the sell order share was executed, then claim is true because by removing this executed share from the sell order either: i) the corresponding buy order share (of equal or higher value) is unmatched on the remainder of the sequence, in which case the claim is true; or ii) this buy order matches some sell order share at an equal or higher price, which has the effect of removing a share from a sell order on the remainder of the sequence, and, by the inductive assumption, this can only increase prices. Hence, we have proven that for all A' which sell N shares $\text{REV}_{A'} \leq \rho$.

We have now established that our revenue satisfies

$$\begin{aligned} 2\log(R)E_{S'\sim A}[\text{REV}_A(S,S')] &\geq \max_{1\leq k\leq N}\{kp_k\} \\ &\geq \rho/\log(N) \\ &\geq \max_{A'}\{\text{REV}_{A'}\}/\log(N), \end{aligned}$$

where A' performs an arbitrary sequence of N sell limit orders. □

3.4 VWAP Results in the Order Book Model

The OWT algorithm from Theorem 6 can be applied to obtain the following VWAP result:

COROLLARY 7. *In the order book model under the order book price variability assumption, there exists an online algorithm A for selling N shares achieving sell VWAP competitive ratio $R_{\text{VWAP}}(A) = O(\log(R)\log(N))$.*

We now make a rather different assumption on the sequences S.

3.4.0.4 Bounded Order Volume and Max Price Assumption.

The set of sequences Σ satisfies the following two properties. First, we assume that each order placed by the market is of volume less than γ, which we view as a mild assumption since typically single orders on the market are not of high volume (due to liquidity issues). This assumption allows our algorithm to place at least one limit order at a time interleaved with approximately γ market executions. Second, we assume that there is "large" volume in the sell order books below the price p_{max}, which means that no orders placed by the market will be executed above the price p_{max}. The simplest way to instantiate this latter assumption in the order book model is to assume that each sequence $S \in \Sigma$ starts by placing a huge number of sell orders (more than V_{max}) at price p_{max}.

Although this assumption has a maximum price parameter, it does not imply that the price ratio R is finite, since it does not imply any lower bound on the prices of buy or executed shares (aside from the trivial one of 0).

THEOREM 8. *Consider the order book model under the bounded order volume and max price assumption. There exists an algorithm A in which after exactly γN market executions have occurred, then A has sold at most N shares and*

$$\begin{aligned} \frac{\text{REV}_A(S,S')}{N} &= \text{VWAP}_A(S,S') \\ &\geq (1-\epsilon)\text{VWAP}_M(S,S') - \frac{p_{max}}{\epsilon N} \end{aligned}$$

where S' is a sequence of N sell limit orders generated by A when observing S.

PROOF. The algorithm divides the trading day into *volume intervals* whose real-time duration may vary. For each period i in which γ shares have been executed in the market, the algorithm computes the market VWAP of only those shares traded in period i; let us denote this by VWAP_i. Following this ith volume interval, the algorithm places a limit order to sell exactly one share at a price "close" to VWAP_i.

More precisely, the algorithm only places orders at the discrete prices $(1-\epsilon)p_{max}, (1-\epsilon)^2p_{max}, \ldots$. Following volume interval i, the algorithm places a limit order to sell one share at the discretized price that is closest to VWAP_i, but which is *strictly smaller*.

For the analysis, we begin by noting that if *all* of the algorithm's limit orders are executed during the day, the total *revenue* received by the algorithm would be at least $(1-\epsilon)\text{VWAP}_M(S,S')N$. To see this, it suffices to note that $\text{VWAP}_M(S,S')$ is a uniform mixture of the VWAP_i (since by definition they each cover the same amount of market volume); and if all the algorithm's limit orders were executed, they each received more than $(1-\epsilon)\text{VWAP}_i$ dollars for the interval i they followed.

We now count the potential "lost revenue" of the algorithm due to unexecuted limit orders. By the assumption that individual orders are placed with volume less than γ, then our algorithm is able to place a limit order during every block of γ shares have been traded. Hence, after γN market orders have been executed, A has placed N orders in the market.

Note that there can be at most one limit order (and thus, at most one share) left unexecuted at each level of the discretized price ladder defined above. This is because following interval i, the algorithm places its limit order strictly below VWAP_i, so if $\text{VWAP}_j \geq \text{VWAP}_i$ for $j > i$, this limit order must have been executed. Thus unexecuted limit orders bound the VWAPs of the remainder of the day, resulting in at most one unexecuted order per price level.

A bound on the lost revenue is thus the sum of the discretized prices: $\sum_{i=1}^{\infty}(1-\epsilon)^i p_{max} \leq p_{max}/\epsilon$. Clearly our algorithm has sold at most N shares. □

Note that as N becomes large, VWAP_A approaches $1-\epsilon$ times the market VWAP. If we knew that the final total volume of the market executions is V, then we can set $\gamma = V/N$, assuming that $\gamma >> 1$. If we have only an upper and lower bound on V we should be able to "guess" and incur a logarithmic loss. The following assumption tries to capture the market volume variability.

3.4.0.5 Order Book Volume Variability Assumption.

We now assume that the total volume (which includes the shares executed by both our algorithm and the market) is variable within some known region and that the market volume will be greater than our algorithms volume. More formally, for all $S \in \Sigma$, assume that the total volume V of shares traded in $exec_M(S,S')$, for any sequence S' of N sell limit orders, satisfies $2N \leq V_{min} \leq V \leq V_{max}$. Let $Q = V_{max}/V_{min}$.

The following corollary is derived using a constant $\epsilon = 1/2$ and observing that if we set γ such that $V \leq \gamma N \leq 2V$ then our algorithm will place between N and $N/2$ limit orders.

COROLLARY 9. *In the order book model, if the bounded order volume and max price assumption, and the order book volume variability assumption hold, there exists an online algorithm A for selling at most N shares such that*

$$\text{VWAP}_A(S,S') \geq \frac{1}{4\log(Q)}\text{VWAP}_M(S,S') - \frac{2p_{max}}{N}$$

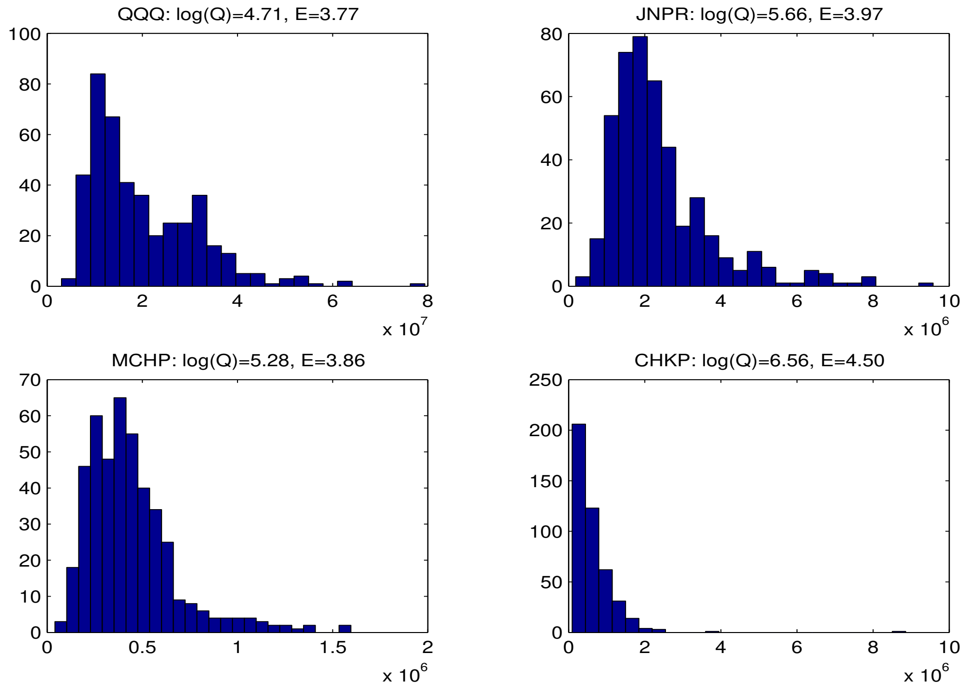

Figure 3: Here we present bounds from Section 4 based on the empirical volume distributions for four real stocks: QQQ, MCHP, JNPR, and CHKP. The plots show histograms for the total daily volumes transacted on Island for these stocks, in the last year and a half, along with the corresponding values of $\log(Q)$ and $\mathcal{E}(P_{vol}^{\text{bins}})$ (denoted by 'E'). We assume that the minimum and maximum daily volumes in the data correspond to V_{min} and V_{max}, respectively. The worst-case competitive ratio bounds (which are twice $\log(Q)$) of our algorithm for those stocks are $9.42, 10.56, 11.32$, and 13.20, respectively. The corresponding bounds on the competitive ratio performance of our algorithm under the volume distribution model (which are twice $\mathcal{E}(P_{vol}^{\text{bins}})$) are better: $7.54, 7.72, 7.94$, and 9.00, respectively (a $20-40\%$ relative improvement). Using a finer volume binning along with a slightly more refined bound on the competitive ratio, we can construct algorithms that, using the empirical volume distribution given as correct, guarantee even better competitive ratios of $2.76, 2.73, 2.75$, and 3.17, respectively for those stocks (details omitted).

4. MACROSCOPIC DISTRIBUTION MODELS

We conclude our results with a return to the price-volume model, where we shall introduce some refined methods of analysis for online trading algorithms. We leave the generalization of these methods to the order book model for future work.

The competitive ratios defined so far measure performance relative to some baseline criterion in the worst case over all market sequences $S \in \Sigma$. It has been observed in many online settings that such worst-case metrics can yield pessimistic results, and various relaxations have been considered, such as permitting a probability distribution over the input sequence.

We now consider distributional models that are considerably weaker than assuming a distribution over complete market sequences $S \in \Sigma$. In the *volume distribution model*, we assume only that there exists a distribution P_{vol} over the *total volume* V traded in the market for the day, and then examine the *worst-case* competitive ratio over sequences consistent with the randomly chosen volume. More precisely, we define

$$R_{\text{VWAP}}(A, P_{vol}) = \mathbf{E}_{V\sim P_{vol}}\left[\max_{S\in seq(V)} \frac{\text{VWAP}_M(S)}{\text{VWAP}_A(S)}\right].$$

Here $V \sim P_{vol}$ denotes that V is chosen with respect to distribution P_{vol}, and $seq(V) \subset \Sigma$ is the set of all market sequences $(p_1, v_1), \ldots, (p_T, v_T)$ satisfying $\sum_{t=1}^{T} v_t = V$.

Similarly, for OWT, we can define

$$R_{\text{OWT}}(A, P_{maxprice}) = \mathbf{E}_{p\sim P_{maxprice}}\left[\max_{S\in seq(p)} \frac{p}{\text{VWAP}_A(S)}\right].$$

Here $P_{maxprice}$ is a distribution over just the maximum price of the day, and we then examine worst-case sequences consistent with this price ($seq(p) \subset \Sigma$ is the set of all market sequences satisfying $\max_{1\le t\le T} p_t = p$). Analogous buy-side definitions can be given.

We emphasize that in these models, only the *distribution* of maximum volume and price is known to the algorithm. We also note that our probabilistic assumptions on S are considerably weaker than typical statistical finance models, which would posit a detailed stochastic model for the step-by-step evolution of (p_t, v_t). Here we instead permit only a distribution over crude, macroscopic measures of the entire day's market activity, such as the total volume and high price, and analyze the worst-case performance consistent with these crude measures. For this reason, we refer to such settings as the *macroscopic distribution model*.

The work of El-Yaniv et al. [3] examines distributional assumptions similar to ours, but they emphasize the *worst-*

case choices for the *distributions* as well, and show that this leads to results no better than the original worst-case analysis over all sequences. In contrast, we feel that the analysis of *specific* distributions P_{vol} and $P_{maxprice}$ is natural in many financial contexts and our preliminary experimental results show significant improvements when this rather crude distributional information is taken into account (see Figure 3). Our results in the VWAP setting examine the cases where these distributions are known exactly or only approximately. Similar results can be obtained for macroscopic distributions of maximum daily price for the one-way trading setting.

4.1 Results in the Macroscopic Distribution Model

We begin by noting that the algorithms examined so far work by binning total volumes or maximum prices into bins of exponentially increasing size, and then "guessing" the *index* of the bin in which the actual quantity falls. It is thus natural that the macroscopic distribution model performance of such algorithms (which are common in competitive analysis) might depend on the distribution of the true bin index.

In the remaining, we assume that Q is a power of 2 and the base of the logarithm is 2. Let P_{vol} denote the distribution of total daily market volume. We define the related distribution P_{vol}^{bins} over bin indices i as follows: for all $i = 1, \dots, \log(Q) - 1$, $P_{vol}^{\text{bins}}(i)$ is equal to the probability, under P_{vol}, that the daily volume falls in the interval $[V_{min}2^{i-1}, V_{min}2^{i})$, and $P_{vol}^{\text{bins}}(\log(Q))$ is for the last interval $[V_{max}/2, V_{max}]$.

We define $\mathcal{E}$ as

$$\begin{aligned}\mathcal{E}(P_{vol}^{\text{bins}}) &\equiv \left(E_{i\sim P_{vol}^{\text{bins}}}\left[\sqrt{1/P_{vol}^{\text{bins}}(i)}\right]\right)^2 \\ &= \left(\sum_{i=1}^{\log(Q)} \sqrt{P_{vol}^{\text{bins}}(i)}\right)^2 .\end{aligned}$$

Since the support of P_{vol}^{bins} has only $\log(Q)$ elements, $\mathcal{E}(P_{vol}^{\text{bins}})$ can vary from 1 (for distributions P_{vol} that place all of their weight in only one of the $\log(Q)$ intervals between $V_{min}, V_{min}2, V_{min}4, \dots, V_{max}$) to $\log(Q)$ (for distributions P_{vol} in which the total daily volume is equally likely to fall in any one of these intervals). Note that distributions P_{vol} of this latter type are far from uniform over the entire range $[V_{min}, V_{max}]$.

THEOREM 10. *In the volume distribution model under the volume variability assumption, there exists an online algorithm A for selling N shares that, using only knowledge of the total volume distribution P_{vol}, achieves $R_{\text{VWAP}}(A, P_{vol}) \le 2\mathcal{E}(P_{vol}^{\text{bins}})$.*

All proofs in this section are provided in the appendix.

As a concrete example, consider the case in which P_{vol} is the uniform distribution over $[V_{min}, V_{max}]$. In that case, P_{vol}^{bins} is exponentially increasing and peaks at the last bin, which, having the largest width, also has the largest weight. In this case $\mathcal{E}(P_{vol}^{\text{bins}})$ is a constant (*i.e.*, independent of Q), leading to a constant competitive ratio. On the other hand, if P_{vol} is exponential, then P_{vol}^{bins} is uniform, leading to an $O(\log(Q))$ competitive ratio, just as in the more adversarial price-volume setting discussed earlier. In Figure 3, we provide additional specific bounds obtained for empirical total daily volume distributions computed for some real stocks.

We now examine the setting in which P_{vol} is unknown, but an approximation $\tilde{P}_{vol}$ is available. Let us define

$$\mathcal{C}(P_{vol}^{\text{bins}}, \tilde{P}_{vol}^{\text{bins}}) = \left[\sum_{j=1}^{\log(Q)} \sqrt{\tilde{P}_{vol}^{\text{bins}}(j)}\right] \left[\sum_{i=1}^{\log(Q)} \frac{P_{vol}^{\text{bins}}(i)}{\sqrt{\tilde{P}_{vol}^{\text{bins}}(i)}}\right].$$

$\mathcal{C}$ is minimized at $\mathcal{C}(P_{vol}^{\text{bins}}, P_{vol}^{\text{bins}}) = \mathcal{E}(P_{vol}^{\text{bins}})$, and $\mathcal{C}$ may be infinite if $\tilde{P}_{vol}^{\text{bins}}(i)$ is 0 when $P_{vol}^{\text{bins}}(i) > 0$.

THEOREM 11. *In the volume distribution model under the volume variability assumption, there exists an online algorithm A for selling N shares that using only knowledge of an approximation $\tilde{P}_{vol}$ of P_{vol} achieves $R_{\text{VWAP}}(A, P_{vol}) \le 2\mathcal{C}(P_{vol}^{\text{bins}}, \tilde{P}_{vol}^{\text{bins}})$.*

As an example of this result, suppose our approximation obeys $(1/\alpha)P_{vol}^{\text{bins}}(i) \le \tilde{P}_{vol}^{\text{bins}}(i) \le \alpha P_{vol}^{\text{bins}}(i)$ for all i, for some $\alpha > 1$. Thus our estimated bin index probabilities are all within a factor of α of the truth. Then it is easy to show that $\mathcal{C}(P_{vol}^{\text{bins}}, \tilde{P}_{vol}^{\text{bins}}) \le \alpha\mathcal{E}(P_{vol}^{\text{bins}})$, so according to Theorems 10 and 11 our penalty for using the approximate distribution is a factor of α in competitive ratio.

5. REFERENCES

[1] B. Awerbuch, Y. Bartal, A. Fiat, and A. Rosén. Competitive non-preemptive call control. In *Proc. 5'th ACM-SIAM Symp. on Discrete Algorithms*, pages 312–320, 1994.

[2] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[3] R. El-Yaniv, A. Fiat, R. M. Karp, and G. Turpin. Optimal search and one-way trading online algorithms. *Algorithmica*, 30:101–139, 2001.

[4] M. Kearns and L. Ortiz. The Penn-Lehman automated trading project. *IEEE Intelligent Systems*, 2003. To appear.

6. APPENDIX

6.1 Proofs from Subsection 2.3

PROOF. (Sketch of Theorem 3) W.l.o.g., assume that $Q = 1$ and the total volume is V. Consider the time t where the fixed schedule f sells the least, then $f_t \le N/T$. Consider the sequences where at time t we have $p_t = p_{max}$, $v_t = V$, and for times $t' \ne t$ we have $p_{t'} = p_{min}$ and $v_{t'} = 0$. The VWAP is p_{max} and the fixed schedule average is $(N/T)p_{max} + (N - N/T)p_{min}$. □

PROOF. (Sketch of Theorem 4) The algorithm simply sells $u_t = (v_t/V_{min})N$ shares at time t. The total number of shares sold U is clearly more than N and

$$U = \sum_t u_t = \sum_t (v_t/V_{min})N = (V/V_{min})N \le QN$$

The average price is

$$VWAP_A(S) = (\sum_t p_t u_t)/U = \sum_t p_t(v_t/V) = VWAP_M(S),$$

where we used the fact that $u_t/U = v_t/V$. □

PROOF. (of Theorem 5) We start with the proof of the lower bound. Consider the following scenario. For the first T time units we have a price of $\sqrt{R}p_{min}$, and a total volume of V_{min}. We observe how many shares the online algorithm has bought. If it has bought more than half of the shares, the remaining time steps have price p_{min} and volume $V_{max} - V_{min}$. Otherwise, the remaining time steps have price p_{max} and negligible volume.

In the first case the online has paid at least $\sqrt{R}p_{min}/2$ while the VWAP is at most $\sqrt{R}p_{min}/Q + p_{min}$. Therefore, in this case the competitive ratio is $\Omega(Q)$. In the second case the online has to buy at least half of the shares at p_{max}, so its average cost is at least $p_{max}/2$. The market VWAP is $\sqrt{R}p_{min} = p_{max}/\sqrt{R}$, hence the competitive ratio is $\Omega(\sqrt{R})$.

For the upper bound, we can get a $\sqrt{R}$ competitive ratio, by buying all the shares once the price drops below $\sqrt{R}p_{min}$. The Q upper bound is derive by running an algorithm that assumes the volume is V_{min}. The online pays a cost of p, while the VWAP will be at least p/Q. □

6.2 Proofs from Section 4

PROOF. (Sketch of Theorem 10) We use the idea of guessing the total volume from Theorem 1, but now allow for the possibility of an arbitrary (but known) distribution over the total volume. In particular, consider constructing a distribution G_{vol}^{bins} over a set of volume values using P_{vol} and use it to guess the total volume V. Let the algorithm guess $\hat{V} = V_{min}2^i$ with probability $G_{vol}^{\text{bins}}(i)$. Then note that, for any price-volume sequence S, if $V \in [V_{min}2^{i-1}, V_{min}2^i]$, $\text{VWAP}_A(S) \geq G_{vol}^{\text{bins}}(i)\text{VWAP}_M(S)/2$. This implies an upper bound on $R_{\text{VWAP}}(A, P_{vol})$ in terms of G_{vol}^{bins}. We then get that $G_{vol}^{\text{bins}}(i) \propto \sqrt{P_{vol}^{\text{bins}}(i)}$ minimizes the upper bound, which leads to the upper bound stated in the theorem. □

PROOF. (Sketch of Theorem 11) Replace P_{vol} with $\tilde{P}_{vol}$ in the expression for G_{vol}^{bins} in the proof sketch for the last result. □