Measuring Execution Quality—Finding the Signal in the Noise

TCA IS STRAIGHTFORWARD IN THEORY BUT HARD IN PRACTICE

The good practice of randomized trading experiments is becoming more widely used, as seen in the growing use of "wheels." But traders still face a big challenge when trying to decide which of several different execution methods is better because of a wide variety of confounding factors and limited data set size available to most traders.

SO WHAT?

The risk is that traders make decisions based on noise, and get worse outcomes for their investors. This research note explores some real-world challenges, and suggests best practices to develop confidence in such comparisons given those challenges.

THE DATA SET

To illustrate the challenges in a "controlled environment," we work with a proprietary data set of 45,000 actual VWAP market orders traded in Q2-Q3 of 2020. We use real orders because many of the challenges in TCA result from the fat-tailed distribution of order characteristics and performance results in real trading data sets, and VWAP is a commonly used algorithm by firms who quantitatively track execution shortfall.

VWAP SF	NUM. OF ORDERS	NOTIONAL VALUE	SPREAD	QTY / ADV	AVG. DURATION
1.26 bps	45,000	$24B	6.6 bps	0.8%	5 hours

TABLE 1

Data summary with value-weighted performance.

A SINGLE SIMULATED TRADING EXPERIMENT

We simulate a typical trader's "experiment." The trader has 400 parent orders per day to work with, split across two algos, A and B, and reviews performance after a 3 month period.

We simulate this experiment by choosing a random 3 month interval from our data set. To mimic a trader splitting the day's basket among algos, for each day in the interval, we randomly assign each order to either group A or group B with a coin toss. Of course, since the same algo traded all the orders and the groups were randomly assigned, groups A and B have the same underlying performance. To simulate the situation where there are actually two different algos used, one better than the other, **we simply improve the average price of each order in group A by 5% of the spread**, or about 0.3 bps on average. This creates two different performance results, one for each algo. Because we're simply improving the average price for the A group, we expect to improve its shortfall regardless of what benchmark we decide to use.

The resulting data set for one such experiment looks like this:

GROUP	VWAP SF	NUM. OF ORDERS	NOTIONAL VALUE	SPREAD	QTY / ADV	AVG. DURATION
A – Better Algo	**1.28 bps (worse SF)**	11,500	$6.4B	6.4 bps	0.86 %	5 hours
B – Worse Algo	**1.21 bps (better SF)**	11,550	$6.9B	6.3 bps	0.88 %	5 hours

TABLE 2

Value-weighted performance summary of a single A/B experiment over 3 months of data.

In this particular experiment, algo B looks slightly better—the opposite of the reality. After 3 months of experiment, splitting flow cleanly across two algos, we still got the wrong answer! But is this just an anomaly?

REPEATED RANDOM SAMPLES

Although in real life a trader only gets to see one outcome of such an experiment, we can simulate the random split of orders between two algos as many times as we want, and we do so 500 times to get a sense for how reliable such an experiment is. What we'd hope is that we consistently see A outperforming B, with perhaps a few anomalous cases where we got the wrong answer. The histogram below shows each such 3-month experiment as a single data point, and the count of these outcomes bucketed by relative outperformance of A over B (negative is good, because lower shortfall). We illustrate the results both in terms of VWAP shortfall and Arrival Price shortfall.

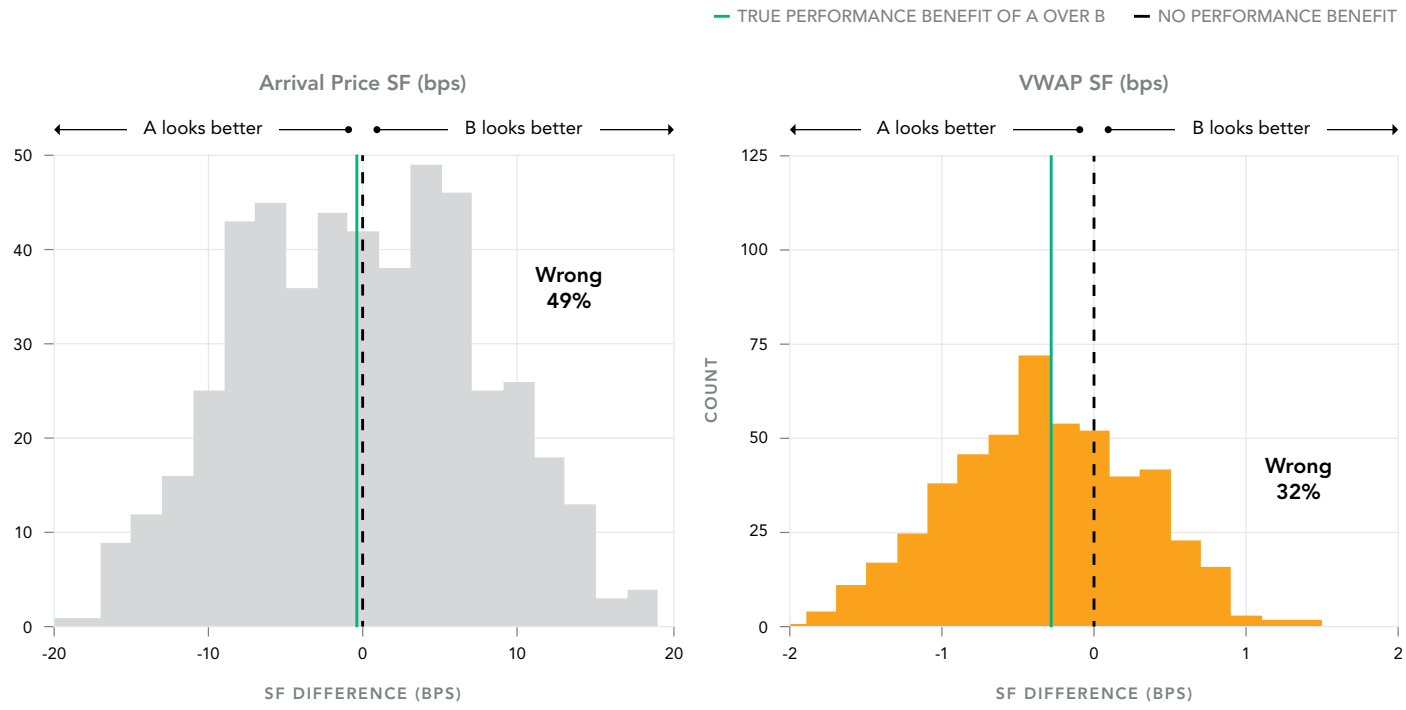
Note the "true" value (A is better than B by about 0.3 bps) is shown by the green line. We see that for VWAP shortfall, the distribution of outcomes is centered around that true value. Yet 1/3 of the time, even this rigorously randomized 3-month long experiment will give the wrong answer, shown by the orange bars to the right of the dotted line, and we'll think that B is actually better than A.

Though for many traders Arrival Price slippage, shown in the left plot, is the true "gold standard" performance metric, it's a much noisier metric.[1] As we see below, measuring by Arrival Price shortfall correctly identifies A as the better algo only 51% of the time, barely more than a random flip of a coin, and erroneously crowns algo B as the winner 49% of the time!

1 For full-day orders, Arrival Price slippage varies by on the order of the stock's daily price change, since there is a single point-in-time benchmark at the start, and trading occurs throughout the day. In contrast, VWAP is effectively a rolling average of prices calculated across the period of the trade, so tends to deviate less from actual average price of an algo that also spreads its trading out across the same period.

FIGURE 1

This figure shows a histogram of the difference between the average shortfall of algos A and B. Each point represents a single experiment as described above, and the histogram shows the distribution of how often each outcome is seen when we repeat the experiment 500 times. Negative values mean that A was observed to be better than B (lower shortfall, the reality), 0 means they're observed to be the same, and positive means B was seen to be better than A.



Trajectory Shortfall

INTRODUCING A LOWER-NOISE SHORTFALL METRIC

Even though VWAP shortfall comparisons require less data than Arrival Price shortfall, they may still require many months to reach reasonably conclusive results—especially if a trader has fewer orders each day or more algos to split them among. To speed up this process, Pragma has developed a lower-noise variant of VWAP shortfall, which we call Trajectory shortfall. The metric is based on splitting VWAP slippage into its two contributors, curve mismatch and microtrading slippage, illustrated in Figure 2.

The goal of Trajectory shortfall is to isolate the microtrading slippage from the idiosyncratic noise that dominates the VWAP benchmark due to day-to-day mismatch between the historical volume pattern and the trading day's actual volume pattern. The details of this calculation are provided in the appendix. Microtrading slippage is by definition everything other than curve mismatch, so it includes order routing, use of short-term signals, order pricing, and so-forth.

If the algo stays relatively close to the average VWAP pattern (as most VWAP algos do), the variability of Trajectory shortfall will typically be a fraction of VWAP shortfall, on the level of 4 bps standard deviation versus the typical VWAP shortfall variability of 18 bps. How does this help in our simulated A/B experiment?

Figure 3 shows the same kind of experiment simulation, but also shows the distribution of Trajectory shortfall outcomes (blue) side by side with VWAP shortfall outcomes (orange). **Trajectory shortfall correctly identifies Algo A as the better algo in 89% of experiments**, whereas the VWAP shortfall gets the right answer only 68% of the time.

Trajectory shortfall dramatically reduces the noise in VWAP slippage, providing significantly more confidence for the trader to choose the right algo for a given amount of data. Alternatively, Trajectory shortfall significantly reduces the amount of data that is needed to reach conclusive results. Note, any reasonable VWAP trajectory can be used to calculate Trajectory shortfall for all provider's algos; each provider's actual VWAP trajectory is not needed.

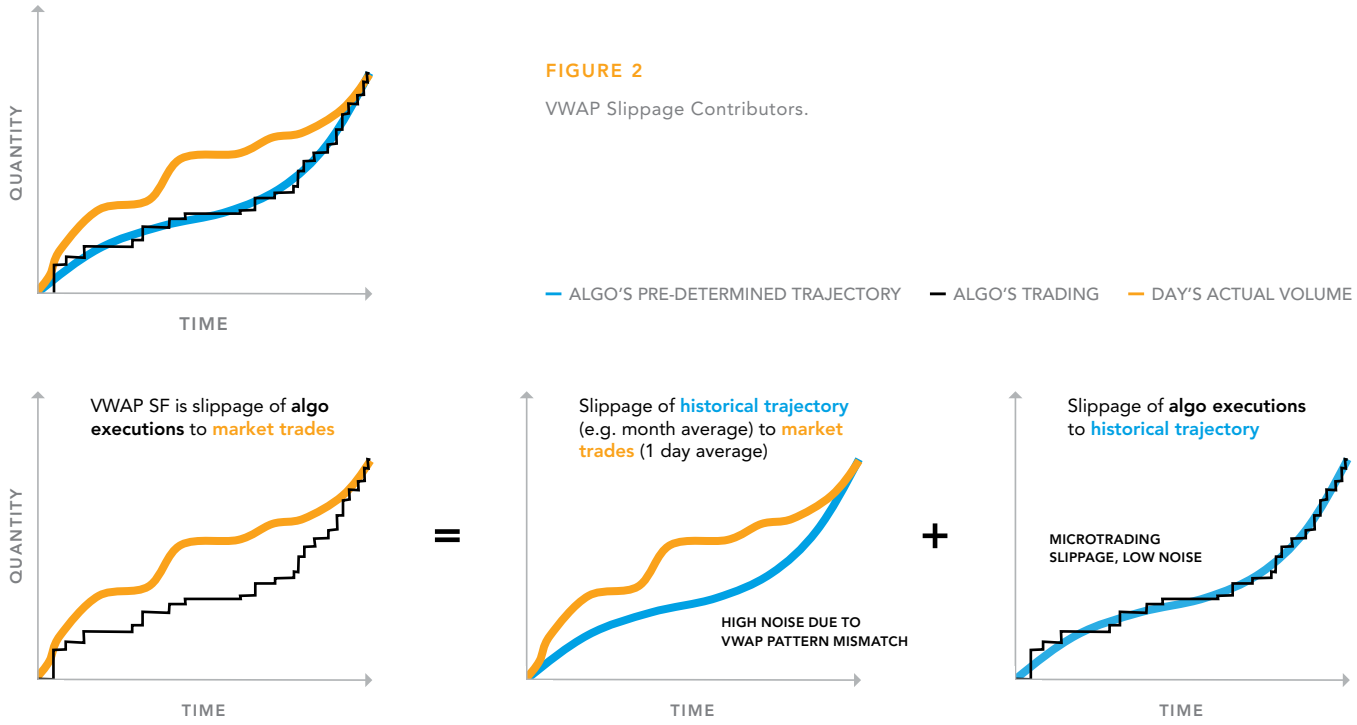
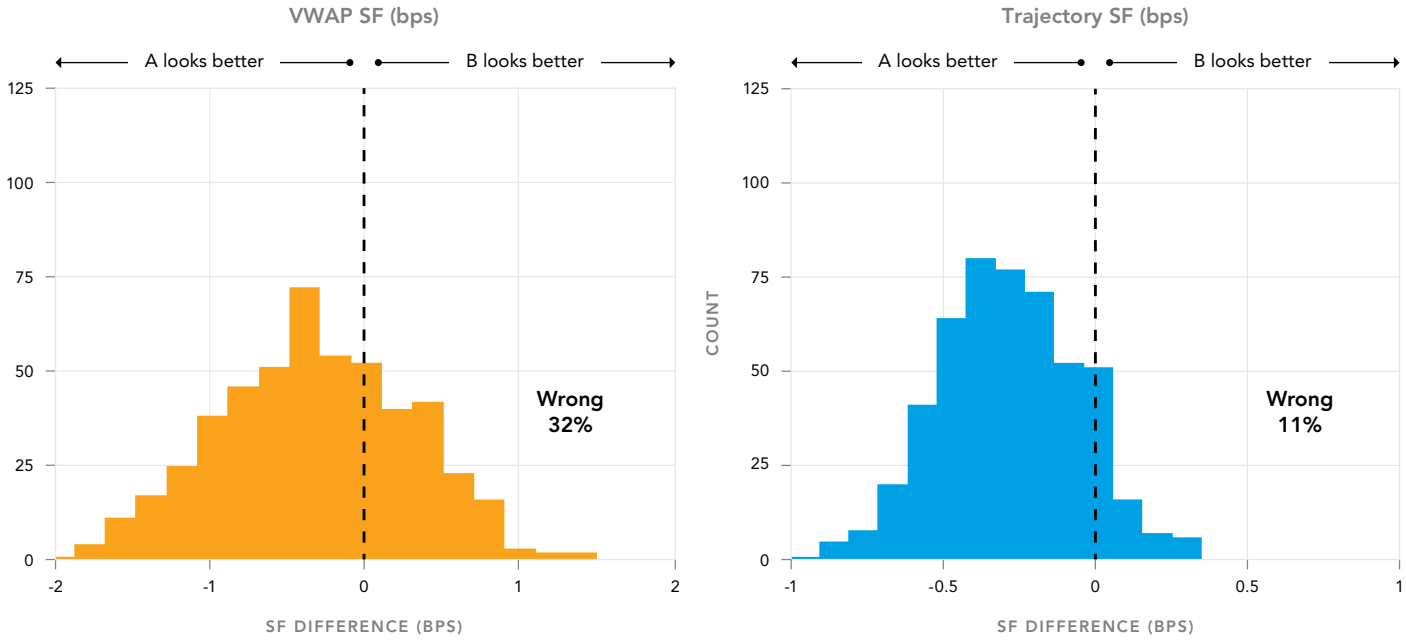


FIGURE 2

VWAP Slippage Contributors.

FIGURE 3 Improvement in better algo detection from using Trajectory shortfall instead of VWAP shortfall.



More Best Practices

RANDOMIZE

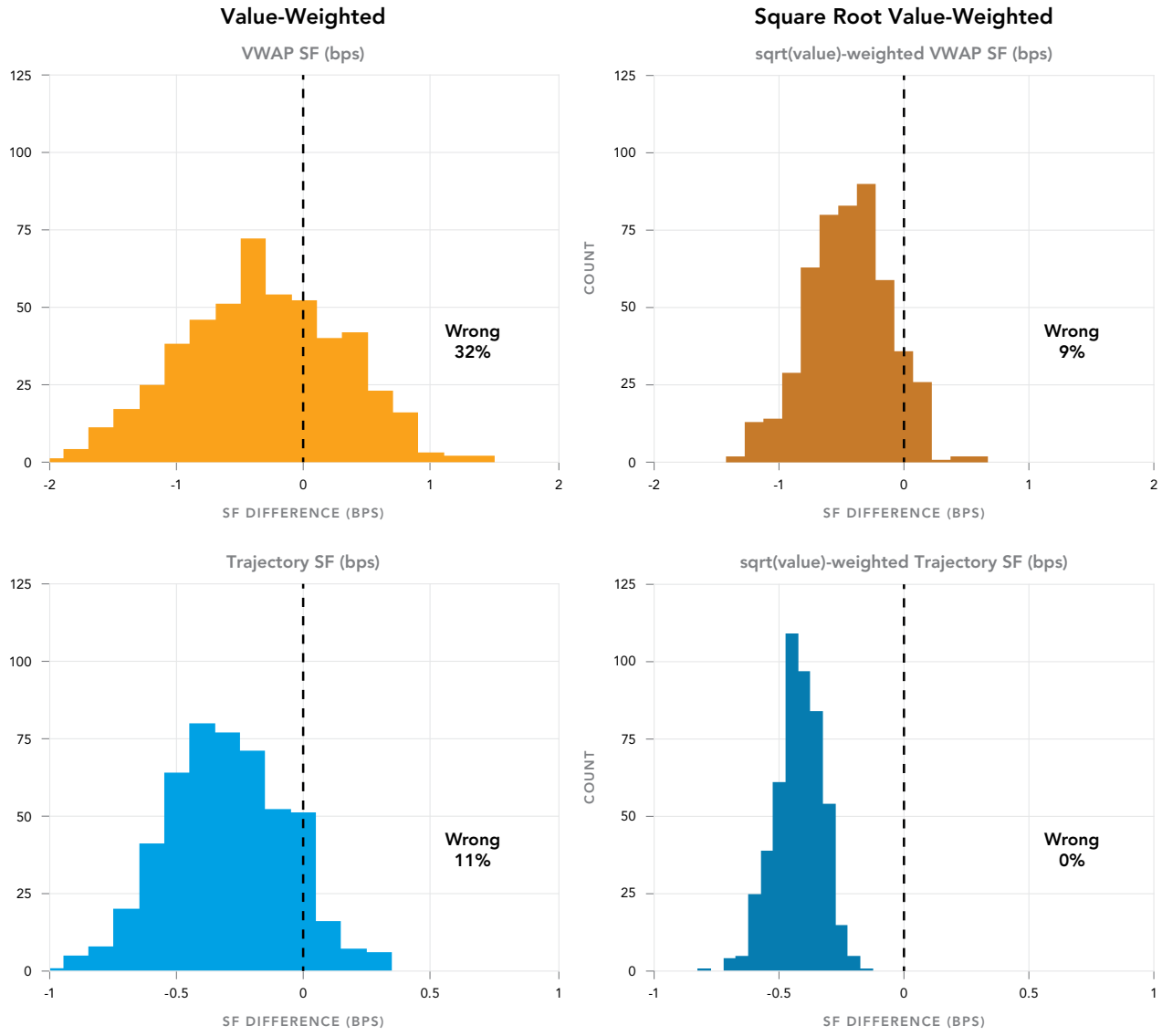
Randomization is a key best practice for doing fair comparisons. Shortcuts like splitting orders by symbol range can introduce systematic differences into the algos' order flows and lead to an invalid experiment. Another appealing approach is to split each parent order into equal parts, routing each part among the algos to be traded simultaneously. But side-by-side parent orders interact with each other in the market—for example one algo might trade more aggressively than another, making the second algo suffer its impact and appear worse, even though the second algo might perform better when trading a parent order alone. Any other built-in bias—for example if individual traders route orders based on their own preferences—can lead to severely biased comparisons. Effectively flipping a coin to see which algo gets the next order, thereby preventing any bias in types of orders an algo gets, is a critical best practice to enable meaningful performance comparisons between brokers or algos.

VALIDATE THAT A/B ORDER FLOWS ARE EQUIVALENT

Even in a well-designed experiment, **it is important to validate that there are no idiosyncratic differences in algos' order flow** by comparing the after-the-fact distributions of symbol characteristics like spread, volatility, ADV, etc., and order characteristics like order durations, % of ADV, spread-normalized market return over the period, to ensure the comparison is fair. If non-negligible differences remain, it may be impossible to determine if algo performance differences are caused by the differences in algos or just the differences in order flow.

SQUARE-ROOT VALUE WEIGHTING

When calculating the average shortfall, it's standard practice to weight the orders by value. However, the fat-tailed distributions common in finance mean that often a small number of orders carry a lot of weight. This contributes to the poor reliability illustrated in our experiment above. Looking at the averages weighted by the **square root of value** can be helpful. This still weighs big orders as more important than small ones,



but in a "gentler" way. In practice, this seems to provide more stable results; the figure above shows that square root value weighted VWAP SF shows the wrong result in only 9% of experiments, rather than the 32% of value-weighted VWAP. Trajectory SF using square root value weighing reduces the chance of identifying the wrong algo as the best to near 0%.

FIGURE 4

Improvement in better algo detection by using square root value weighing.

TRIMMING OUTLIERS

Fat tails also mean that by chance a few orders often have extremely good and extremely bad performance. Such outliers can add a lot of noise to results and contribute to unstable comparisons. Removing just a handful of orders—say the orders that constitute the best and worst 1% or 2% of the data set can dramatically reverse the results. In this

FIGURE 5

particular data set there were not enough large outliers to significantly improve the comparison, so we omit the illustration. However, it's a good practice to include when doing any performance comparison: **a result that rests heavily on a small handful of orders should be treated as unconvincing**.

SLICE AND DICE

Finally, a helpful technique to build confidence—especially in the presence of order flow differences—is to compare performance in buckets of various characteristics, for example, buckets of spread, planned POV %, etc. Consistency in performance across various buckets adds support to the story told in the overall performance numbers, while inconsistency suggests the overall performance may not tell the full story, and care should be taken to use that one number alone to compare the quality of algos.

Conclusion

A few firms have the luxury of thousands of orders per day to spread across their algo providers, and if they randomize properly, can get solid experimental results based on VWAP shortfall within a few months. Detecting even a substantial performance difference like the one used in this paper (5% of the spread) based on arrival price shortfall can take as much as **100 times as much data**. With limited data, most traders are forced to focus on "proxy" benchmarks like VWAP, and to take more care in analyzing performance results.

None of the methods presented in this paper is a silver bullet, but together they can provide improved confidence—or conversely identify situations where apparent differences are likely to be random noise.

Traders who don't make such efforts risk "garbage-in, garbage-out"—acting on information that has the appearance of quantitative rigor, but leads to spurious conclusions that subvert the goals of best execution and cost their investors money.

Appendix

MOTIVATION BEHIND TRAJECTORY SHORTFALL

Figure 6A is a stylized illustration of an ideal VWAP algo that always executes at the midpoint, and thereby has zero VWAP slippage. The first figure row shows an excerpt of the planned VWAP trajectory, which for convenience is a straight line trajectory executing 100 shares every time slice. The second row shows how the market trades during this time as orange circles, showing a pattern that diverges from the average, with some slices having much more volume than others. The third row shows the ideal algo's executions, which are always able to get the midpoint in every slice, as shown by the executions (black squares) being in the middle of the market volume (orange circles). This *ideal* algo *underperforms* VWAP due to the mismatch in how trades print that day compared to the average VWAP pattern. Trajectory shortfall corrects the volume pattern mismatch that VWAP suffers from by re-scaling the volume printed in each slice according to the average historical VWAP pattern. This correction reduces slippage to 0, as expected from an ideal algo.

To build onto the motivation, we next consider a slightly sub-optimal algo in Figure 6B, which incurs

FIGURE 6A

Stylized explanation of the motivation behind Trajectory shortfall using an ideal algo.

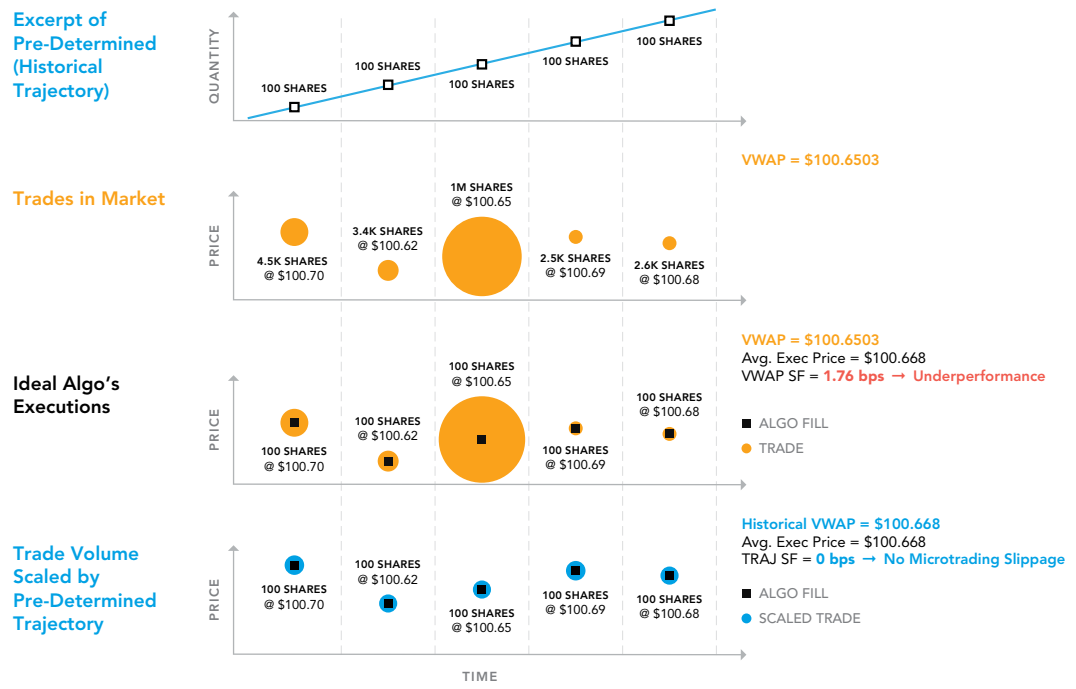
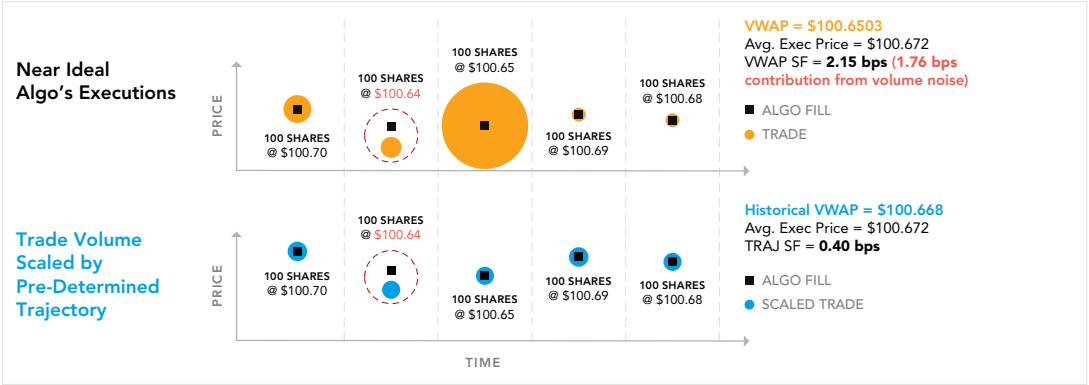


FIGURE 6B

Stylized example of an almost ideal algo that incurs some slippage. The VWAP slippage due to the non-ideal execution is dominated by noise from the volume curve mismatch. Trajectory shortfall removes the noise due to the volume mismatch.



some slippage in one of the five slices, with the rest executing at the midpoint. This results in a VWAP SF of 2.15 bps, a measure of execution quality that is dominated by 1.76 bps slippage due to noise. Trajectory shortfall isolates the slippage from the pattern noise, showing a shortfall of 0.4 bps.

CALCULATING TRAJECTORY SHORTFALL

Figure 7 walks through the steps to calculate Trajectory shortfall using a pre-determined VWAP trajectory. Note, the trajectory can be any available VWAP trajectory, and does not have to match the trajectory used by various algo providers.

- **Row 1:** Partition the planned trajectory into time slices of short duration, and calculate the quantity that is to be executed in each slice.
- **Row 2:** In each time slice, calculate the local VWAP price using all the trades in the time slice.
- **Row 3:** Replace the volume executed in each slice with the quantity that will be executed according to the pre-determined trajectory. Calculate the trajectory-adjusted VWAP as the trajectory-quantity-weighted average of all the slices' VWAPs.

FIGURE 7

Illustrated guide to calculating the Trajectory shortfall.