

Quantifying Long-Term Market Impact

Campbell R. Harvey, Anthony Ledford, Emidio Sciulli,
Philipp Ustinov, and Stefan Zohren

Campbell R. Harvey

is a professor of finance at Duke University in Durham, NC, and a research associate at the National Bureau of Economic Research in Cambridge, MA.
cam.harvey@duke.edu

Anthony Ledford

is chief scientist at Man-AHL in Oxford, UK.
anthony.ledford@man.com

Emidio Sciulli

is head of fast trading strategies at Man-AHL in London, UK.
emidio.sciulli@man.com

Philipp Ustinov

is a principal quant at Man-AHL in London, UK.
philipp.ustinov@man.com

Stefan Zohren

is a principal quant at Man-AHL in London, UK, and deputy director of the Oxford-Man Institute for Quantitative Finance at the University of Oxford in Oxford, UK.
stefan.zohren@man.com

KEY FINDINGS

- Market impact costs are a crucial component of strategy performance—yet these costs are routinely ignored in most academic research studies.
- Although there is considerable research on static trade execution, the analysis of impact costs is very complex if there is autocorrelation in the orders (i.e., a model of dynamic impact).
- We offer a simple method to measure impact when orders are autocorrelated. Our application to a sample of real transactions shows that our method, the expected future flow shortfall, performs well compared to alternative data-hungry approaches and is more responsive to changing market conditions.

ABSTRACT

Impact costs occur when large buy or sell orders move market prices. The measurement of these costs is crucial for the evaluation of potential trading strategies and the successful execution of systematic investment strategies. However, common approaches suffer from a type of myopia: impact is only measured for the current transaction. In many cases, orders are correlated, and the impact of the first order will affect the execution of future orders. The authors propose a new measure that quantifies the long-term effects of market impact: expected future flow shortfall (EFFS). Their method is both intuitive and straightforward to implement. Importantly, the EFFS method performs competitively with far more complex and data-hungry approaches. The method should be useful for both the evaluation of execution methods and the sizing of orders.

The easiest way to boost the alpha of a trading strategy is to reduce costs. Although it is well known that execution costs are important, there is relatively little research on execution compared to signal development. Remarkably, in most published papers, execution costs are routinely ignored. With the rise of systematic trading and machine learning–driven investment strategies (Lopez de Prado 2018), investors are moving toward faster, higher turnover strategies. For those strategies, the effects of trading costs are even more important and could render some strategies unprofitable if the costs are not properly considered.

We propose a new method to measure long-term impact costs. The fact that one trade can affect the price of the next trade is not new. Modern trading systems routinely break larger orders (metaorders) into sequences of smaller (child) orders. Trading more slowly through sequences of child orders reduces costs but also increases risk. There is a vast literature on how to optimally execute those child

orders to trade off costs and risk (Bertsimas and Lo 1998; Bertsimas, Lo, and Hummel 1999; Almgren and Chriss 1999, 2001; Engle, Ferstenberg, and Russell 2012).

Breaking down larger metaorders into child orders introduces autocorrelation at the level of the individual child orders. However, for systematic strategies, even the metaorders can exhibit autocorrelation over longer time horizons; this is what we refer to as *long-term* impact cost. Our method, expected future flow shortfall (EFFS), is designed to measure these longer-term costs. Note that dealing with autocorrelated sequences of metaorders is a different problem setting than the more commonly studied trading of child orders of a single metaorder.

Execution costs can be divided into direct costs, which include commissions, fees, and taxes, and indirect costs, which include the bid–ask spread and market impact. There is existing literature on the effects of both direct and indirect execution costs (see Grinold and Kahn 1999, Bouchaud et al. 2018, and references therein). Market impact is simply the disturbance trading inflicts on prices—in other words, the difference between prices observed during and after trading and the prices that would have been observed had trading not occurred. This quantity, which is not directly observable, has been the subject of considerable study and is often characterized by the so-called *square-root law*, in which the market impact is seen to grow like the square root of the order size (Torre and Ferrari 1997; Grinold and Kahn 1999; Almgren et al. 2005; Moro et al. 2009). Aside from the empirical evidence, attempts have also been made to explain the economic foundations of the square-root law (Kyle and Obizhaeva 2018). In dollar terms, the market impact costs of reasonably sized orders are typically an order of magnitude larger than direct execution costs.

A classical tool of investment management used to measure trading effects is the implementation shortfall (IS) metric (Perold 1988). This shortfall reflects the discrepancy in performance between an ideal “paper” portfolio and the real portfolio. The IS is the difference between transacting at realized prices and transacting at the model (or decision) prices, usually taken to be the midpoint prices when investment decisions are made. The shortfall can be decomposed into execution costs and opportunity costs. Execution costs relate to already executed trades, whereas opportunity costs relate to unexecuted transactions (i.e., past trading opportunities that were unrealized). Faster execution generally increases execution costs and lowers opportunity costs.¹

The focus of this article is to provide an analogue of the IS that quantifies how the market impact of orders can influence the decision prices and hence the trading costs of subsequent metaorders. We illustrate this using a simple week-long scenario with a series of metaorders, although our approach is more suited for shorter horizons. We execute one metaorder per day. In this scenario, an investment decision is made using the decision price (e.g., the opening price). After execution of a given metaorder, the number of shares traded in that metaorder is multiplied by the difference between the weighted average execution price and the decision price. This is recorded as the IS for that day. The next trading day, the process is repeated, and the IS is calculated using the new decision price. This approach works consistently if the investment decisions made on subsequent days are independent. However, this approach fails when orders are correlated. That is, the initial trading decision must account for the potential correlations of orders to yield a reasonable estimate of the cost of the trade—and the profitability of the strategy.

¹To give an example, suppose we wanted to buy 10,000 shares of a stock and spread out the execution over the course of the day to avoid market impact. Over the first hour of trading, we executed 1,000 shares; thus, 9,000 remain unexecuted. If, over the course of the same hour, the price moved from \$100 to \$101, we would have incurred an opportunity cost of \$9,000; that is, we missed out on a price move of \$1 on the 9,000 shares that we had not yet executed.

In practice, ignoring metaorder correlation can lead to flawed implementations of portfolios and biased estimates of the capacity of the underlying strategies. Correlation between metaorders can arise in a variety of situations. Typical examples are momentum and reversion strategies, in which order flow has positive and negative correlations with past price returns, respectively. More widely, any systematic strategy that employs any form of smoothing on raw signals in general creates autocorrelations in the order flow. The approach that we propose suggests a practical way of estimating the additional effect on profit and loss (P&L) coming from the metaorder correlations.

TRADING COSTS AND MARKET IMPACT

We now set up our model and define the terminology. Consider portfolio managers making investment decisions and expressing them in the form of desired trades (i.e., shares to buy or sell). The decision is communicated to the execution agent (which can be part of the same firm or an external service). The execution agent trades the desired amount using elementary actions such as submitting market orders/limit orders to the exchange, cancelling limit orders, and so on. If the desired trade is small—for example, if it is equal to 100 shares—the whole size will be submitted to the exchange in a single order. In the more typical situation for large investors, submitting the whole quantity is impractical (e.g., the desired trade size may exceed the total liquidity available at the exchange at that moment in time). In this case, the execution agent typically splits the desired trade into smaller suborders and executes them incrementally. Those suborders are called *child orders*. We will call the desired trade that was communicated to the execution agent a *metaorder*.

When the investment decision is made, the latest available market price (e.g., the latest midpoint price or previous closing auction price) is noted, and we call this the *decision price*. When the instruction arrives at the execution agent, the latest available market price may be noted again and is called the *arrival price*. We assume that the processes of communication are very fast and efficient so that the decision and arrival prices coincide; accordingly, these terms are used interchangeably in what follows.

We begin our analysis with a single buy metaorder. The portfolio manager cares about how the weighted average execution price of all the spawned child orders compares to the decision price for the metaorder. In practice, the weighted execution price can be extremely variable, as we illustrate in Panel A of Exhibit 1, which shows the evolution of 1,000 simulated price moves during the execution of a buy metaorder starting at a decision price of 100.² For buy orders, the price path moves higher in slightly more than 50% of cases and moves lower in the remainder. On average, across many metaorders, the average execution price path is higher than the decision price. This is not immediately clear in Panel A because the mean impact curve looks flat, but this is simply due to the plotting scale.

In Panel B of Exhibit 1, we focus on a narrower range of prices to see how the impact drives prices higher. Comparing the two panels, it is obvious that the standard deviation of the price moves in Panel A, approximately 100 bps, is much larger than the impact, approximately 10 bps. This illustration is also suggestive of the difficulty

²We generate the trajectories according to the following formula for the price:

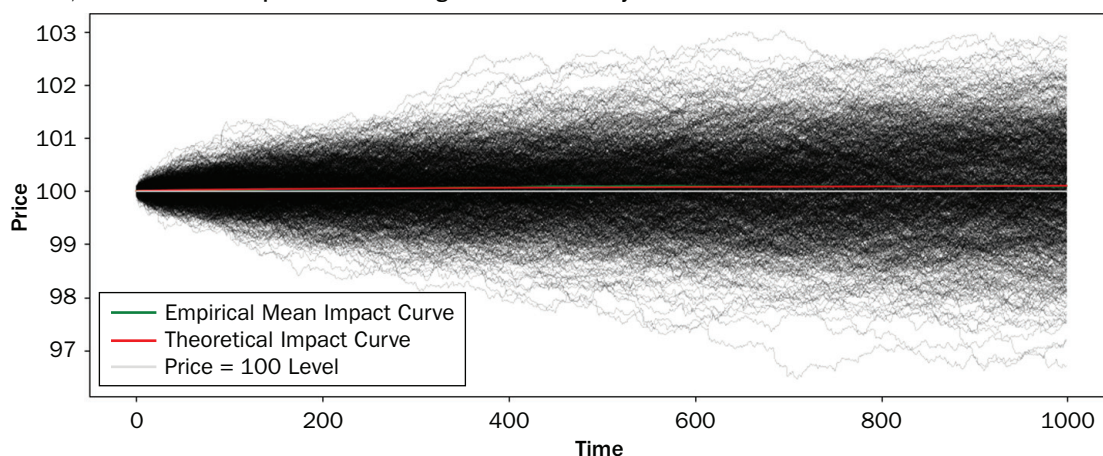
$$P_k = 100 + \sum_{i=0}^{k-1} \sqrt{3/5 \cdot 10^{-3}} X_i + 0.1 \sqrt{i \cdot 10^{-3}},$$

where X_i is the standard Student's t distribution with five degrees of freedom. This expression is chosen to show the theoretical price impact at a realistic scale compared to price moves.

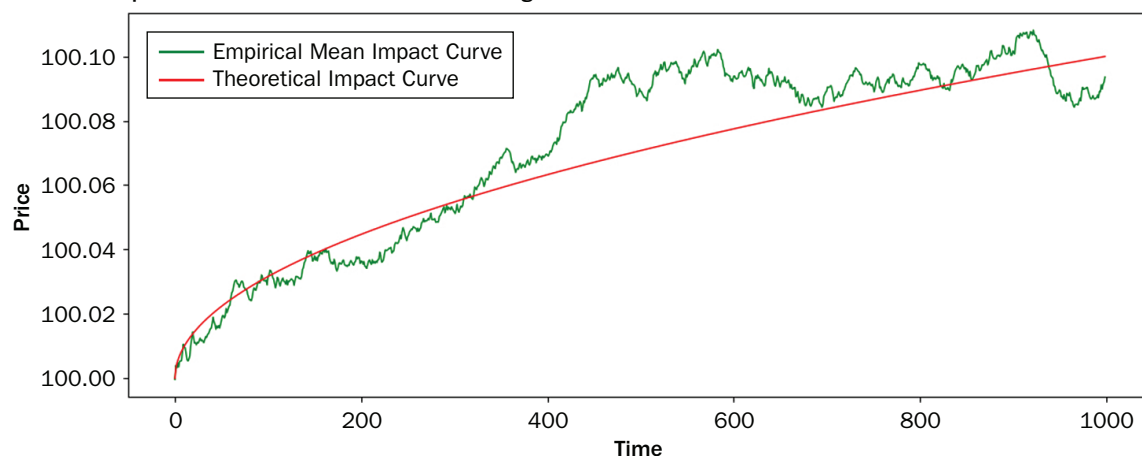
EXHIBIT 1

Market Impact Based on 1,000 Monte Carlo Simulations

Panel A: 1,000 Simulated Midprice Moves During Execution of a Buy Metaorder



Panel B: Midprice Moves under a Narrower Price Range



NOTES: Panel A shows 1,000 simulated midprice moves (black) during the execution of a buy metaorder. The average of those midprice paths gives the empirical mean price impact curve, shown in green, together with the theoretical curve from the square root law in red. These empirical and theoretical impact curves are much clearer in Panel B because it focuses on a narrower range of prices. Note that price impact is around one to two orders of magnitude smaller than typical midpoint price moves, as becomes apparent when comparing the scales of Panels A and B.

we face in measuring price impact curves in practice. An upward change in the asset price that is unrelated to the metaorder may be incorrectly attributed to market impact. Given that the variance of the asset price is so high (compared to the impact cost), attribution to impact is challenging. The difference between the weighted execution price and the decision price expressed relative to the decision price is called *unit slippage* or *unit implementation shortfall* for a buy order. For sell orders, we look at the difference between the decision and weighted execution prices. When multiplied by the quantity traded, this is called the *dollar slippage* or IS.

There are several contributing factors that make the expected IS positive. One is spread cost. For a small metaorder, if an impatient investor submits the entire quantity in one market order, the expected unit cost would be half the bid–ask spread. Even for a more patient investor, trying to use limit orders only, the executed price

may typically be some fraction of the bid–ask spread. The second cause is market impact. Consider the case of a buy metaorder. The basic laws of supply and demand suggest that greater demand should drive prices up. It is also possible that prices rise more than simple supply and demand considerations would suggest because other traders—observing prices going up—may infer this to signal future demand (i.e., prices are cheap). Finally, the third cause is alpha. On average, the portfolio manager tries to buy assets when the expectation is for the price to go up.

We primarily focus on the second component—market impact—but consider impact across different metaorders rather than within a metaorder. This is especially relevant for faster trading strategies, in which one cannot expect that the price impact resulting from a single metaorder has decayed away by the end of the metaorder. Instead, the impact is likely to persist to some degree after the end of that metaorder and thereby affect prices for subsequent metaorders.

There is both theoretical and empirical work that examines how market impact evolves during and after metaorder execution. However, there are several hurdles: Given the variance of asset prices, a large volume of data on trading decisions and execution results is needed to calibrate a model well. To make it even more challenging, this information is typically private to an individual investor. If one has only access to child orders of multiple investors without knowing their identity, reconstructing the corresponding metaorders with high fidelity may prove impossible. Without identity, cause and effect can easily be confused.

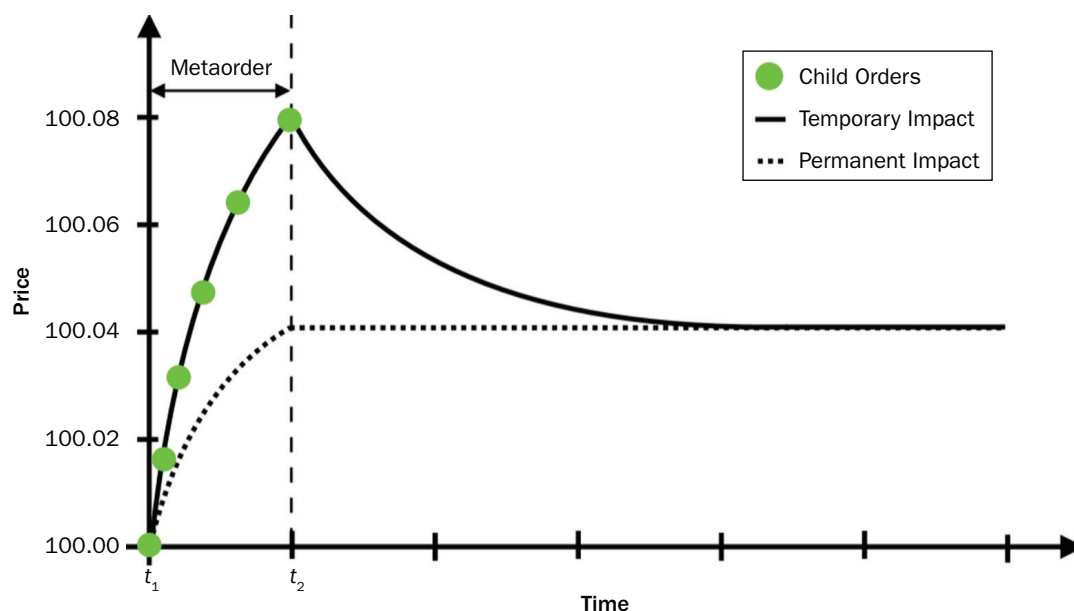
The broad stylized facts of market impact are as follows (see Bouchaud et al. 2018 for a review): The midprice, on average, increases throughout the buy metaorder duration and will likely start reverting after the metaorder finishes trading. It may or may not revert completely. The part of the impact that never reverts (or is extremely slow to revert) is called *permanent impact*. The stylized picture for the price impact of an isolated buy metaorder is presented in Exhibit 2. The price trajectory over the metaorder can be linear or nonlinear (typically concave).

Direct measurement of the permanent market impact is practically impossible because subsequent metaorders may start trading before the reversion of the initial metaorder has finished. For large-scale asset managers, it is often impractical to submit the whole desired amount in one metaorder. Indeed, the amount may not even be possible to trade over a full day. This leads to an autocorrelation effect: buy metaorders tend to be followed, on average, by buy metaorders, and vice versa for sell metaorders. Exhibit 3 presents a stylized picture for trading two consecutive buy metaorders.

Note in Exhibit 3 that the decision price of the second metaorder is approximately 100.025 and thus higher than 100 owing to the impact of the first metaorder (both temporary and permanent). There are several insights here. First, the difference in price at time t_4 (100.025 to 100) is not reflected in the usual slippage metric for either metaorder. Second, if the profitability of the trade idea depends on the execution of both metaorders, it is essential to consider the impact of the first metaorder on the price used to benchmark the second metaorder. Such considerations may also influence the sizing decision of each metaorder for best overall execution. Research or paper trading often assumes that both metaorders could be executed at a price of 100 plus IS. This would misrepresent the viability of the trade idea. This effect on the price, cost, and profitability of the subsequent trades is sometimes called *hidden slippage* because it is not reflected in the traditional slippage metric. The EDFS model enables measurement of this hidden slippage. One might also consider joining the two metaorders into one composite metaorder. Later we discuss the shortcomings of this naïve approach and how the EDFS model overcomes the shortcomings.

EXHIBIT 2

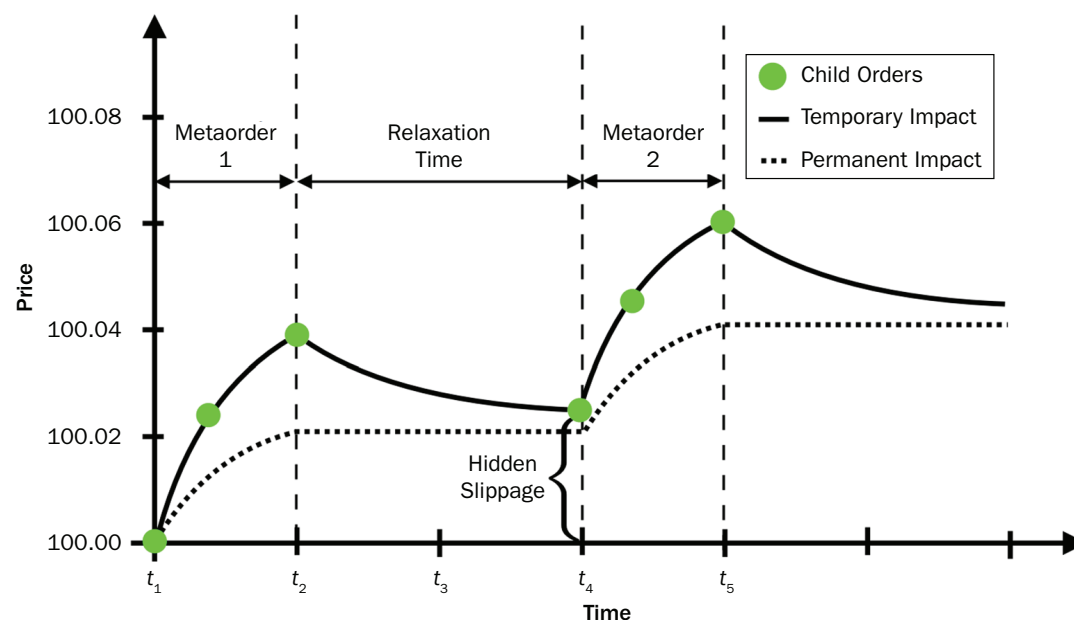
Stylized Representation of Average Permanent vs. Temporary Price Impact



NOTES: The exhibit shows the average effect of the execution of a metaorder consisting of six (child) orders, as well as the resulting average temporary and permanent price impact. The decision price (100) is the price that would be expected at each time point if there was no trading.

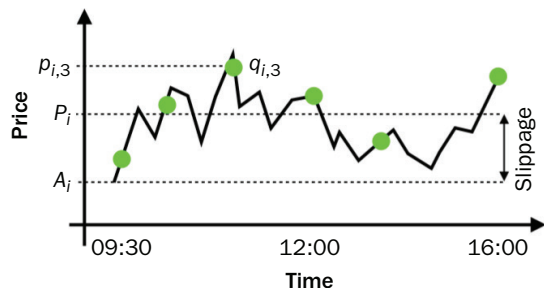
EXHIBIT 3

Execution of Two Metaorders with a Relaxation Time



NOTES: Execution of two metaorders with a relaxation time in between. The relaxation time, between t_2 and t_4 , is used in practice to allow part of the temporary impact of the first metaorder to decay. However, the decision price of the second metaorder is still affected by the impact of the first metaorder, leading to hidden slippage.

EXHIBIT 4

Execution of the i th Metaorder over Time

NOTES: This exhibit illustrates the execution of the i th metaorder over time. The arrival/decision price is denoted by A_i . The green circles indicate six buy child orders. For the k th child order of the i th metaorder, the size and price are denoted by $q_{i,k}$ and $p_{i,k}$, respectively. The volume-weighted average fill price of all child orders is denoted by P_i (e.g., the third child order has volume $q_{i,3}$ and price $p_{i,3}$).

EFFS MODEL

Model Setup

We denote the i th metaorder decision/arrival price by A_i and its size (in shares or lots) by Q_i . Note that we assume sizes to be signed (i.e., to be positive for buy orders and negative for sell orders). We also refer to buy and sell orders as orders of a different side. Execution costs are measured relative to A_i . We assume that an individual metaorder with arrival price A_i and size Q_i is broken into a sequence of K_i child orders of sizes $q_{i,k}$ (so that $Q_i = \sum_{k=1}^{K_i} q_{i,k}$) and that these are executed at prices $p_{i,k}$ for each $k = 1, \dots, K_i$ (Exhibit 4). Note that we assume that metaorders are broken down into child orders of the same side; that is, if the metaorder is a buy order, all child orders will be buy orders as well. The split of the metaorder into child orders is usually undertaken to reduce total execution costs by smoothing market participation (and thereby impact) across the available execution horizon.

The volume-weighted average fill price of the child orders of the i th metaorder is therefore given by $P_i = \text{Total amount paid} / \text{Total order size} = (\sum_k |q_{i,k}| p_{i,k}) / (\sum_k |q_{i,k}|)$. The dollar slippage or execution cost (ignoring any predetermined fees) of the i th metaorder is therefore

$$C_i = (P_i - A_i) \cdot Q_i.$$

Rather than just focusing on the cost of this trade, we are interested in the effect of the price move on future metaorders (which we call *the expected future flow* [EFF]). After execution of the metaorder is completed, the price has potentially moved and the next metaorder starts at arrival price A_{i+1} . The price move $(A_{i+1} - A_i)$ affects subsequent metaorders. We propose the following simple formula to proxy for the effects that this has on the profitability of a strategy:

$$EFFS = \underbrace{(A_{i+1} - A_i)}_{\text{Price move}} \times \underbrace{E_i \left[\sum_{j=1}^N Q_{i+j} \right]}_{\text{Expected future flow}},$$

where $E_i[\cdot]$ denotes the expected value (at the time of A_i) and N is the number of future metaorders to include in the calculation. We typically choose N so that the autocorrelation within the sequence of metaorders at this separation is close to zero, in which case the subsequent contributions to the sum become small. Here the quantity $E_i[\sum_{j=1}^N Q_{i+j}]$ is the net EFF at the arrival time of A_i . This measurement can be done ex post³ once we know the price A_{i+1} . Note that the EFFS for individual trades is very noisy. Because of this, we average it over many trades, say over the course of a month. As a result, computing the EFFS on an ex post basis, day by day,

³ Another application of the EFFS formula is where ex ante estimates of $A_{i+1} - A_i$ are available. This is, for example, the case when there is an option to choose between two execution algorithms and we know that, on average, they cause different market impact. In this situation, we may use the ex ante estimated quantities $E_i(A_{i+1} - A_i) \sum_{j=1}^{N-i} Q_{i+j}$ combined with ex ante slippage estimates for the two algorithms to inform our choice.

is sufficiently reactive and allows us to use it in real time. This allows us to use the EFFS calibrated on historical data for current trading.

We show how to implement the EFFS formula and how the estimation problem works in practice. Furthermore, in Appendix A, we present a further simplified version that approximates the EFFS formula with an example application.

Properties of the EFFS Formula

The EFFS formula is a natural generalization of the classical IS formula to the case when trades are not known in advance. Indeed, if the classical IS formula is applied to one composite metaorder consisting of individual metaorders rather than child orders, it can be rearranged as

$$\begin{aligned} \text{Composite metaorder IS} &= \sum_{i=1}^N (P_i - A_1) Q_i \\ &= \underbrace{\sum_{i=1}^N (P_i - A_i) Q_i}_{\text{Individual metaorder IS}} + \underbrace{\sum_{i=1}^N (A_{i+1} - A_i)}_{\text{Price move}} \times \underbrace{\sum_{j=1}^{N-i} Q_{i+j}}_{\text{Known future flow}}. \end{aligned}$$

The first sum in the decomposition (IS of each metaorder) is the slippage to the most recent benchmark price (which includes spread costs). The second sum consists of the terms that structurally differ from the EFFS formula. In the case of IS for the composite metaorder, the quantities Q_{i+j} and N are known. In the context of the EFFS, however, the future flow needs to be estimated, thus leading to the expectation in the EFFS formula. Furthermore, N is also unknown and needs to be chosen or estimated when computing the EFFS. In the following, we comment further on how this relates to alternative approaches such as the stitching approach.

Benchmarking each metaorder to past decision prices. Appendix B provides another perspective on the EFFS formula by the rearrangement of terms. Each metaorder is decomposed into a sum of many metaorders (potentially of different parity/sign; i.e., buy or sell), with each new metaorder having a different decision price. To give a simple example, suppose that on Monday, we expect to buy 1,000 shares on Wednesday. On Tuesday, we revise this to 1,200 shares (to be bought on Wednesday). Finally, on Wednesday we end up submitting a metaorder for 1,300 shares. We can interpret this as a sum of three metaorders: First, buy 1,000 shares benchmarked to Monday's decision price; second, buy 200 shares benchmarked to Tuesday's decision price; and third, buy 100 shares benchmarked to Wednesday's decision price. If the values in the example instead had been 1,000, 1,300, and 1,200, then the metaorder sequence would have been buy 1,000, buy 300, and sell 100.

Alpha and the EFFS Formula

The EFFS will produce a high value if the expected future flow is high and/or the price move between metaorders is large. Here we focus on the price move. A large price move can be caused by either the market impact of trading or by external events that might have been the source of the alpha of the strategy. In the former case, a reduction in the EFFS might be achieved through more efficient execution or through trading in smaller quantities. However, if the EFFS shortfall is high primarily because of the alpha of the strategy, this signifies an effect of opportunity cost, and efforts should be focused on better timing of the investment decisions across metaorders.

EXHIBIT 5

Illustration of Stitching Metaorder into Composite Metaorders



NOTES: This exhibit illustrates stitching of 16 metaorders into 5 composite metaorders. Each green (red) circle represents a buy (sell) metaorder. The green (red) lines connecting buy (sell) metaorders represent composite stitched metaorders. This involves grouping together subsequences of metaorders that are in the same trade direction. The times t_1, \dots, t_5 denote the commencement of each composite order. This example illustrates a simple reversion strategy. Given that, on average, we sell in rising markets and buy in falling markets, the IS of the five composite metaorders is negative.

Temporary and permanent market impact in the context of the EFFT formula. Estimating the effects on prices long after a metaorder has finished is challenging because the noise around the observed prices increases roughly as a square root of time. Based on several empirical studies (see Bouchaud et al. 2018), it is generally believed that the impact of a trade can be decomposed into two components, consistent with the discussion around Exhibit 2:

1. Temporary impact, which eventually decays away; and
2. Permanent impact, which does not fully decay. Sometimes the permanent component is theoretically postulated to be a linear function of trade size. However, empirical work suggests the decay behaves like a power law. A number of researchers have tried to model this impact. For example, Farmer et al. (2013) suggested that the permanent impact declines to two-thirds of the peak impact.

In practice, impact often declines very slowly over long horizons, making it hard to distinguish between temporary and permanent impact. One way to deal with this slow decay is to introduce waiting or relaxation times between metaorders to give time for temporary impact to run its course, as also illustrated in Exhibit 3. However, in practice, it is often impossible to distinguish between truly permanent and slowly decaying temporary impact.

At first glance, one might think that the EFFT is only consistent with a view of the world in which all market impact is permanent—that is, a permanent price shift arises that affects all future orders. However, as we demonstrate later in our empirical work, the EFFT formula is also consistent with temporary market impact.

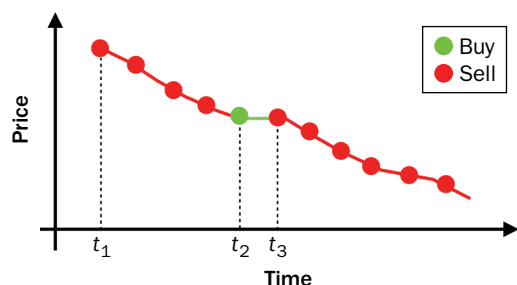
The EFFT Formula in Context of Alternative Approaches

Stitching of metaorders. Given the similarity of the EFFT formula to the classic IS formula, why not just apply the IS formula directly to metaorders? An approach popular in the industry is to do just that—stitch together several metaorders into one large composite order based on trade direction (see Gomes and Waelbroeck 2014 and references therein). Exhibit 5 presents an example: a sequence of several buy metaorders in a row (such as those denoted by the first four green circles in Exhibit 5) would be redefined as one composite metaorder, with the corresponding decision price defined as the arrival price of the first metaorder in the sequence. The standard IS formula then would be applied to this new composite metaorder.

Unfortunately, this approach is problematic in several situations. First, suppose that trades are correlated with price moves, such as in a mean reversion strategy. In this case, prices may drift down over multiple metaorders, leading to a longer sequence of buys, followed by sequences of sells when the price is moving up (as illustrated in Exhibit 5). This would likely lead to low, or even negative, IS when benchmarked to the starting metaorder price. Importantly, this shortfall may have nothing to do with market impact. Second, even for the case of independent metaorders, the IS formula applied to composite metaorders yields an extra bias on P&L, in contrast to the EFFT. Indeed, for completely independent metaorders, the IS

EXHIBIT 6

Stitched Sequence of 10 Sell Metaorders Interrupted by a Buy Metaorder



NOTES: This exhibit shows a stitched sequence of 10 sell metaorders interrupted by a buy metaorder. As in the previous exhibit, each green (red) circle represents a buy (sell) metaorder, and the times t_1 , t_2 , and t_3 denote the commencement of each composite order. The green (red) lines connecting buy (sell) metaorders represent composite stitched metaorders. Without the (small) buy order in the middle, the stitching approach would treat the entire sequence as one composite metaorder benchmarked to the decision price at t_1 . However, with the buy order, it is treated as two composite sell metaorders benchmarked at t_1 and t_3 (as well as the buy metaorder at t_2). This has a large effect on the IS measure, even though we would expect that a very small buy order would have a negligible effect on the overall impact of the sequence of (larger) sell orders.

without any stitching would be the correct measure because there is no autocorrelation. However, if, purely by chance, we observe consecutive orders of the same direction that are then stitched, the IS will be conceptually wrong. This can also be seen in the simulation study in the next section. Finally, consider the case when a perfectly correlated sequence (e.g., a very large composite trade) is interspersed with small metaorders of the opposite side (buy/sell), as depicted in Exhibit 6, a situation that often arises due to a volatility adjustment or short-term alpha. This type of pattern would drastically reduce the IS effect as measured by stitching together. In the case of Exhibit 6, the EFFS formula is more robust because the metaorder of small size yields a correspondingly small EFFS contribution.

A simple way one might think about this scenario is the following. In situations in which the future flow is known, composite metaorder IS seems like an adequate measure. When future flow is unknown, it is impossible to measure the composite metaorder IS, and that is precisely where the EFFS measure comes in. However, this intuition is not general enough. For example, if we were to know the future flow in Exhibit 6, composite metaorders would still be a poor measure, unless we as humans were to look at it and say that, rather than three composite metaorders, it should be treated as one given that the middle order is negligi-

ble. This is, however, a discretionary decision that goes beyond simple stitching. The EFFS is the systematic way to do this and to automate this process.

Propagator models. One approach to deal with correlated order flow is the propagator model (Bouchaud et al. 2004). This model makes the simplifying assumption that each metaorder can be treated separately and that the effects of each metaorder can be summed linearly. For an individual metaorder of size Q_i starting at time t_i^{start} and finishing by the time t_i^{end} , the market impact after the execution has completed is modeled as follows:

$$\text{Market Impact}(t) = H(Q_i, \sigma_D, V_D) \cdot G(t - t_i^{end}).$$

Here the function H describes the price impact at the end of the metaorder, which is commonly modeled by the square-root law as being proportional to $\sigma_D \sqrt{|Q_i|/V_D}$, where σ_D and V_D are the price volatility and the traded market volume during the time horizon of the execution. Furthermore, the (typically) monotonically decreasing function G describes the decay this impact experiences through time, a phenomenon sometimes referred to as *relaxation*.⁴

Using the propagator model, the market impact of each realized trade can be subtracted from the observed prices to obtain *cleaned prices* (i.e., the prices that would have been observed without the impact of the trading). These hypothetical prices can be used to calculate the cleaned P&L of the strategy. The difference between this cleaned P&L and the realized P&L provides a measure of the additional effect on P&L beyond slippage, similar in spirit to the quantity captured by the EFFS. It is useful to

⁴Typical examples are power-law decay $G(t) = t^{-\beta}$, $\beta > 0$ and exponential decay $G(t) = e^{-\kappa t}$, $\kappa > 0$.

contrast the EFFS formula and the propagator model in the context of the IS formula: the EFFS formula reflects both the execution cost and the opportunity cost of the IS, whereas the propagator model only reflects the execution cost.

Despite the conceptual simplicity of the propagator model, there are several hurdles to overcome to fit it. Estimation requires access to market volumes, which may not be available or may be hard to access for some over-the-counter markets, especially historically. It also typically requires a long history of trading data, with potentially millions of metaorders required for adequate estimation. This gives the propagator model limited adaptivity, thus making it ill-suited to identifying changes in execution quality. Furthermore, it requires the fitting of multiple linear regressions, which can be challenging in practice owing to the number of inputs, collinearity, hyperparameter choices, and so on. Indeed, there are even choices to be made on basic input variables, such as whether to use calendar time, sample time, or volume time. We present a numerical comparison of the propagator model and the EFFS model in the next section.

Consider the EFFS formula:

$$EFFS = (A_{i+1} - A_i) E_i \left[\sum_{j=1}^N Q_{i+j} \right].$$

The formula consists of two parts: (1) the difference of the decision prices, $A_{i+1} - A_i$, and (2) the EFF (i.e., the expected future metaorder flow $E_i[\sum_{j=1}^N Q_{i+j}]$). Recall that the expected sum of future metaorders is truncated after N terms. Intuitively, N is the number of periods/days such that the correlation of signed metaorders separated by this number of periods is negligible, so the market impact of metaorders more than N periods later is close to zero in expectation. This is a fixed hyperparameter, and later we will show how to set it.

Computing the difference in decision prices, $(A_{i+1} - A_i)$, is usually done ex post. Note that this still allows for real-time updates, in which the EFFS is evaluated at the end of each period/day. Given the level of noise typically present in trade data, one usually averages EFFS over periods of at least a month. Thus, a one-day lag due to the ex post calculation has little influence on the rolling estimate.

The bigger challenge is to estimate the EFF, given by $E_i[\sum_{j=1}^N Q_{i+j}]$, which is the subject of this section. The EFF might be deduced from the specific trading strategy, or it might need to be estimated from historical data. In the following, we consider both cases.

Once we have estimated both terms, we can simply multiply them to obtain the daily EFFS (this is done ex post, as we will explain). The sum of the EFFSs for each metaorder yields the total dollar EFFS.

Implementing the EFFS Formula

Computing the EFFS when target position or trade sequences are known in advance.

The first step in computing the EFFS, namely obtaining the daily EFF, given by $E_i[\sum_{j=1}^N Q_{i+j}]$, is the most involved. Depending on the investment decision process, it can be approached in different ways.

The simplest case is when the terminal target position (after many metaorders) is known, potentially as an output of the specific trading system. The expected future flow is

$$E_i \left[\sum_{j=1}^N Q_{i+j} \right] \approx \text{TargetPosition}_i - \text{CurrentPosition}_i - Q_i.$$

EXHIBIT 7

Computation of EFFS

Time (day)	Desired Position (lots)	Metaorder Size (lots)	Decision Price (\$)	EFF (lots)	Price Delta (\$)	EFFS (shortfall) (\$100)
1	0	10	101	20	2	40
2	10	10	103	10	-1	-10
3	20	10	102	0	3	0
4	30	0	105	0	1	0
5	30	0	106	0	-	-

NOTES: This exhibit is an example illustrating the computation of the EFFS for a strategy in which a transparent investment process allows us to infer future flow in advance. We know on day one that we are entering a position of 30 lots over the course of the next three days. Here the EFF can be obtained without estimation (given our assumptions of zero impact on future order flow after three days).

Thus, in this particular case, there is no need to estimate the expectation on the left-hand side of the equation, and therefore there is no requirement to choose a value for N . If we know the target position, why is this an approximation? Because of trading conditions and constraints, it is possible that the target is not met. For example, suppose our goal is to get to a target position of X from 0 over a period of 10 days. However, there is a constraint that the system cannot trade more than 10% of average daily volume. It is possible that it will take more than 10 days if liquidity dries up. Furthermore, we assume that the effect of the initial order Q_i on any order flow after reaching the target position is approximately zero. However, this is a very special case. In practice, we usually do not know the target position and need to resort to the approaches shown in the following.

In some cases, we not only know the target position, but also the details on how to trade into this position. Most commonly, this is the case when the alpha/signal construction process is transparent (i.e., not part of an opaque machine learning algorithm) and available to the execution agent. In this case, the expected future flow can be deduced directly. A simple example could be an event-based signal that would change its value from 0 to 30 lots in steps of 10 lots on three consecutive days/metaorders after the triggering event (one lot being 100 shares). The expected future flow (after the event and the initial 10-lot trade) would decrease from 20 lots to 0 in steps of 10 lots.

Exhibit 7 provides the details of this event-based signal example. Note that the desired position and the resulting metaorders are determined by the strategy, and we know this in advance. The decision prices are the ones observed in the market (e.g., each day's opening price). Given this information (columns 1–4), we now compute the EFFS. First, given that the sequence of metaorders is known in advance, we can infer the EFF at each day by simply summing up future metaorders, as shown in the fifth column. Next, we compute the price delta, $A_{t+1} - A_t$, in the sixth column; note that it is the delta between the next and this day's decision price. This is only available ex post; that is, we compute the price delta for a given day when observing the next day's opening. Now we can obtain the EFFS contribution on each day (seventh column) by simply multiplying the EFF (column 5) and the price delta (column 6). Because it depends on the price delta, it is also only available ex post. The total EFFS for a given period is the sum over the daily EFFS for all days in this period. For the entire sample period in the example, the EFFS is thus 30 (in units of \$100).

Consider the mechanics of Exhibit 7. The trade is triggered by an event; we know the target position is 30, and we aim to trade over three days. Again, we are abstracting from any impact beyond three days and assuming that we are unconstrained and

able to do trades of 10 per day. The initial decision price is 101, and the trader begins with a metaorder of 10. The initial trade and/or general market condition moves the price up by 2. The trader knows there are 20 more to go; hence, the EFFS is $2 \times 20 = 40$ (given that the price moved by 2 prior to execution). The next day, the trader executes 10 lots, but the price drops by 1. The trader has 10 lots to go, and the EFFS is $-1 \times 10 = -10$. The third day, the trader executes the last leg, another 10 lots, and the price goes up by 3. The total sum of EFFS is $40 - 10 = 30$. Let us for simplicity assume that the three trades filled at 103, 102, and 105.

The standard IS applied to each metaorder yields $+10 \times 2 (= 103 - 101) - 10 \times 1 (= 102 - 103) + 10 \times 3 (= 105 - 102)$, which is 40. However, benchmarking to the initial decision price (in other words, viewing it as one composite metaorder), it is $+10 \times 2 (= 103 - 101) + 10 \times 1 (= 102 - 101) + 10 \times 4 (= 105 - 101) = 70$. The difference is the EFFS. The EFFS measures the hidden impact cost, which is 30 ($= 70 - 40$). Note that benchmarking to the initial arrival price is identical to the stitching of metaorders into composite metaorders. Although this approach produces an identical answer to that using the EFFS in this simple setting, it is flawed in general, as we highlight in the discussion surrounding Exhibits 5 and 6 in the previous section.

This approach may be very effective and avoids estimation errors given that the EFF can be obtained exactly from the strategy. However, it may be challenging if the set of signals is large and diverse, and involves many different signal smoothing techniques. It may be infeasible if the signal construction is not transparent.

Data-driven estimation of the EFFS. The next approach is data driven and is arguably the most versatile, being applicable to any strategy. We use this approach for the empirical study of actual trade data.

In general terms, the idea is to use a statistical model, such as a parametric model involving some predictors, to infer the EFF given by $E_i[\sum_{j=1}^N Q_{i+j}]$. Once the model is fit to a set of historical training data, we can then make live predictions during run-time. We find that metaorder size, Q_i , is often enough to infer the EFF with reasonable accuracy. We thus use Q_i as our only predictor. Intuitively, this makes sense: large metaorders are likely to be followed by large metaorders of the same side (buy or sell), whereas small metaorders are likely to be followed by small metaorders, with a lower probability of consistency in side (so the EFF is closer to zero).⁵ Given the predictor variable, we now have to formulate the statistical model.

In practice, we have found that a piecewise constant function works well. For this, we split the range of possible values of Q_i in b intervals or buckets.⁶ The prediction for each interval is simply the historical mean over the historical future flows, $\sum_{j=1}^N Q_{k+j}$, of each order Q_k in that bucket. Mathematically, we can write this as

$$E_i \left[\sum_{j=1}^N Q_{i+j} \right] \approx \text{Mean} \left[\left\{ \sum_{j=1}^N Q_{k+j}; \text{ for all } k, \text{ such that } Q_k \text{ is in same bucket as } Q_i \right\} \right].$$

We make this model more statistically robust by exploiting the following symmetry: If a buy metaorder of size Q has an EFF of a given value x , we expect the equivalent sell metaorder of size $-Q$ to have an EFF of value $-x$. We can then group the metaorders in intervals of absolute size $|Q_i|$. To obtain an estimate of the future flow of a given order Q_i , we first identify the size bucket in which Q_i resides and denote this by b^* . Then, restricting attention to those orders Q_k in bucket b^* , we compute the mean of

⁵This is related to long-range autocorrelations in trade flow.

⁶There are different methods to partition the data in buckets. Here we use a quantile-based approach in which the boundary of the intervals/buckets is given by quantiles of the distribution of Q_i . For example, to obtain four buckets we would split at the 25th, 50th, and 75th quantiles.

the side-adjusted future flows $\text{sign}(Q_k) \sum_{j=1}^N Q_{k+j}$. Last, we then restore the side, multiplying by $\text{sign}(Q_i)$. This can formally be written as

$$E_i \left[\sum_{j=1}^N Q_{i+j} \right] \approx \text{sign}(Q_i) \text{Mean} \left[\left\{ \text{sign}(Q_k) \sum_{j=1}^N Q_{k+j}; \text{ for all } k \text{ such that } |Q_k| \text{ in same bucket as } |Q_i| \right\} \right].$$

This is the approach that we use in our empirical study in the next section. To clarify and explain each step in more detail, we also present an example of this implementation in Appendix C.

SIMULATION AND EMPIRICAL RESULTS

We now show that the EDFS formula gives estimates that are consistent with the challenging-to-implement propagator models of market impact, and it avoids the flaws of the stitching approach. First, we report the results of a simulation study with market dynamics driven by a propagator model of market impact. Second, we report the results of applying the EDFS to a set of proprietary order flow and execution data from Man Group. We then compare the EDFS and the propagator model fitted to the same data.

Results from a Simulation Study

In the simulation study, we make the simplifying assumption that the trades contain no alpha, which we believe is a reasonable approximation in practice when the holding period is long with respect to the execution horizon. That is, if the trade is designed to capture alpha that will be realized over one month and the execution happens in one day, it is safe to assume that the alpha will not all be realized on the execution day. We also assume that there are no spread costs and the trades can be completed at midprices. These simplifying assumptions ensure our focus is on market impact effects across the sequence of metaorders.

We consider 10 periods of trading—which can be thought of as days during which each metaorder is executed, except the final special point in time, which will be used to mark to market our position. Think of the final point in time as being well after the execution has been completed. The price (before allowing for impact effects) is a simple continuous random walk, with normal innovations between the start of each period and total volatility of 100 bps during each of the 10 periods.

Using a simple propagator model for the data-generating process, each trade is assumed to add a $0.1\sigma\sqrt{|Q|}$ increment to the arrival price in the direction of the trade. This is equivalent to the immediate impact function described earlier, which follows the square-root law of market impact. This initial impact is assumed to decay partially so that by the next arrival time the individual trade effect is reduced⁷ to $0.05\sigma\sqrt{|Q|}$ and then to $0.02\sigma\sqrt{|Q|}$. All trade impact reverts to zero (i.e., completely dissipates) by the final period. The final period is special—it can be thought of as a longer gap, such as year-end. In practice, marking to market the P&L at observed prices without allowing sufficient passage of time for the impact to revert may lead to overly optimistic P&L estimates. For simplicity, in this simulation, there is no permanent impact. We use a (discrete kernel) function G to represent the decay formulation, rather than a

⁷ In other words, $G(1) = 0.5$, $G(2) = 0.2$, $G(i) = 0$ for $i > 2$.

more complex power-law decay. Together, this immediate impact and decay/relaxation specification parallels the stylized behavior of the propagator model.

We consider illustrative strategies of correlated trades with an autocorrelation coefficient of 0.85. Such autocorrelation could be caused by smoothing a raw trading signal. Indeed, an exponentially weighted moving average of a raw trading signal with a half-life of one week would result in an autocorrelation of comparable size. Furthermore, we consider three different cases for the trade innovation—that is, the surprise or independent component of each trade. We can decompose every new metaorder into a sum of the expected component and the residual that is not predictable. The cases are

1. **Neutral:** The trade innovations are normal and uncorrelated with past price moves.
2. **Reversion strategy:** The correlation of the normal trade innovation (the part not predictable from past trades) to the most recent price return is -0.2 .
3. **Momentum strategy:** The correlation of the trade innovation to the most recent price return is 0.2 .

Note that the overall correlation between trades and the most recent price returns would be a relatively modest 10% (in absolute value) for the latter two cases.⁸ In general, the correlation is an outcome of the holding period of the strategy and how frequently metaorders are issued. In practice, this correlation can be higher or lower than the value chosen for our illustrations. In our Monte Carlo analysis, we generate 100,000 scenarios, each containing 10 periods as described earlier. (We can think of the 100,000 scenarios as 100 replications for each stock of a universe of 1,000 stocks, which corresponds to a realistic amount of data.)

The effect we are estimating is the expected effect on P&L by the end period. We do so by using the EDFS formula and two alternative methods: (1) benchmarking all metaorders to the first arrival price (First Arrival in Exhibit 8); and (2) metaorder stitching, as described in the previous section. We report the dollar costs divided by notional traded in basis points in Exhibit 8. In this simple setting, we can use the first implementation of the EDFS formula, in which the target position is known. Exhibit 8 shows that the EDFS formula adequately captures the P&L effect of trading. Benchmarking to the first arrival price works only for the neutral strategy case (because the expectation of the first arrival price is the same as the expected value of the final period price). The first arrival method does poorly with the reversion and momentum strategies. The stitching method gives biased results in all three cases. Notably, for the reversion strategy, both the first arrival and the stitching method estimate the wrong sign (i.e., suggesting that the trading impact would be to the benefit of P&L). The wrong sign appears because in the reversion strategy we are, on average, buying in falling markets and selling in rising markets, as also illustrated in Exhibit 5.

Results from Execution Data

In practice, the laws governing market impact and market impact decay are not readily observable and, unlike in the simulation study shown earlier, have to be estimated from real trade data. A common approach is to fit a propagator model. Once the market impact model is available, the P&L effect can be estimated by evaluating the fitted propagator model on the realized trades. Fitting a propagator model can be a challenging task for several reasons. The first challenge is data availability

⁸ Given the decomposition of a metaorder as $Q = Q_{\text{expected}} + Q_{\text{innovation}}$, for the momentum strategy we have $\text{corr}(Q_{\text{innovation}}, \text{Return}) = 0.2$, but $\text{corr}(Q, \text{Return}) \approx 0.1$ (and similarly for the reversion strategy).

EXHIBIT 8

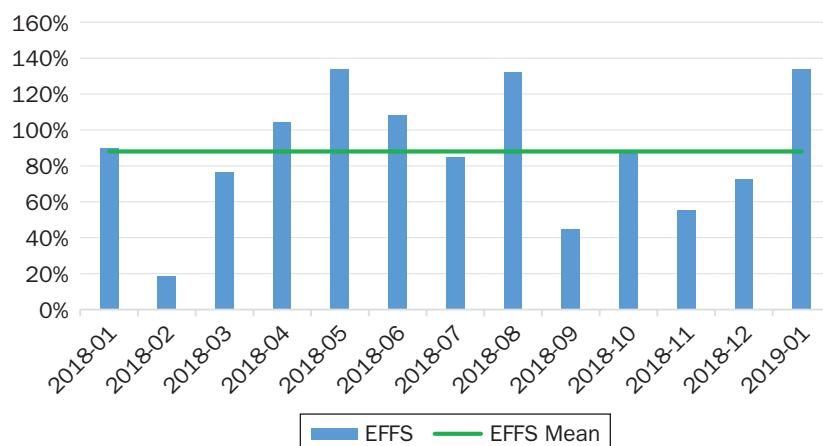
Estimated Effects on P&L

	True Effect	EFFS Model	First Arrival	Stitching
Neutral	-14.07 ± 0.3	-14.14 ± 0.2	-14.05 ± 0.3	-16.29 ± 0.3
Reversion	-13.94 ± 0.3	-14.16 ± 0.2	23.10 ± 0.3	14.16 ± 0.3
Momentum	-13.97 ± 0.3	-14.31 ± 0.2	-51.49 ± 0.3	-47.27 ± 0.3

NOTES: This exhibit shows the estimated effects on P&L (in basis points) due to long-range price impact. The results are obtained using 100,000 Monte Carlo simulations of 10 trading periods for the three strategies described earlier. We compute average effect and present \pm two standard deviations from the Monte Carlo simulations. Given that we know the underlying impact model, we can compute the true effect. Here we use the first estimation procedure explained in the previous section. Evaluation of the first arrival and stitching methods is straightforward, but neither yields consistent results.

EXHIBIT 9

EFFS Shortfall Relative to a Propagator Model



NOTES: For the period from January 2018 to January 2019, we show the monthly average shortfall computed using the propagator model and the EFFS model. Statistics are based on approximately 730,000 trades. The propagator model is normalized to 100% (only relative performance can be shared owing to the proprietary nature of the data). Hence, a value of 80% means the EFFS model returns a shortfall that is 80% of the size of the propagator model. The EFFS model gives estimates that are close to estimates from the propagator model when averaged over the entire year. Note the larger difference in the volatile month of February 2018.

(the estimates in the literature use millions of metaorders; Brokmann et al. 2015; Bucci et al. 2018, 2019). Second, this is a high-dimensional estimation problem that is confounded by a potentially significant degree of correlation in regressors. Finally, there are many choices to make regarding ways of pooling the data, choosing priors, regularization methods, and lookback periods. The EFFS method, in contrast, is considerably easier to implement and is much less parametric.

To illustrate the strengths of our method and performance relative to a propagator model approach, we report results from applying both methods to metaorders from a proprietary trading strategy. We report estimates for every month in the period January 2018 to January 2019 in Exhibit 9. To focus on relative performance, we normalize the measurements so that the propagator estimate equals 100% every month. To be clear, a value of 80% means that the EFFS has 80% of the propagator shortfall.

There are two important takeaways from Exhibit 9. First, over the entire sample period, the simple EFFS model performs similarly to the difficult-to-estimate propagator model. The average difference in dollar terms between the two models over the period is approximately 12%, only 6% if the tumultuous month February is excluded.

The performance of the EFFT in February points to the second important observation: the EFFT estimate is much more adaptive because it uses only one month of data. The propagator model, in contrast, uses approximately one year of data and thus is naturally smoother and much less sensitive to changes in the distribution of price moves and trades.

The adaptability of the EFFT method is also illustrated by computing the median of the absolute change in shortfall month on month, which is 47% for the EFFT model and only 15% for the propagator model. The relatively high variance in the EFFT estimates is exactly what we expect, given that the estimates rely on standard execution slippage measurements calculated per metaorder—which, as already noted, are extremely variable, being strongly affected by price moves during the metaorder.

One might wonder whether it is possible to make the propagator model more adaptive by using, say, only one month of data. However, as we have emphasized, the propagator model is much more complex, and it is often not feasible in practice to fit the model on small samples of data. Although each approach has its merits, the EFFT model is much more reactive during periods of sudden change, such as crises.

CONCLUSIONS

In this era of machine learning and big data, trading occurs much more frequently. As such, it is increasingly important to have the most efficient execution of trades. Although it is commonplace to divide larger metaorders into smaller child orders to reduce market impact, we investigate the case in which metaorders themselves are correlated. Here we introduce a new technique that we call the EFFT.

We provide simulation evidence in a number of different market scenarios that shows that the EFFT approach has distinct advantages over other approaches. For example, it is tempting to simply stitch together all the metaorders. However, our evidence shows that this method fails and gives counterintuitive results in periods of price reversion.

We also provide a real trading example using proprietary data. Here we compare the performance of the simple EFFT model and a propagator model. The propagator model is highly parameterized, difficult to estimate, and requires a very large amount of data to be calibrated adequately. In contrast, the EFFT model requires significantly less data, which allows it to be much more reactive to market conditions.

Our empirical analysis reveals that, over the entire sample, the EFFT model and the complex propagator model perform similarly. However, there are sharp differences when the results are examined month by month. The EFFT is much more responsive to changing market conditions. Our results show that the EFFT is more than three times as reactive when computing the median absolute change in shortfall month on month.

Being able to evaluate the EFFT and the hidden slippage of a trading strategy on a monthly basis can be important when using it to construct portfolios of strategies that are optimized on a regular basis. Let us further illustrate this use case: Imagine we want to allocate to two strategies, A and B, both with an expected annual performance of 12% before costs and both with equal volatility and tail risk. Both strategies have visible costs equivalent to 3%, leading to an apparent after-cost performance of 9%. Given the preceding information, a portfolio manager would be allocating equally to these strategies. However, using the EFFT, one can evaluate the hidden slippage, which could be 2% for strategy A and 6% for strategy B, yielding a true performance of 7% and 3%, respectively. Being able to evaluate the hidden costs using the EFFT can lead to significant improvements in portfolio allocation. Not only is the EFFT useful in portfolio construction, it is also useful in alpha research itself. Many data-driven

alphas are calibrated to maximize risk-adjusted after-cost performance. It is crucial to take all costs into account.

In summary, the theoretical properties and empirical evidence point to the benefits of deploying the EDFS formula as a simple way to estimate the trading effects on P&L beyond standard slippage estimates. This is particularly important for faster systematic strategies and those with autocorrelated order flow. Our model provides an alternative to both the stitched metaorder and propagator approaches. In particular, our model performs demonstrably better than the heuristic stitched metaorder approach. It also obtains results comparable to the propagator model, while being simpler and less data hungry, allowing it to be more adaptive to changing market conditions.

APPENDIX A

AN APPROXIMATION TO THE EDFS FORMULA

A quick estimate of the order of magnitude of EDFS can be obtained using just a few quantities that are easily empirically measured: the probability p that the next trade is the same sign as the previous one; the notionally weighted average price move to next arrival time, $\bar{Q} \cdot (A_2 - A_1)$; and the average of the signed and absolute trade size, \bar{Q} and $|\bar{Q}|$, respectively. Here the overline denotes the sample average that will serve as our expectations. The contribution to the P&L of future order flow in this case is simply:

$$EDF \text{ shortfall} \approx \frac{\bar{Q} \cdot (A_2 - A_1)}{\bar{Q}} \cdot |\bar{Q}| \frac{2p - 1}{2 - 2p}.$$

This expression would be precise for the case of a Markov chain trade-generation process with the trade signs switching randomly at each step with probability $(1 - p)$ and sizes independent of the signs. Indeed, in this case, the expected net order flow in the next step is equal to $(p - (1 - p))E|Q|$; in the next one it is $(p - (1 - p))^2 E|Q|$; and in the $(k + 1)$ -th step, it is $(p - (1 - p))^k E|Q|$. Summing the steps to infinity, we get a geometric progression with a value of $\left(\frac{2p - 1}{2(1 - p)} \right) E|Q|$.

In our simulation study, we estimate $p \approx 0.82$; hence, the future flow multiplier is approximately 1.83. With the average price move of 5.65 bps, we get an estimate of the effect (in basis points) on P&L (equal to minus EDFS) of approximately -10 bps, for all three strategies considered. Because the conditions of the simplified formula are not met precisely, the estimate is slightly biased, yielding an EDFS of 10 bps (thus an effect on P&L of -10) compared to the true -14 bps. However, it gives a reasonable estimate, and it is much better than stitching or benchmarking to the first arrival price for the reversion and momentum strategy when comparing to the estimates in Exhibit 8. In practice, the approximation provides a quick estimate that, if material, can be further investigated by using the full EDFS model via the implementation outlined in the main text.

APPENDIX B

BENCHMARKING TO MANY DECISION PRICES

If we apply the EDFS formula to a set of consecutive metaorders, we can show that it is equivalent to splitting each metaorder into many new metaorders (of potentially different sides), with each being benchmarked to one past decision price. Let us illustrate this for

two metaorders. We write the expressions for the full cost—that is, a sum of (standard) slippage and EDFS—associated with each metaorder's period:

$$\text{Cost}(1) = \text{Slippage}(1) + \text{EDFS}(1) = (P_1 - A_1) \cdot Q_1 + (A_2 - A_1) E_1 \left(\sum_{i=2}^{\infty} Q_i \right),$$

$$\text{Cost}(2) = \text{Slippage}(2) + \text{EDFS}(2) = (P_2 - A_2) \cdot Q_2 + (A_3 - A_2) E_2 \left(\sum_{i=3}^{\infty} Q_i \right).$$

Adding and subtracting $E_1(Q_2)$ to Q_2 in the second expression, summing the two expressions, and rearranging the terms yields:

$$\text{Cost}(1) + \text{Cost}(2) = (P_1 - A_1) \cdot Q_1 + (P_2 - A_1) \cdot E_1(Q_2) + (P_2 - A_2)(Q_2 - E_1(Q_2)) + \dots,$$

where ... refers to terms that do not contain Q_1 and Q_2 . This can be interpreted as benchmarking Q_1 to A_1 ; benchmarking $E_1(Q_2)$, the part of Q_2 expected at time 1, to its decision price of A_1 ; and benchmarking the surprise or unexpected part of Q_2 , equal to $Q_2 - E_1(Q_2)$, to its decision price of A_2 . Note this corresponds to the “hybrid cost” of Bordigoni et al. (2021).

This can be generalized to infinitely many past decision prices so that we arrive at a backward-looking representation of cost and EDFS.

Backward-Looking Representation

The full cost associated with trading Q_i is

$$\text{Cost} = (P_i - A_i) \cdot (Q_i - E_{i-1}(Q_i)) + \sum_{j=-\infty}^{i-1} (P_i - A_j) \cdot (E_j(Q_i) - E_{j-1}(Q_{i-1})).$$

This should be contrasted to the representation we had before the rearrangement, which we can call forward looking.

Forward-Looking Representation

Indeed, adding slippage to the EDFS formula, as derived in the main text, we obtain the full cost associated with the period i as

$$\text{Cost} = (P_i - A_i) \cdot V_1 + (A_2 - A_1) \cdot E_1 \left(\sum_{i=2}^{\infty} V_i \right).$$

This emphasizes effect of the price action during period i on the future flow.

APPENDIX C

AN EXAMPLE ILLUSTRATING THE DATA-DRIVEN IMPLEMENTATION OF THE EDFS FORMULA

Let us recall that as inputs we have historical data of sequences of decision prices, A_i , and metaorders, Q_i . Furthermore, we have two hyperparameters that we must choose: N , the number of future metaorders to compute the EFF, given by $\sum_{j=1}^N Q_{i+j}$; and b , the number of buckets. In practical applications, values around $N = 30$ (e.g., 30 days or approximately one month) serve as a good starting point. If one is unsure, one can start

EXHIBIT C1

Example Illustrating the Data-Driven Approach to Compute the EFFS

Time (day)	Metaorder Size (lots)	Decision Price (\$)	Absolute Size Bucket (lots)	Side-Adjusted Future Flow (lots)	EFF Est. (lots)	Price Delta (\$)	EFFS Est. (100\$)
1	16	100	[10–20)	8	7	3	21
2	8	103	[0–10)	–1	1.5	2	3
3	–1	105	[0–10)	–1	–1.5	–2	3
4	1	103	[0–10)	–3	1.5	2	3
5	–3	105	[0–10)	11	–1.5	–2	3
6	–11	103	[10–20)	6	–7	–3	21
7	–6	100	[0–10)	–	–1.5	–	–

NOTE: The steps to arrive at each column are explained in the main text.

with a lower value and increase N , stopping when the final value of the EFFS does not change materially upon increasing N (i.e., it has converged). This means that any given order has diminishing influence on any future orders beyond N steps ahead, thus including those would not change the result, which is why we truncate the sum at N , rather than, say, summing until the end of the date sample. Regarding b , in practice, using 10 buckets is a good rule of thumb. If enough data are available, one can use cross-validation to determine the optimal hyperparameters.

In the example in Exhibit C1, we choose $N = 1$ and $b = 2$ (two buckets) for simplicity. With $N = 1$, we only look one step ahead for the future flow estimation. This is less realistic than the typical choice of $N = 30$, but it is straightforward to extend this to any arbitrary N and any number of buckets. Given the historical sequence of metaorders and decision prices for each day (column 2 and 3), we now go through each step to arrive at the EFFS. The first step of the computation is to assign each metaorder to an absolute size bucket and compute its side-adjusted (i.e., multiplied by the metaorder sign) future flow.

We choose two buckets, one for metaorders with absolute size between 0 and 10 and one for orders with absolute size between 10 and 20. We now assign each metaorder to its corresponding bucket, resulting in the fourth column. Next, we compute the side-adjusted future flow for each metaorder: For each metaorder, we simply sum the sizes of the N subsequent orders and multiply this by the sign of the original metaorder. The resulting values for $N = 1$ are shown in the fifth column.⁹

The next step is to compute the EFF estimate as an average over absolute size buckets. The first bucket, [0–10), contains four metaorders for which we computed the side-adjusted future flow (rows 2–5) with values –1, –1, –3, 11. The average is thus

$$\text{Side adjusted future flow for bucket } [0,10) = \frac{-1 - 1 - 3 + 11}{4} = 1.5.$$

In the same way we obtain

$$\text{Side adjusted future flow for bucket } [10,20) = \frac{8 + 6}{2} = 7.$$

To obtain the estimated EFFS for each metaorder, we simply take the average side-adjusted future flow for the corresponding bucket and multiply it by the sign of the original metaorder. The resulting values are shown in the sixth column. The remainder of the computation proceeds as in the earlier example: We take the price (column 3) and compute

⁹ If N were 2, then the first value would have been $8 - 1 = 7$.

the price delta (column 7). Next, we multiply the price delta (column 7) by the estimated EFF (column 6) to obtain the EFFS for each metaorder (column 8). Summing over the EFFS for each metaorder during the whole period yields the total EFFS estimate for the period, which in this case is 54 (in units of \$100).

The mechanics are as follows. The initial decision price is 100, and the trader put in an order for 16. Given the order size of 16, one can estimate that the future order will be 7 (based on a historical calibration with buckets—but in this case, we are using the actual data to do this). Given the metaorder moved the price by 3, we multiply 7 times 3 to get the EFFS estimate of 21 in the final column of Exhibit C1. Day 2 arrives, and the decision price is 103. The metaorder is 8. Given the order of 8, we try to estimate the future order. The historical calibration for this bucket is 1.5. Given the price moved by 2, the EFFS is 3 (in the eighth column). On day 3, the decision price is 105. The metaorder is of size −1. Given the −1 order, we now expect the next trade to be of size −1.5, given we are in the small size bucket and selling. The price dropped on the first sell by 2; the EFFS is 3. This continues through day 7. The total EFFS is just the sum of each individual day's EFFS.

This example should allow the reader to quickly implement the EFFS model. Note that given the low signal-to-noise ratio, one generally requires thousands to tens of thousands of observations to obtain robust estimates.

ACKNOWLEDGMENTS

We appreciate the comments of Morad Elsaify, Frank Fabozzi, Ed Hoyle, and Otto van Hemert.

REFERENCES

- Almgren, R., and N. Chriss. 1999. "Value under Liquidation." *Risk* 12: 61–63.
- . 2001. "Optimal Execution of Portfolio Transactions." *Journal of Risk* 3: 5–40.
- Almgren, R., C. Thum, E. Hauptmann, and H. Li. 2005. "Direct Estimation of Equity Market Impact." *Risk* 18 (7): 58–62.
- Bertsimas, D., and A. Lo. 1998. "Optimal Control of Execution Costs." *Journal of Financial Markets* 1: 1–50.
- Bertsimas, D., A. Lo, and P. Hummel. 1999. "Optimal Control of Execution Costs for Portfolios." *Computational Finance* 40 (6): 40–53.
- Bordigoni, G., A. Figalli, A. Ledford, and P. Ustinov. "Strategic Execution Trajectories." Working paper, 2021.
- Bouchaud, J. P., J. Bonart, J. Donier, and M. Gould. *Trades, Quotes and Prices: Financial Markets under the Microscope*. Cambridge, UK: Cambridge University Press, 2018.
- Bouchaud, J.-P., Y. Gefen, M. Potters, and M. Wyart. 2004. "Fluctuations and Response in Financial Markets: The Subtle Nature of Random Price Changes." *Quantitative Finance* 4 (2): 176–190.
- Brokmann, X., E. Serie, J. Kockelkoren, and J. P. Bouchaud. 2015. "Slow Decay of Impact in Equity Markets." *Market Microstructure and Liquidity* 1 (2): 1550007.
- Bucci, F., M. Benzaquen, F. Lillo, and J. P. Bouchaud. 2018. "Slow Decay of Impact in Equity Markets: Insights from the ANcerno Database." *Market Microstructure and Liquidity* 4 (03n04): 1950006.
- . 2019. "Crossover from Linear to Square-Root Market Impact." *Physical Review Letters* 122 (10): 108302.

- Engle, R., R. Ferstenberg, and J. Russell. 2012. "Measuring and Modeling Execution Cost and Risk." *The Journal of Portfolio Management* 38 (2): 14.
- Farmer, J. D., A. Gerig, F. Lillo, and H. Waelbroeck. 2013. "How Efficiency Shapes Market Impact." *Quantitative Finance* 13 (11): 1743–1758.
- Gomes, C., and H. Waelbroeck. 2014. "Is Market Impact a Measure of the Information Value of Trades? Market Response to Liquidity vs. Informed Trades." *Quantitative Finance* 15 (5): 773–793.
- Grinold, R. C., and R. N. Kahn. *Active Portfolio Management*. New York: McGraw-Hill, 1999.
- Kyle, A. S., and A. A. Obizhaeva. "The Market Impact Puzzle." SSRN, 2018, <https://ssrn.com/abstract=3124502>.
- Lopez de Prado, M. *Advances in Financial Machine Learning*. Hoboken: Wiley, 2018.
- Moro, E., J. Vicente, L. G. Moyano, A. Gerig, J. D. Farmer, G. Vaglica, F. Lillo, and R. N. Mantegna. 2009. "Market Impact and Trading Profile of Hidden Orders in Stock Markets." *Physical Review E* 80: 066102.
- Perold, A. F. 1988. "The Implementation Shortfall: Paper versus Reality." *The Journal of Portfolio Management* 14 (3): 4.
- Torre, N., and M. Ferrari. *Market Impact Model Handbook*. Berkeley: BARRA, Inc., 1997.