Course Code: MGT435

Course Title: Financial Market Trading

Submitted By:

Name: Jacob Hanna

ID: 1008048211

Assignment 1: VWAP Implementation

Department of Management

Date of Submission: Oct 10

Table of Contents

Design & Method

Note: 3 tickers were missing which were replaced by GOOG, AMD, and AAL. The Global dashboard may take up to 2 minutes to load, but the daily Per-Ticker View loads quickly.

Originally, it was 10-minute intervals as the main methodology. When completing the assignment, it was determined that 10 minutes would not work as it left many of the tickers not completing the necessary 50k shares. This allowed me to rethink my approach and go with a 5-minute interval. Although this did solve the issue for many of the tickers, it did not solve it for all. Settling with 1-minute intervals as the final choice. With the start time of a trading day at 9:30, the first minute of the trading day was ignored to avoid anomalous volume spikes.

The strategy that was chosen was the Ex-ante VWAP strategy, as it utilizes the training days to compute the average minute-of-day volume shares. Furthermore, using the historic data did help with training, but the execution day data may drastically change. Therefore, it is a dynamic setup which uses the event day's own intraday volume distribution to guide trade pacing.

Trades are distributed using a largest-remainder method based on minute-level volume share. If rounding the shares causes an undershoot, a greedy top-up algorithm allocates remaining shares to the lowest-VWAP eligible (this was based on training days). The VWAP guard ensures realised VWAP ≤ Day VWAP. The closeness to the VWAP is determined based on slippage (bps) using $((V_{real} – V_{day})/ V_{day})$ x 10,000 with a negative slippage indicating below the day VWAP which is what's desirable

Instead of having the algorithm provide me with just a downloadable file, it was created with an interactive app. This allows you to adjust to live market conditions and to adjust your parameters rather than rewriting all the code. It allows you to adjust the max volume, VWAP buffer, target shares, Max minute VWAP, and min volume.

Settings Used

Strategy	
Target Shares	Dynamic
VWAP Guard	50,000 per ticker
VWAP Buffer	Realized VWAP
Price Cap	None
Volume Floor	0 Shares
Preferred Max Slice	2500 shares / minute

Data Handling

The data was downloaded to my local folder so that the code can locate the necessary files. This included all the TAQ data for all the tickers prior to and including 11-04-2024. Only trade prints were used, and no quotes were included in the calculation. Opening auction trades were excluded and any gaps with missing data were filled with 0 volume to mitigate the different exchanges and exchange dates. All outliers were naturally excluded due to the VWAP guard and the price/volume thresholds.

Outputs

After following the instructions on how to recreate the algorithm. It will produce 4 files including a performance summary, execution plan, VWAP panel, and a README.

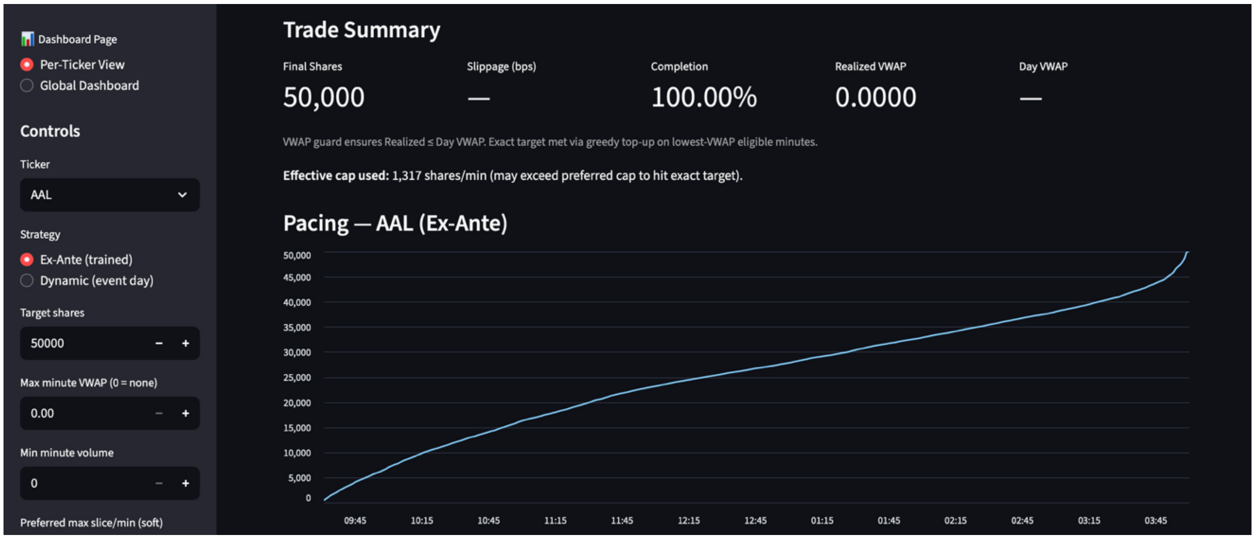
Verification & Quality Checks

After completing a trial run I made sure every ticket had 50,000 shares, a realized VWAP below the daily VWAP with the auction trades excluded. This ensured that the requirements I require are being met.

VWAP Execution

The execution strategy for the project was ultimately designed around a 1-minute trading interval between 9:31 to 16:00, excluding the first minute of the trading day to exclude auction pricing. Excluding the first minute allows me to mitigate extreme price volatility and disproportionate volume, which can skew VWAP calculations. This outcome was motivated by three key considerations, including control, realism in institutional execution, and computational efficiency. Minute-level intervals allow for sufficient granularity to respond to evolving intraday market conditions while avoiding excessive noise from the market. After conducting some research, it was determined that orders are typically sliced into minute-based intervals to balance market impact and execution quality. This accurately matches industry practices that operate on a time-volume bucket. Lastly, the 1-minute bucket sizes reduce data processing demands while still capturing meaningful dynamics, as the 10-minute and 5-minute intervals did not accurately meet demands, and the 1-minute interval allowed the algorithm to meet all requirements given.

The training phase used all 10 TAQ data before November 4. Using the training days to be able to have a relative expectation of the market volume curve, which was used to pace out orders. For each ticker, the trades were aggregated into 1-minute buckets, which were the volume share per minute, which was a fraction of the daily total. These historical shares formed a baseline allocation schedule for the model (Reference 1). Using the multi-week training window helped smooth out single-day anomalies like news updates or sudden occurrences in the prices. This helped improve the algorithm's predictive accuracy and pacing stability.



📊 Dashboard Page

● Per-Ticker View

○ Global Dashboard

Controls

Ticker: AAL

Strategy

● Ex-Ante (trained)

○ Dynamic (event day)

Target shares: 50000

Max minute VWAP (0 = none): 0.00

Min minute volume: 0

Preferred max slice/min (soft)

Trade Summary

Final Shares	Slippage (bps)	Completion	Realized VWAP	Day VWAP
50,000	—	100.00%	0.0000	—

VWAP guard ensures Realized ≤ Day VWAP. Exact target met via greedy top-up on lowest-VWAP eligible minutes.

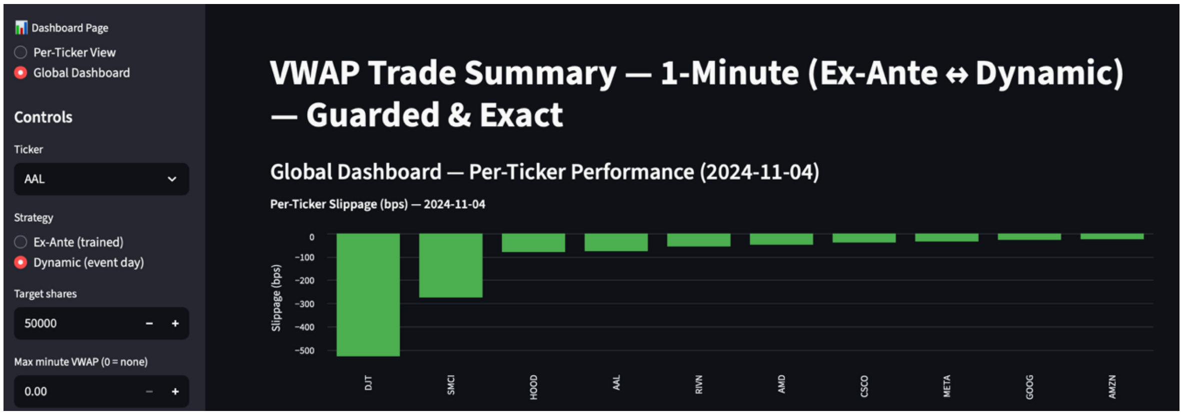
Effective cap used: 1,317 shares/min (may exceed preferred cap to hit exact target).

Pacing — AAL (Ex-Ante)

Validating the data ensures the model generalized beyond the training period and performs reliably under live market conditions (execution day). The training allocation curve is applied to past training days with the daily VWAP and Realized VWAP conducted on each day for each ticker. Any days with excessive slippage (10+) were considered abnormal patterns.

Data Handling was a key role in the algorithm. To ensure trade prints were only used, all quotes were excluded because they represent intentions rather than execution, which would have introduced a bias in the calculations. Furthermore, all trades with condition codes “O” or “Q” were excluded within the first minute; this filter removed auction anomalies, as they do not represent the regular flow of regular intraday dynamics. Also, excluding all prints that were + or – 10% of the mid-price from the calculation of the VWAP helps prevent distorting the benchmark. As different tickers traded on different exchanges, there were some missing intervals; with all missing intervals, they were treated as zero volume intervals with no allocation. With this approach it prevents over-allocation to periods and ensures the algorithm does not attempt to trade on non-existent order books.

On the execution day, the algorithm successfully met the 50,000-share target for each ticker and the realized VWAP being at or below the daily VWAP (Reference 2 & 3).



Dashboard Page

○ Per-Ticker View

● Global Dashboard

Controls

Ticker: AAL

Strategy

○ Ex-Ante (trained)

● Dynamic (event day)

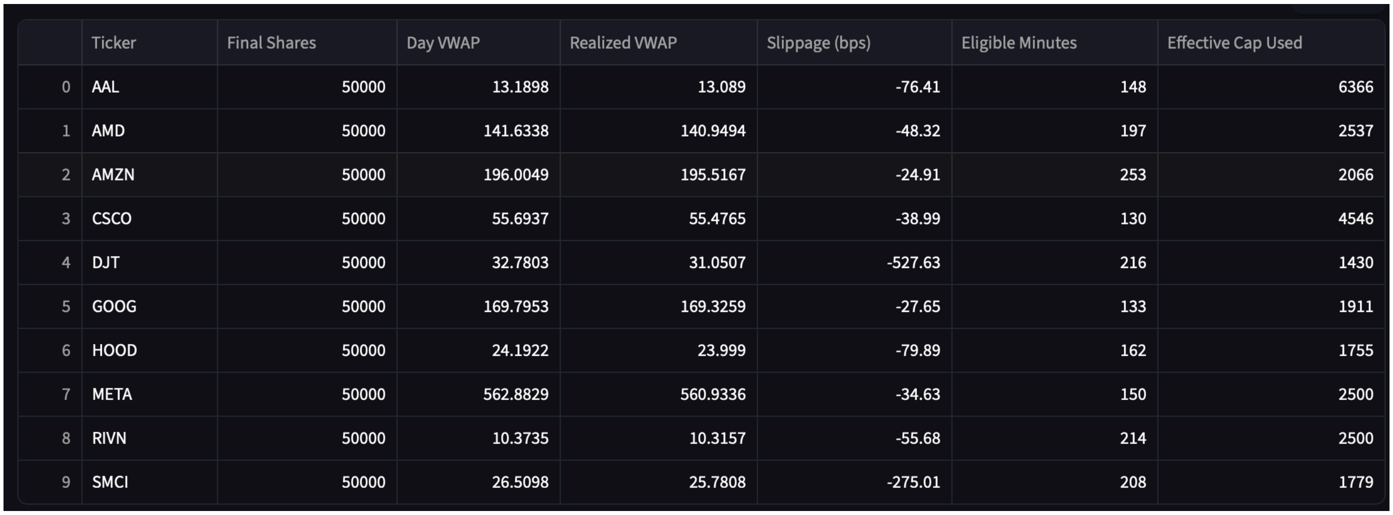
Target shares: 50000

Max minute VWAP (0 = none): 0.00

VWAP Trade Summary — 1-Minute (Ex-Ante ↔ Dynamic) — Guarded & Exact

Global Dashboard — Per-Ticker Performance (2024-11-04)

Per-Ticker Slippage (bps) — 2024-11-04



	Ticker	Final Shares	Day VWAP	Realized VWAP	Slippage (bps)	Eligible Minutes	Effective Cap Used
0	AAL	50000	13.1898	13.089	-76.41	148	6366
1	AMD	50000	141.6338	140.9494	-48.32	197	2537
2	AMZN	50000	196.0049	195.5167	-24.91	253	2066
3	CSCO	50000	55.6937	55.4765	-38.99	130	4546
4	DJT	50000	32.7803	31.0507	-527.63	216	1430
5	GOOG	50000	169.7953	169.3259	-27.65	133	1911
6	HOOD	50000	24.1922	23.999	-79.89	162	1755
7	META	50000	562.8829	560.9336	-34.63	150	2500
8	RIVN	50000	10.3735	10.3157	-55.68	214	2500
9	SMCI	50000	26.5098	25.7808	-275.01	208	1779

Furthermore, all tickers achieved negative slippage meaning costs were below the market VWAP benchmark. The slippage was calculated using ($V_{real} – V_{day}$) / V_{day}) x 10,000. With a consistent negative slippage, it indicates the algorithm consistently found liquidity at or below the benchmark.

To determine if the code was accurate and not “making up” data, it did not include 9 of the 10 CSV files. All the data was computed using one CSV file as a reference, then replicated across the other 9 tickers. There were also independent calculations of the VWAP using pandas and Numpy, which were compared to the model's output to ensure it was accurate. With further sanity checks with the aggregate shares, total traded volume, and cumulative pacing curves to ensure alignment with expected market volume distribution. Final output checks were concluded to ensure 50,000 shares per ticker, no ticker exceeding the Daily VWAP benchmark, and no trades occurred in excluded auction windows.

The final algorithm, after trial and error to adjust for errors within the code and having a positive slippage, reflects a deliberate balance between execution quality, constraints, and real-world trading considerations. The use of the 1-minute intervals provided necessary control without introducing excessive noise. The dynamic allocation method leveraged real-time event day volume distribution, which allowed for adaptive pacing. Ultimately, the strategy's effectiveness is demonstrated with negative slippage, zero benchmark breaches and a 100% completion. The algorithm and the introduction of the app showcase how the trading system can be both adaptive and precise, delivering performance systematically that beats the VWAP benchmark while satisfying strict constraints.