Multidimensional Latency for Queue Tomography

Aryan Ayyar

December 2025

Contents

Chapter 1

The Range to Target

"Reverify our range to target. ONE PING only."

—Captain Marco Ramius

In the cold waters of the North Atlantic, a Soviet submarine captain faces an existential problem: he must determine his distance to an unseen target using the most basic of tools — a sound wave sent into the darkness, waiting for the echo to return. One ping only. Not because restraint is virtue, but because every acoustic transmission risks exposure. Every signal sent is a declaration of presence. Yet without that signal, the darkness yields nothing. The parallel to modern financial markets is not immediately obvious, but it is precise. In the limit order books that govern global equity trading, a similar darkness persists. Institutional investors and market makers stand at a precipice, needing to know how deep the visible queue of liquidity truly runs before executing large orders. Yet the most critical information—the location of hidden reserves concealed behind iceberg orders, which remains acoustic silence. The order book shows a picture of surface depth, but beneath lies an invisible structure, a landscape of dormant volume that reshapes execution dynamics in ways that traditional models cannot capture.This chapter establishes that parallel through analogy, not through mathematics. We begin with how sonar works, how it distinguishes between active and passive measurement, and how tomographic principles allow us to reconstruct hidden structures from acoustic echo patterns. We then trace the evolution of limit order books as a mechanism for price discovery and execution, showing how these modern electronic markets can be understood as queuing systems with a peculiar constraint: most of the queue is invisible. Finally, we introduce the core intuition behind our solution: by transmitting "pings"—carefully constructed probe orders—into the limit order book and listening for the "echoes" of execution, we can reconstruct the true depth of the hidden queue, much as sonar reveals the shape of a submarine.

1.1 Measurement as a Form of Revelation

To understand why measurement itself becomes an act of revelation, we must first appreciate how information is extracted from darkness. Sonar, derived from the term "sound navigation and ranging," represents humanity's most mature technology for detecting objects in opaque media. Unlike light, which is absorbed and scattered in water, sound propagates with remarkable clarity across the ocean, allowing acoustic waves to travel thousands of kilometers. Yet this clarity comes with a fundamental constraint: one cannot simply listen passively to find an object. Silence tells you nothing. Consider the distinction between two approaches.

1.1.1 Passive Measurement

The first is passive sonar, in which a submarine operator listens to ambient acoustic noise—the propeller signatures of distant ships, the creaks and groans of distant submarines, the low-frequency calls of whales, the industrial hum of ocean-floor geological activity. From these signals, one can infer the presence and rough location of distant objects. The passive approach is silent; it reveals the listener's position to no one. But it provides only fragmentary information. A ship's engine signature might be old, refracted through multiple thermal layers, degraded by distance. The listener cannot verify distance directly; only experience and pattern recognition suggest proximity.

1.1.2 Active Measurement

The second approach is active sonar, exemplified by naval radar and sonar systems. Here, the observer sends out a signal—a pulse of acoustic energy and listens for the reflection. If an object exists at range r, the signal will bounce off that object and return to the transmitter. Given the speed of sound in water, $c \approx 1500$ meters per second, and the round-trip time τ of the reflected pulse, the range follows immediately: $r = c\tau/2$. This is direct measurement, mathematically unambiguous. But it carries a price: the outgoing pulse announces the observer's presence and intentions to anyone in the surrounding ocean. It is an act of revelation that demands commitment. The submarine captain in Clancy's opening chooses active sonar for precisely this reason. In a moment where stealth is no longer possible, i.e. where the target is close enough that the outcome will be determined by who sees first, the captain chooses the certainty of active measurement over the ambiguity of passive observation. A single ping, aimed at maximum range, provides unambiguous information about distance and bearing. That information justifies the exposure.

1.2 Reconstructing Invisible Structure

The insight that measurement can reveal hidden structure extends beyond simple distance determination. In medical imaging, computed tomography represents one of the most powerful diagnostic tools precisely because it operates on a principle of reconstructing three-dimensional structure from many one-dimensional measurements. A CT scanner sends X-rays through a patient's body from multiple angles. Each ray is absorbed differently depending on the tissue

density along its path. By collecting many such measurements from many angles, and applying mathematical inversion techniques, the scanner reconstructs a complete three-dimensional image of internal structures. No individual ray tells the full story; collectively, they reveal what cannot be seen directly. The principle underlying tomography is fundamentally one of conservation. Consider a simple example. Suppose we send a ray of light through a two-dimensional cross-section of an object, from left to right. The total absorption of that ray depends on the integrated density of material along the ray's path. Now suppose we send another ray from a different angle—perhaps from top to bottom. Again, absorption reveals something about the integrated density along that new path. With enough rays from enough angles, the pattern of absorptions becomes an overdetermined system, and the distribution of density can be inverted uniquely.

The same principle applies to acoustic measurement. When sonar sends a pulse toward a distant object, the echo that returns carries information not just about distance but about the shape and composition of the reflecting surface. If we send multiple pulses, varying in frequency and arrival time, the pattern of reflections begins to reveal detailed structure. Submarines use this principle in modern active sonar to construct not just a point estimate of distance but a full acoustic image of the ocean floor, revealing canyons, ridges, and underwater geological features. The key is variation: different measurement angles or timings create different patterns of reflection, and these patterns, when combined, yield more information than any single measurement could provide.

1.3 Limit Order Books as Queues

To apply these sonar principles to financial markets, we must first understand how modern markets came to be structured as electronic limit order books, and why this structure naturally admits a queue-theoretic interpretation. For most of financial history, the matching of buyers and sellers occurred through a human intermediary: a market maker standing on an exchange floor, facilitating trades through voice negotiation and hand signals. The NYSE in the late twentieth century still relied on specialists who physically stood at a post on the exchange floor, maintaining order books on paper and on chalk boards, matching buy and sell orders by voice and informal convention. This process had inherent limitations. The flow of information was slow, constrained by the speed at which a human could process information and communicate intentions. The order book itself was not fully transparent; the specialist could see the full depth of interest, but did not have obligation to reveal all of it to the public.

1.3.1 Electronic Limit Order Book

The transition to electronic markets fundamentally changed this structure. Beginning in the 1970s with electronic communication networks (ECNs) in equities, and accelerating with the NASDAQ stock market, trading moved from physical floors to computer networks. With this transition came the limit order book: a data structure that maintains a queue of unfilled limit orders, sorted by price and arrival time, with the highest-priority orders at the front of each

queue. When a market order arrives, it executes against the best-priced limit orders in queue order, removing shares as it progresses down the queue. The introduction of fully electronic limit order books revealed a simple but profound insight: the problem of order execution in markets with price-time priority is structurally identical to the classical queuing problem studied in operations research. Consider the perspective of an institutional trader who wishes to execute a large order. The trader can either (a) send a market order, which executes immediately but pays the spread, or (b) send a limit order, which sits in queue hoping to execute as other orders deplete the queue above it. The execution time depends on how deep the order sits in the queue, how fast that queue depletes (determined by the arrival rate of market orders and the cancellation rate), and when new orders arrive to cut ahead (which happens when the security's price moves).

1.4 The Beauty of Queues

The formal structure of this problem maps directly onto M/M/1 queue theory. Arrivals of market orders can be modeled as a Poisson process with rate λ_M. Cancellations of limit orders follow some rate λ_C. The queue depth Q at any moment determines the wait time for a new order. Standard queueing results tell us that the expected wait time is $E[\tau] = Q/(\lambda_M + \lambda_C)$. Traders understood this intuitively long before formal analysis: the deeper the queue, the longer the wait; the faster orders clear, the faster execution follows.

1.4.1 Queues Became Invisible

Yet here emerges a critical complication that breaks the simple queueing model. Modern markets provide real-time information about the visible order book—typically the best ten price levels on each side—to market participants through data feeds. But market participants can also hide their true intent through *iceberg orders*. An iceberg order is a large standing order, the vast majority of which is concealed. Only a small "visible portion" appears in the public order book at any moment. As that visible portion executes, the exchange automatically replenishes it from the hidden reserve, maintaining the appearance of a shallow queue while a much deeper hidden queue actually exists. Iceberg orders emerged as a rational response to the information leakage problem. When an institutional investor places a very large visible order in the market, that visibility itself becomes information. Other market participants can see the order size and infer the participant's intentions. This inference creates adverse selection: other traders can estimate that the institutional participant likely faces inventory pressure and begin to price ahead of that assumed demand, widening spreads and increasing execution cost. By hiding most of the order, the institutional participant disguises their true demand, reducing adverse selection. But this concealment creates a new problem: it makes the true queue invisible to other participants. From a queue-theoretic perspective, the true queue depth Q_{true} that determines execution time is no longer equal to the visible queue depth Q_{visible} observable from public market data. Instead, we have

$$Q_{\text{true}} = Q_{\text{visible}} + H \tag{1.1}$$

where $H \geq 0$ represents the hidden volume concealed in iceberg orders. For a trader placing a new order, this hidden volume is invisible. The trader observes Q_{visible} and estimates execution time as $\widehat{\tau}_{\text{naive}} = Q_{\text{visible}}/(\lambda_M + \lambda_C)$. But the true execution time will be $\tau_{\text{true}} = Q_{\text{true}}/(\lambda_M + \lambda_C)$, which is larger by a factor of $(1 + H/Q_{\text{visible}})$ whenever hidden orders exist. This opacity represents the core economic problem. In modern limit order books, the fundamental state variable that determines execution quality is no longer fully observable. Large institutional investors and market makers make execution decisions based on incomplete information. Execution algorithms that assume perfect liquidity observability will systematically underestimate execution latency, leading to inefficient time-weighted average price (TWAP) and volume-weighted average price (VWAP) algorithms, and ultimately to higher implementation shortfall (the difference between the price at decision time and the actual execution price weighted by volume).

1.4.2 Limit Order Book Tomography

Having established both the queue-theoretic structure of limit order books and the invisibility problem created by iceberg orders, we can now articulate the parallel to sonar measurement. The fundamental insight is this: *just as a submarine operator must send an acoustic signal into darkness to measure distance unambiguously, a trader or execution algorithm must send a signal into the limit order book to measure the true queue depth.* The mechanism is straightforward, and we introduce it here through analogy rather than mathematics. First, a critical preliminary observation: in modern limit order books operating under price-time priority, orders at a given price level execute in strict First-In-First-Out (FIFO) order. This is the electronic market's equivalent of the disciplined queue in classical service theory. When a new order arrives at a price where others are already waiting, it joins the back of the queue. When market orders arrive, they execute against the front of the queue. This discipline is essential to what follows. Now consider the sonar analogy more carefully. A submarine sends an acoustic pulse at time t_1 and receives an echo at time t_2. The round-trip time $\tau = t_2 - t_1$ encodes information about distance. If the ocean were empty, the echo would return instantly (or not at all). If the ocean is full of objects at various distances, the echo reflects off the nearest object, returning at a time proportional to that distance. Apply this structure to the limit order book. We "send a pulse" by submitting a limit order P_1 of unit size at the best bid price at time t_1. This order joins the end of the queue at that price level. We then wait for the "echo", the time T_1 at which our order executes, meaning all orders that were ahead of it in queue have been removed. The quantity $T_1 - t_1$ tells us something about the queue depth at that moment.

But a single pulse tells us only about the queue depth at the moment of submission. Just as sonar practitioners send multiple pulses to build up a detailed acoustic image, we can submit a second probe order P_2 at a slightly later time $t_2 = t_1 + \delta$, where δ is a small, controlled time gap. This second order also joins the end of the queue—but now the queue has evolved. New limit orders may have arrived, adding to the queue depth. More importantly, market orders and cancellations have also occurred, removing orders from the front of the queue. When P_2 executes at time T_2, the interval $T_2 - T_1$ encodes information not about absolute queue depth, but about the queue depth evolution over the interval $[t_1, t_2]$. Here is where the tomographic principle

enters. The distance between P_1 and P_2 in the queue can be decomposed into two components: the visible volume that arrived between their submission times (which we can observe from the market data feed), and the hidden volume from iceberg orders (which we cannot observe). The conservation principle, which we will formalize mathematically in Chapter Two states that the volume removed from the queue between the two execution times must equal the sum of these two components. Stated intuitively: if we remove P_1 and P_2 from the queue, the total amount of additional volume that must be depleted to get from P_1 filling to P_2 filling is exactly the volume that stood between them when P_2 was submitted. This depletion is observable from the market tape: we can count every trade and every cancellation that occurred at that price level between T_1 and T_2. By comparing observable depletion to observable visible arrivals, we infer the hidden volume.

1.4.3 From Analogy to Precision

The sonar metaphor illuminates the core insight, but it necessarily simplifies. Real sonar must contend with thermal layers in the ocean that refract sound, with multiple reflections that create false echoes, with ambient noise that obscures the signal. Real limit order book probing must similarly account for complications: order flow imbalance that modulates the rate at which the queue depletes, regime changes in hidden liquidity usage by institutional traders, the possibility that large market orders arrive and hit both probes simultaneously, the impact of latency on the precision of our measurements. These complications do not invalidate the principle; they require that we sharpen it. In Chapter Two, we will formalize the sonar metaphor through mathematics, introducing the conservation law that justifies our measurement, deriving the exact form of the iceberg density estimator, and accounting for order flow imbalance through normalized latency. In Chapter Three, we will validate these derivations against historical market data from the LOBSTER database, showing that the algorithm recovers hidden queue structure with high precision across different market regimes. But before we undertake that mathematical formalization, it is essential to understand the intuition. The submarine captain utters “ONE PING only” not because one ping is theoretically optimal, but because commitment to a single, well-designed measurement is more valuable than many tentative, cautious attempts. In limit order book tomography, we apply the same principle: by committing to a sequence of carefully structured probes, two limit orders submitted milliseconds apar we extract information about the hidden queue structure that passive observation cannot reveal. We transform darkness into measurement. We turn an invisible queue into visible knowledge. In the chapters that follow, we make this intuition rigorous. But the intuition itself—that active measurement in a queue behaves like sonar in an ocean, that conservation of volume in a FIFO queue reveals hidden structure, that the echo of execution timing encodes information about invisibility—this intuition is the foundation upon which everything that follows rests.

Chapter 2

One Ping Only

2.1 Theoretical Foundation

We formalize the limit order book (henceforth LOB) dynamics under the assumption of a Price-Time Priority matching engine, where orders at a given price level are executed according to a First-In-First-Out discipline.

2.1.1 The Queue Geometry and Price-Time Priority

The state of the queue at price p and time t, denoted by $Q(t,p)$, evolves as a stochastic process driven by the interplay of liquidity provision as well as consumption. Specifically, the queue length is governed by the net aggregate of limit order arrivals, market order executions, and cancellations, given by

$$Q(t,p) = \sum_i v_i^L \mathbb{I}_{\{t_i^L \leq t\}} - \sum_j v_j^M \mathbb{I}_{\{t_j^M \leq t\}} - \sum_k v_k^C \mathbb{I}_{\{t_k^C \leq t\}} \tag{2.1}$$

where v_i^L, v_j^M, and v_k^C represent the volumes of the i-th limit order, j-th market order, and k-th cancellation, respectively. Under PTP, the queue position is the primary determinant of execution quality. An order's position governs its execution probability, as front-of-queue orders are filled prior to those at the back; it dictates adverse selection risk, as orders deeper in the queue are more exposed to toxic flow and "picking-off" risks; and it defines the expected fill rate, directly impacting the opportunity cost of waiting. Crucially, the observable queue $Q_{visible}(t,p)$ reported by market data feeds is often a strict subset of the true liquidity available. The true queue depth $Q_{true}(t,p)$ accounts for hidden liquidity, commonly referred to as "iceberg" or reserve orders, such that:

$$Q_{true}(t,p) = Q_{visible}(t,p) + H(t,p) \tag{2.2}$$

where $H(t,p) \geq 0$ represents the latent volume concealed from the public tape. To estimate this latent component, we introduce a probe-based active inference mechanism. First, we submit two sequential limit orders, P_1 and P_2, both of unit quantity, at the best bid(resp. ask) price p. P_1 is submitted at $t = 0$, and P_2 is submitted after a deterministic latency δ, at $t = \delta$. Between the submission of P_1 and P_2, the visible queue expands due to new limit order arrivals. We define

the *Gap Volume*, V_{gap}, as the cumulative visible volume added to the queue during the interval $[0, \delta]$. Consequently, the true distance between the queue positions of P_1 and P_2, denoted q_1 and q_2, is the sum of the visible gap volume and the unknown hidden volume accumulated in that interval:

$$q_2 - q_1 = V_{gap} + H_{gap} \tag{2.3}$$

where H_{gap} is the hidden volume added between the probes. The core objective of the Multidimensional Latency Tomography algorithm is to recover H_{gap} by analyzing the differential execution times of P_1 and P_2.

2.1.2 Multidimensional Intensity Framework

We model the order flow as a multivariate point process $N_t = (N_t^L, N_t^C, N_t^M)$, representing the counting processes for limit orders, cancellations, and market orders, respectively. The dynamics of these processes are characterized by their conditional execution intensities $\lambda(t)$, defined as the expected arrival rate conditioned on the filtration $\mathcal{F}_t$ of market history:

$$\lambda^X(t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N_{t+\Delta t}^X - N_t^X \mid \mathcal{F}_t]}{\Delta t}, \quad X \in \{L, C, M\} \tag{2.4}$$

Following the microstructure models of Cont et al. (2010) and Bacry et al. (2015), we specify these intensities using a Hawkes process framework to capture the self-exciting and cross-exciting nature of order book events. The intensity for event type i is given by a baseline intensity μ_i augmented by a convolution of past events with an excitation kernel ϕ_{ij}:

$$\lambda_i(t) = \mu_i + \sum_j \int_0^t \phi_{ij}(t-s) dN_j(s) \tag{2.5}$$

For computational tractability, we employ an exponential decay kernel $\phi_{ij}(u) = \alpha_{ij} e^{-\beta_{ij} u}$, which allows for efficient recursive estimation of the intensities. The effective rate of queue depletion, which drives the execution of our probe orders, is the aggregate intensity of volume-removing events:

$$\lambda_{depletion}(t) = \lambda^M(t) + \lambda^C(t) \tag{2.6}$$

2.1.3 Conservation and Tomographic Estimator

The theoretical anchor of the MDLT algorithm is a conservation law relating observable time intervals to latent volume. Let T_1 and T_2 denote the execution times of probes P_1 and P_2, respectively. Since P_2 cannot execute until all orders preceding it, both visible and hidden have been removed, the total volume depleted from the queue during the interval $[T_1, T_2]$ must exactly equal the volume standing between P_1 and P_2. We define the Observed Depletion, D_{obs}, as the cumulative volume of market orders and cancellations recorded on the public tape at price p between T_1 and T_2. This yields the fundamental conservation equation:

$$D_{obs} = \int_{T_1}^{T_2} (dM_t + dC_t) = V_{gap} + H_{gap} \tag{2.7}$$

rearranging this identity allows us to solve for the unobservable hidden volume H_{gap} in closed form:

$$H_{gap} = D_{obs} - V_{gap} \tag{2.8}$$

From this, we derive the *Iceberg Density Coefficient*, ρ, which quantifies the ratio of hidden to visible liquidity in the local order book:

$$\rho = \frac{H_{gap}}{V_{gap}} = \frac{D_{obs}}{V_{gap}} - 1 \tag{2.9}$$

A value of $\rho \approx 0$ indicates a transparent order book, while $\rho > 0$ signals the presence of iceberg orders. This coefficient is then used to construct the MDLT Priority Metric, Q_{MDLT}, a rigorous estimate of the true effective queue position facing a new limit order:

$$Q_{MDLT}(t) = Q_{visible}(t) \cdot (1 + \bar{\rho}_t) \tag{2.10}$$

where $\bar{\rho}_t$ is an exponentially weighted moving average of the iceberg density, smoothing out microstructure noise. This metric provides a corrected input for optimal execution algorithms, replacing the naive $Q_{visible}$ with a latency-adjusted measure of queue priority.

2.1.4 Probe Order Placement

To actively interrogate the queue structure, we employ a differential latency measurement technique using paired probe orders. Let the current time be t_0. We define a probe pair as a sequence of two limit orders, denoted P_1 and P_2, submitted to the same side of the book (e.g., best bid) with identical unit quantity size $s_p = 1$. The submission mechanism follows a strict temporal discipline:

1. **Probe P_1:** Submitted at time $t_1 = t_0$. Upon acceptance by the matching engine, it is assigned a queue position $q_1 = Q_{true}(t_1, p) + 1$.

2. **Probe P_2:** Submitted at time $t_2 = t_0 + \delta$, where $\delta > 0$ is a deterministic inter-arrival gap. Upon acceptance, it is assigned a queue position $q_2 = Q_{true}(t_2, p) + 1$.

During the interval $(t_1, t_2]$, the queue dynamics continue to evolve. New limit orders may arrive, adding to the visible depth, while hidden orders (icebergs) may also be injected into the queue. We define the *Visible Gap Volume*, V_{gap}, as the cumulative size of all visible limit orders arriving at price p between the two probe submissions:

$$V_{gap} = \sum_k v_k^L \cdot \mathbb{I}_{\{t_1 < t_k^L \leq t_2\}} \tag{2.11}$$

Similarly, let H_{gap} denote the unobservable hidden volume arriving during this same interval. The fundamental geometric relationship between the queue positions of the two probes is thus:

$$q_2 - q_1 = V_{gap} + H_{gap} \tag{2.12}$$

This equation establishes that the "distance" between our probes in the execution queue is strictly equal to the sum of visible and hidden liquidity added during the inter-arrival latency δ.

2.1.5 Iceberg Density Ratio

We invoke the principle of volume conservation to derive the hidden liquidity parameters. Let T_1 and T_2 denote the stochastic execution timestamps of probes P_1 and P_2, respectively. Under the assumption of a FIFO matching algorithm, P_2 executes only after all orders preceding it in the queue have been depleted. Therefore, the total volume removed from the book between T_1 and T_2 must exactly match the queue volume standing between the two probes. We define the *Observed Depletion*, D_{obs}, as the integral of the order flow depletion rate (market orders and cancellations) over the execution interval $[T_1, T_2]$. Since market data feeds report these trades and cancellations explicitly, D_{obs} is a fully observable quantity:

$$D_{obs} = \int_{T_1}^{T_2} (\lambda^M(t) + \lambda^C(t))\, dt = \sum_j v_j^M \mathbb{I}_{\{T_1 \le t_j^M \le T_2\}} + \sum_k v_k^C \mathbb{I}_{\{T_1 \le t_k^C \le T_2\}} \tag{2.13}$$

By equating the volume depleted to the volume separating the probes, we obtain the *Conservation Law of Queue Tomography*:

$$D_{obs} = q_2 - q_1 = V_{gap} + H_{gap} \tag{2.14}$$

This identity allows us to isolate the unknown latent variable H_{gap}. Rearranging Equation (2.14), we solve for the hidden volume:

$$H_{gap} = D_{obs} - V_{gap} \tag{2.15}$$

To generalize this finding across different market regimes and asset classes, we define the *Iceberg Density Coefficient*, ρ, as the ratio of hidden volume to visible volume added. This dimensionless metric normalizes the hidden liquidity relative to the observable order flow:

$$\rho = \frac{H_{gap}}{V_{gap}} \tag{2.16}$$

Substituting the expression for H_{gap}, we arrive at the operational formula for the MDLT estimator:

$$\rho = \frac{D_{obs} - V_{gap}}{V_{gap}} = \frac{D_{obs}}{V_{gap}} - 1 \tag{2.17}$$

Hence, by observing only public data (D_{obs} and V_{gap}), we can recover the scalar parameter ρ that characterizes the hidden depth of the limit order book. A value of $\rho \approx 0$ implies $D_{obs} \approx V_{gap}$, consistent with a fully lit market. Conversely, $\rho > 0$ provides a direct measure of dark liquidity intensity.

The Priority Metric

While the iceberg density coefficient ρ provides an instantaneous snapshot of hidden liquidity, raw measurements derived from individual probe pairs are subject to stochastic microstructure noise, arising from latency jitter and transient liquidity fluctuations. To construct a robust estimator suitable for algorithmic execution, we employ an Exponentially Weighted Moving Averag to smooth the density sequence. Let ρ_k denote the raw density estimate derived from the k-th probe pair. The smoothed density state variable, $\bar{\rho}_k$, evolves according to the recursive filter:

$$\bar{\rho}_k = \alpha\rho_k + (1-\alpha)\bar{\rho}_{k-1} \tag{2.18}$$

where $\alpha \in (0,1)$ is the decay factor controlling the memory of the estimator. A higher α increases responsiveness to regime shifts in hidden liquidity usage, while a lower α enhances stability against measurement noise.

We define the *MDLT Priority Metric*, denoted as $Q_{MDLT}(t,p)$, as the effective queue position adjusted for this latent volume. This metric transforms the observable queue depth reported by the exchange into a "virtual" queue depth that reflects the true liquidity barrier facing a new limit order. For a visible queue size $Q_{visible}(t,p)$, the effective position is given by:

$$Q_{MDLT}(t,p) = Q_{visible}(t,p) \cdot (1+\bar{\rho}_t) \tag{2.19}$$

This formulation implies that for every unit of visible volume, the market participant must anticipate competing against an additional $\bar{\rho}_t$ units of hidden volume. Under the assumption that order arrivals follow a locally stationary Poisson process with depletion intensity $\lambda_{depletion} = \lambda^M + \lambda^C$, we can derive the expected time-to-fill, $\mathbb{E}[\tau]$, for a newly submitted limit order. Standard queueing theory dictates that the wait time is the ratio of the queue length to the service rate. Substituting our adjusted metric yields:

$$\mathbb{E}[\tau_{fill}] = \frac{Q_{MDLT}(t,p)}{\lambda^M(t)+\lambda^C(t)} = \frac{Q_{visible}(t,p)(1+\bar{\rho}_t)}{\lambda_{depletion}(t)} \tag{2.20}$$

This equation highlights the critical deficiency of naive models: strategies relying solely on $Q_{visible}$ systematically underestimate execution latency by a factor of $(1+\bar{\rho}_t)$, leading to optimal execution schedules that are overly passive and prone to adverse selection. Q_{MDLT} corrects this bias, providing a mathematically consistent basis for execution logic.

Expected Wait Time Estimation

We formalize the execution latency, τ, as the first passage time of the cumulative depletion process reaching the order's effective queue position. Let $D(t)$ represent the cumulative volume removed from the queue via market orders and cancellations over the interval $[0,t]$. For a limit order positioned at queue depth Q, the execution time is the stochastic stopping time defined by:

$$\tau(Q) = \inf\{t > 0 : D(t) \geq Q\} \tag{2.21}$$

Under the assumption that the depletion process $D(t)$ follows a compound Poisson process with a constant aggregate intensity $\lambda_{depletion} = \lambda^M + \lambda^C$ and unit volume increments, the expectation of the stopping time is linear with respect to the queue depth. Standard queueing theory yields the first moment:

$$\mathbb{E}[\tau(Q)] = \frac{Q}{\lambda_{depletion}} \tag{2.22}$$

In a market regime characterized by hidden liquidity, utilizing the observable queue depth $Q_{visible}$ yields a *naive* wait time estimator, $\hat{\tau}_{naive}$. However, as derived in the previous section, the true barrier to execution is Q_{MDLT}. Consequently, the corrected MDLT wait time estimator, $\hat{\tau}_{MDLT}$, is given by:

$$\hat{\tau}_{MDLT} = \frac{Q_{MDLT}}{\lambda^M + \lambda^C} = \frac{Q_{visible}(1 + \bar{\rho})}{\lambda^M + \lambda^C} \tag{2.23}$$

The discrepancy between these two estimators represents the *Hidden Latency Bias*. We can express the relationship between the true and naive expectations as:

$$\hat{\tau}_{MDLT} = \hat{\tau}_{naive} \cdot (1 + \bar{\rho}) \tag{2.24}$$

This multiplicative relationship highlights the non-linear risk of ignoring iceberg orders. In regimes where $\bar{\rho} \approx 1$ (hidden volume equals visible), the naive model underestimates the time-to-fill by 50%. Such underestimation directly impacts optimal execution logic, particularly for Almgren-Chriss style trajectories, where the estimated variance of execution cost is a function of time. By substituting $\hat{\tau}_{MDLT}$ into the cost function, traders can accurately price the risk of "resting" in the queue versus paying the spread, thereby minimizing the implementation shortfall caused by unexpected delays.

2.2 Order Flow Imbalance and Regime Normalization

A critical challenge in latency tomography is decoupling the structural properties of the queue (depth) from the stochastic intensity of the arrival process (speed).

2.2.1 OFI-Corrected Latency Normalization

The raw execution latency $T = T_2 - T_1$ is inversely proportional to the queue depletion rate. Consequently, a decrease in T could ambiguously signal either a shallower queue or a surge in market aggressiveness. To resolve this ambiguity, we control for the Order Flow Imbalance, which acts as a good measure for short-term buying or selling pressure. We define the Order Flow Imbalance over an interval Δt as the net flow of liquidity demanding events:

$$OFI_t = \sum_{t-\Delta t<s\leq t} v_s^M \cdot \mathbb{I}_{\{dir_s=buy\}} - \sum_{t-\Delta t<s\leq t} v_s^M \cdot \mathbb{I}_{\{dir_s=sell\}} \tag{2.25}$$

High-magnitude OFI regimes are characterized by elevated arrival intensities $\lambda^M(t)$, which systematically bias raw latency measurements downward. To isolate the queue depth contribu-

tion, we introduce the Normalized Latency, τ_{norm}. This metric rescales the raw time-domain measurement into "volume-time" units, effectively normalizing for the varying speed of market depletion:

$$\tau_{norm} = (T_2 - T_1) \cdot \left(\hat{\lambda}^M + \hat{\lambda}^C\right) \tag{2.26}$$

By multiplying the time duration by the estimated depletion intensity, τ_{norm} approximates the total volume processed by the market during the probe interval. Unlike raw latency, this quantity is invariant to changes in trading tempo and provides a more stable basis for estimating the effective queue size Q_{MDLT} across different volatility regimes.

2.2.2 Regime-Dependent Density Estimation

Empirical evidence suggests that the presence of iceberg orders is not uniform but highly state-dependent. Institutional algorithms tend to vary their concealment logic based on market urgency and volatility. Therefore, a global average $\bar{\rho}$ may lack the specificity required for precision execution. To address this, we adopt a regime-switching framework conditioned on the OFI distribution. We partition the trading day into K distinct regimes based on the quintiles of the OFI distribution, denoted as $\mathcal{R}_k$ for $k \in \{1, \dots, 5\}$. We maintain separate exponentially weighted moving averages for the iceberg density coefficient within each regime. Let $\bar{\rho}^{(k)}$ represent the density estimator specific to the k-th OFI quintile. The update rule is applied conditionally:

$$\bar{\rho}_t^{(k)} = \begin{cases} \alpha\rho_t + (1-\alpha)\bar{\rho}_{t-1}^{(k)} & \text{if } OFI_t \in \mathcal{R}_k \\ \bar{\rho}_{t-1}^{(k)} & \text{otherwise} \end{cases} \tag{2.27}$$

The final Priority Metric is then constructed dynamically by selecting the density coefficient corresponding to the current market regime:

$$Q_{MDLT}(t) = Q_{visible}(t) \cdot \left(1 + \sum_{k=1}^{5} \mathbb{I}_{\{OFI_t \in \mathcal{R}_k\}} \bar{\rho}_{t-1}^{(k)}\right) \tag{2.28}$$

This stratified approach allows the MDLT algorithm to adapt to changing market microstructures, applying a higher "hidden liquidity penalty" in regimes known to feature heavy iceberg usage (e.g., low-volatility accumulation periods) while relaxing the penalty in high-velocity trends where liquidity is predominantly visible.

2.2.3 Real-Time Intensity Estimation

Algorithm 1 Estimate Order Flow Intensities

Input: Live market feed $\{(t_i, \text{type}_i, v_i, p_i)\}_{i=1}^{n}$, lookback window T_{win}
Output: Intensity vector $[\lambda_L, \lambda_C, \lambda_M]$
Initialize counters: $N_L \leftarrow 0, N_C \leftarrow 0, N_M \leftarrow 0$
for each event i in feed **do**
    if $t_{\text{now}} - t_i < T_{\text{win}}$ **then**
        if $\text{type}_i =$ "Limit" **then**
            $N_L \leftarrow N_L + 1$
        else if $\text{type}_i =$ "Cancel" **then**
            $N_C \leftarrow N_C + 1$
        else if $\text{type}_i =$ "Trade" **then**
            $N_M \leftarrow N_M + 1$
        end if
    end if
end for
$\lambda_L \leftarrow N_L / T_{\text{win}}$
$\lambda_C \leftarrow N_C / T_{\text{win}}$
$\lambda_M \leftarrow N_M / T_{\text{win}}$
Return $[\lambda_L, \lambda_C, \lambda_M]$

2.2.4 Probe Pair Execution with Tomographic Scan

Algorithm 2 MDLT Probe Pair Execution

Input: Best bid price p^*, gap δ (ms), quantity $q = 1$
Output: $(T_1, T_2, D_{\text{obs}}, V_{\text{gap}})$
Step 1: Observe $Q_{\text{visible}} \leftarrow$ current LOB depth at p^*
Step 2: Submit P_1: Limit Buy, Qty=1, Price=p^*
Step 3: Wait for fill, record $T_1 \leftarrow$ execution timestamp
Step 4: Sleep δ milliseconds
Step 5: Submit P_2: Limit Buy, Qty=1, Price=p^*
Step 6: Wait for fill, record $T_2 \leftarrow$ execution timestamp
Step 7: Scan market tape during $[T_1, T_2]$:
    $D_{\text{obs}} \leftarrow 0$
for each event e in $[T_1, T_2]$ **do**
    if $\text{type}(e) =$ "Trade" and $\text{price}(e) = p^*$ **then**
        $D_{\text{obs}} \leftarrow D_{\text{obs}} + \text{volume}(e)$
    else if $\text{type}(e) =$ "Cancel" and $\text{price}(e) = p^*$ **then**
        $D_{\text{obs}} \leftarrow D_{\text{obs}} + \text{volume}(e)$
    end if
end for
Step 8: Calculate visible adds:
    $V_{\text{gap}} \leftarrow$ sum of Limit orders at p^* during $(0, \delta)$
Step 9: Compute $\rho \leftarrow (D_{\text{obs}} / V_{\text{gap}}) - 1$
Step 10: Update rolling average:
    $\rho_{\text{smooth}} \leftarrow 0.9 \times \rho_{\text{smooth}} + 0.1 \times \rho$
Return $(T_1, T_2, D_{\text{obs}}, V_{\text{gap}})$

2.2.5 Priority Metric Calculation

Algorithm 3 Compute Q_MDLT

1: **Input:** Q_{visible}, ρ_{smooth}, $[\lambda_L, \lambda_C, \lambda_M]$
2: $Q_{\text{MDLT}} \leftarrow Q_{\text{visible}} \times (1 + \rho_{\text{smooth}})$
3: $\mu_{\text{depletion}} \leftarrow \lambda_M + \lambda_C$
4: $\mathbb{E}[\tau] \leftarrow Q_{\text{MDLT}}/\mu_{\text{depletion}}$
5: **Return** $(Q_{\text{MDLT}}, \mathbb{E}[\tau])$

2.3 Numerical Example

In this section, we ground the abstract principles developed thus far in a concrete market scenario. We present a detailed worked example showing how the tomographic measurement principle operates in practice, from the submission of probe orders through the calculation of hidden liquidity and the implications for execution strategy. This example is not merely illustrative; it demonstrates the mechanical operation of the MDLT framework and validates the claim that passive observation of the order book leaves critical information hidden.

2.3.1 Scenario Setup

We consider a liquid equity market at mid-morning trading hours, when volatility is moderate and order flow is predictable. The conditions are as follows:

Table 2.1: Market Conditions at Probe Submission Time

Parameter	**Value**
Security	Apple Inc. (AAPL)
Best Bid Price	\$100.00
Best Ask Price	\$100.01
Bid-Ask Spread	\$0.01 (1 cent)
Visible Queue Depth at Bid	500 shares
Market Time	10:30:00.000 (mid-morning)
Market Regime	Moderate volatility, normal activity

The visible queue of 500 shares represents limit buy orders placed at the best bid price of \$100.00. These are the orders that any market participant can observe through the public order book feed. However, as discussed in Section **??**, this visible depth likely understates the true queue depth because of iceberg orders. Our goal is to measure this hidden component through active probing.

2.3.2 Probe Sequence and Execution Timeline

We now trace the sequence of events as our two probe orders proceed through the matching engine. Each probe is a limit buy order of unit size (one share) submitted to the best bid price. The temporal spacing between submissions is critical: it defines the window over which we will observe queue dynamics.

Table 2.2: Probe Order Timeline: Submission and Execution

Time (HH:MM:SS.mmm)	**Event**	**Details**
10:30:00.000	Submit P_1	Limit Buy 1 share @ \$100.00
10:30:00.025	Submit P_2	Limit Buy 1 share @ \$100.00 ($\delta = 25$ ms)
10:30:00.058	P_1 executes	Execution time $T_1 = 58$ ms after submission
10:30:00.087	P_2 executes	Execution time $T_2 = 87$ ms after P_1 submission

The inter-probe gap is $\delta = 25$ milliseconds. This gap is chosen to be long enough to allow meaningful market activity (new limit orders, cancellations, market orders) to occur between submissions, but short enough that market regime (volatility, order flow intensity) remains approximately stationary. The execution times $T_1 = 58$ ms and $T_2 = 87$ ms reflect the time elapsed from the initial submission of P_1 until each probe fills.

The key observation is that P_1 and P_2 do not execute instantaneously. Each must wait for all orders ahead of it in the FIFO queue to be removed through either market order execution or cancellation. The wait time for P_1 is 58 milliseconds. By the time P_2 executes, an additional 29 milliseconds have passed. This additional waiting time encodes information about the queue state at the moment P_2 was submitted.

2.3.3 Inter-Execution Market Activity

Between the execution of P_1 (at 58 ms) and the execution of P_2 (at 87 ms), the order book is not quiescent. Market orders arrive and execute against standing limit orders. Some traders cancel their orders. The public market tape records all of these events. We now enumerate what occurred during this 29-millisecond interval.

Table 2.3: Market Tape Events in the Interval $[T_1, T_2]$ (Execution Interval)

Event Type at \$100.00 Bid	**Volume (shares)**	**Cumulative Volume**
Market Sell @ \$100.00	80	80
Market Sell @ \$100.00	120	200
Cancel (Limit Order) @ \$100.00	50	250
Market Sell @ \$100.00	90	340
Market Sell @ \$100.00	60	400
Total Volume Removed		**400**

The table above represents the complete market activity at the best bid price during the execution interval. A market sell is an aggressive order that executes immediately against the best standing bid, removing shares from the queue. A cancellation is a limit order withdrawal, also removing shares from the queue but not resulting in a transaction.

We aggregate across event types to obtain the total observed depletion:

$$D_{\text{obs}} = (\text{market orders executed}) + (\text{limit orders cancelled}) = 350 + 50 = 400 \text{ shares} \quad (2.29)$$

This quantity D_{obs} is fully observable from the market data feed. Every trade is time-stamped and reported. Every cancellation is announced to the market. Therefore, $D_{\text{obs}} = 400$ shares is a fact, not an estimate or inference.

2.3.4 Observable Queue Additions

While market activity removes volume from the queue during the interval $[T_1, T_2]$, other market participants are adding volume to the queue. Specifically, new limit orders arrive at the best bid price after P_1 is submitted but before P_2 executes. These arrivals are equally observable from the market data feed. We define the gap volume as the cumulative size of all limit orders that arrive at the best bid price during the inter-probe interval $[0, \delta]$, where time zero is the submission of P_1 and time $\delta = 25$ ms is the submission of P_2:

Table 2.4: Limit Order Arrivals During the Probe Gap $[0, \delta]$

Time (HH:MM:SS.mmm)	**Event: Limit Buy Arrivals at $100.00**
10:30:00.005	Arrival of 20 shares
10:30:00.018	Arrival of 30 shares
Total Gap Volume	**50 shares**

These arrivals represent new buy-side limit orders placed at the best bid price. They become part of the queue at the bid price, appearing in the public order book for all market participants to see. Thus, the gap volume $V_{\text{gap}} = 50$ shares is also fully observable.

2.3.5 Conservation Principle

We now invoke the conservation principle introduced in Section **??**. This principle states that the volume removed from the queue between the execution times of the two probes must equal the distance separating those probes in the queue. Formally, the distance between P_1 and P_2 in the execution queue is the sum of two components: the visible volume that arrived between their submission times, plus any hidden volume from iceberg orders:

$$\text{Distance between } P_1 \text{ and } P_2 = V_{\text{gap}} + H_{\text{gap}} \tag{2.30}$$

Here, V_{gap} is the observable gap volume (which we computed as 50 shares), and H_{gap} is the unobservable hidden volume from iceberg orders in the same interval.

Now, a fundamental fact about FIFO queue discipline: an order cannot execute until all orders ahead of it have been removed. When P_2 executes at time T_2, this means all volume separating P_1 from P_2 must have been depleted between the execution times T_1 and T_2. The volume depleted is precisely what we observe from the market tape: $D_{\text{obs}} = 400$ shares.

By conservation:

$$D_{\text{obs}} = V_{\text{gap}} + H_{\text{gap}} \tag{2.31}$$

Rearranging to solve for the hidden component:

$$H_{\text{gap}} = D_{\text{obs}} - V_{\text{gap}} = 400 - 50 = 350 \text{ shares} \tag{2.32}$$

This is the key result. Between the submission of our two probes, hidden iceberg orders concealed 350 shares of volume. This volume was never visible in the public order book, yet it constrained execution, added to the effective queue depth, and affected the execution dynamics of any trader trying to execute at the best bid.

2.3.6 Iceberg Density Estimation

We now normalize the hidden volume relative to the visible volume to create a dimensionless measure of iceberg intensity. The iceberg density coefficient ρ is defined as the ratio of hidden to visible volume:

$$\rho = \frac{H_{\text{gap}}}{V_{\text{gap}}} \tag{2.33}$$

Equivalently, substituting our expression for H_{gap}:

$$\rho = \frac{D_{\text{obs}}}{V_{\text{gap}}} - 1 \tag{2.34}$$

In our numerical example:

$$\rho = \frac{400}{50} - 1 \tag{2.35}$$

$$= 8 - 1 \tag{2.36}$$

$$= 7.0 \tag{2.37}$$

This result indicates that for every one share of visible liquidity in this interval, seven shares of hidden liquidity existed. Stated differently, the hidden volume is 700% of the visible volume, or equivalently, the true queue is eight times deeper than the visible queue suggests.

On Iceberg Density

An iceberg density of 7.0 is high, indicating unusually heavy use of hidden orders during this interval. In normal market conditions, typical values of ρ range from 0.2 to 0.6, indicating that hidden volume is 20% to 60% of visible volume. The elevated value in our scenario suggests one of several possibilities: (a) a large institutional investor is executing a significant block trade and has hidden most of their order; (b) market makers are using iceberg orders to manage inventory risks during a volatile period; or (c) the visible queue is unusually shallow due to earlier trading activity, making hidden orders appear more prominent.

The interpretation is straightforward: $\rho = 0$ would mean the order book is fully transparent, with no hidden liquidity. $\rho > 0$ indicates the presence of iceberg orders. Higher values of ρ indicate heavier reliance on concealment strategies.

2.3.7 Adjusted Queue Position

Having measured the iceberg density from our probe pair, we can now apply this information to refine our understanding of the queue depth at subsequent times. Suppose that at time $t = 87$ ms (the moment when P_2 executes), a trader wishes to submit a new large limit order at the same price level. The trader observes from the public order book that the visible queue depth is $Q_{\text{visible}} = 500$ shares.

If the trader naively assumes that this visible depth is the true queue depth, they will make execution decisions under the assumption that the queue is shallow. However, our probe measurement has just revealed that during the recent interval, the iceberg density was $\rho = 7.0$. Assuming this density persists (an assumption we will refine in Chapter Two through smoothing), the true effective queue depth is:

$$\begin{aligned} Q_{\text{MDLT}} &= Q_{\text{visible}} \times (1 + \rho) && (2.38)\\ &= 500 \times (1 + 7.0) && (2.39)\\ &= 500 \times 8 && (2.40)\\ &= 4000 \text{ shares} && (2.41) \end{aligned}$$

The MDLT metric adjusts the visible queue by the factor $(1 + \rho)$ to account for hidden liquidity. In this case, the adjustment is substantial: a visible queue of 500 shares becomes an effective queue of 4000 shares. This adjustment captures the intuition that hidden icebergs act as additional layers of queueing depth, even though they are not visible.

2.3.8 Wait Time Estimation

With an adjusted queue position in hand, we can now estimate expected execution times using queueing theory. Recall that under the M/M/1 queue model, the expected wait time for an order at queue position Q is

$$\mathbb{E}[\tau] = \frac{Q}{\lambda_M + \lambda_C} \tag{2.42}$$

where λ_M is the rate of market order arrivals and λ_C is the rate of cancellations (both measured in shares per second). For our scenario, we estimate from recent market data that the combined depletion rate is $\lambda_M + \lambda_C = 50$ shares per second.

Naive Estimate

A trader who observes only the visible queue would estimate:

$$\mathbb{E}[\tau_{\text{naive}}] = \frac{Q_{\text{visible}}}{\lambda_M + \lambda_C} \tag{2.43}$$
$$= \frac{500}{50} \tag{2.44}$$
$$= 10 \text{ seconds} \tag{2.45}$$

This estimate suggests that the queue will clear in 10 seconds, a reasonable wait time. Based on this estimate, the trader might decide that joining the queue at the best bid is preferable to paying the spread through a market order.

Multidimensional Latency Estimates

Our measurement, however, reveals a different picture:

$$\mathbb{E}[\tau_{\text{MDLT}}] = \frac{Q_{\text{MDLT}}}{\lambda_M + \lambda_C} \tag{2.46}$$
$$= \frac{4000}{50} \tag{2.47}$$
$$= 80 \text{ seconds} \tag{2.48}$$
$$\approx 1.3 \text{ minutes} \tag{2.49}$$

The MDLT estimate suggests that the order will wait approximately 80 seconds—a much longer duration. This dramatic difference arises entirely from the hidden liquidity revealed by our probes.

Decision Rule

The trader now faces a different calculus. An 80-second wait exposes the position to significant price risk. If the market price moves by even a few cents against the position during that wait, the cost of the move will exceed the spread savings from joining the limit order queue. The decision rule might be structured as follows:

First, we classify order sizes into categories based on the expected wait time and associated risks:

(a) **Small Orders** ($N < 100$ shares): Even with an 80-second wait, the order is small enough that it likely clears quickly from the queue. The decision is to join the queue at the best bid. Expected wait time is less than 1-2 seconds even after MDLT adjustment.

(b) **Medium Orders** ($100 \leq N \leq 1000$ shares): The wait time becomes material. The trader should consider a time-weighted average price (TWAP) algorithm that spreads the execution across a longer time horizon (e.g., 10-15 minutes), reducing the impact of any single segment of the order joining the queue at a given moment.

(c) **Large Orders** ($N > 1000$ shares): The wait time in a queue with depth equivalent to 4000 shares is prohibitive. The trader is better served by using market orders (paying the

spread immediately) or seeking out hidden liquidity pools and alternative trading venues where the queue structure may be different.

You Can't Ignore Me

To quantify the cost of ignoring the hidden liquidity, consider a specific scenario. Suppose the trader places a 1000-share order at the best bid, intending to wait for execution. Under the naive model, the trader expects execution in 20 seconds (1000/50). However, the MDLT model reveals the true wait time is 160 seconds. During that additional 140-second wait, the market price might move. If the midpoint price rises by just 0.05 (five cents), the trader loses $1000 \times 0.05 = \$50$ due to the price move, an amount that vastly exceeds the \$0.01 spread savings from using a limit order. Conversely, if the trader had used the MDLT measurement to inform the execution strategy, they might have chosen to (a) submit smaller segments of the order across multiple price levels, (b) access hidden liquidity through alternative venues, or (c) use market orders to ensure immediate execution at a known price. Each of these alternatives protects against the risk of unexpected price movement during the wait.

2.4 Risk Analysis

Every active measurement carries an economic cost. Unlike passive inference, which requires only data observation, active probing requires sending orders into the market. These orders must execute to generate the signal we need, and execution incurs trading costs. Understanding and managing these costs is critical to ensuring that the value of measurement exceeds its price.The MDLT framework provides a principled approach to measuring hidden queue depth, yet like all measurement systems operating in complex environments, it is subject to costs, model assumptions, and failure modes. This section systematically examines these constraints and proposes mitigation strategies. Understanding these limitations is essential: a robust measurement system is one that explicitly acknowledges where it may fail and implements safeguards accordingly.

Spread Cost Per Probe Pair

The direct cost of submitting a probe pair arises from the bid-ask spread. When we submit a limit buy order at the best bid price, it executes at that bid price. We thus "pay" the full bid-ask spread in the sense that we sell to the market at the bid price, which is lower than the contemporaneous ask price. For a probe order of unit size (one share), the cost is the spread itself:

$$C_{\text{probe pair}} = 2 \times \frac{s}{2} = s \tag{2.50}$$

where s denotes the bid-ask spread. Each of our two probes costs half the spread (since we execute at the bid and the ask midpoint is halfway between bid and ask). Summing both probes yields a total cost equal to the full spread. For highly liquid securities such as AAPL, which typically trade with spreads of one penny, the cost per probe pair is:

$$C_{\text{probe pair}} = \$0.01 \tag{2.51}$$

This cost is minimal in absolute terms. However, for the measurement to create positive economic value, the information gain must justify this cost. We therefore define a break-even condition.

Break-Even Analysis

The information extracted from a probe pair is valuable only if it prevents greater losses in the subsequent main order execution. Let Δ_{slippage} denote the per-share reduction in slippage (measured in dollars per share) that results from using MDLT-informed execution versus naive execution. For a main order of size N shares, the total benefit from improved execution is:

$$\text{Benefit} = N \times \Delta_{\text{slippage}} \tag{2.52}$$

For the measurement to be economical, the benefit must exceed the cost:

$$N \times \Delta_{\text{slippage}} > C_{\text{probe pair}} \tag{2.53}$$

Rearranging to solve for the break-even order size:

$$N_{\text{break-even}} = \frac{C_{\text{probe pair}}}{\Delta_{\text{slippage}}} = \frac{s}{\Delta_{\text{slippage}}} \tag{2.54}$$

To make this concrete, consider a realistic scenario. Suppose accurate queue depth measurement prevents one basis point (0.01%) of slippage per share. For AAPL trading at approximately \$150 per share, one basis point is $150 \times 0.0001 = \$0.015$ per share. With a probe cost of \$0.01 per pair:

$$N_{\text{break-even}} = \frac{0.01}{0.015} \approx 667 \text{ shares} \tag{2.55}$$

Alternatively, if we estimate more conservatively that MDLT prevents 1 basis point of slippage in dollar terms (not percentage terms), then:

$$N_{\text{break-even}} = \frac{0.01}{0.0001} = 100 \text{ shares} \tag{2.56}$$

In practice, institutional investors executing orders of 100 to 10,000 shares are common in equity markets. The break-even threshold of 100–700 shares is well within the range of institutional order sizes. For smaller retail orders (fewer than 100 shares), the measurement cost exceeds the likely benefit. For institutional orders, the measurement is economical.

Probe Non-Execution

A subtle but important risk arises if probe orders fail to execute promptly. Our methodology assumes that both P_1 and P_2 execute within a short time window (typically tens to hundreds of milliseconds). If the market price moves away from the best bid during the measurement interval, our limit buy orders will sit unfilled in the queue without contributing to the measurement

signal. Specifically, if the security's price rises above our limit bid price (e.g., if the best bid moves from $100.00 to $100.01), our limit orders become "out of the money" and will not execute until the price falls back. This creates two problems. First, we have submitted orders but received no signal; the measurement is incomplete. Second, if the price does later drop back, our old orders may execute far later than intended, at a time when market conditions have changed and the measurement signal has become stale.nTo mitigate this risk, we recommend using immediate-or-cancel (IOC) orders for probes rather than persistent limit orders. An IOC probe is a limit order that executes any portion that matches immediately, and any remainder is automatically cancelled. We would typically set a timeout window (e.g., 200 milliseconds) within which the probe must execute. If it does not execute within that window, it is cancelled, and we attempt a fresh probe in the next measurement cycle. The tradeoff is that IOC probes may not execute at all if market conditions are adverse (e.g., large spreads, shallow depth). In that case, we obtain no measurement signal. However, a non-signal in an adverse market regime is arguably more informative than a delayed signal that reflects stale conditions. We recommend monitoring the probe execution rate: if the fraction of probe pairs that execute drops below 80%, this indicates either a regime change (wider spreads, lower liquidity) or technical issues with order submission, both of which warrant immediate recalibration.

Bibliography

[1] Almgren, R., & Chriss, N. (2001). Optimal execution of portfolio transactions. *Journal of Risk*, 3, 5–40.

[2] Bacry, E., Mastromatteo, I., & Muzy, J.-F. (2015). Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01), 1550005.

[3] Christensen, C., & Woodmansey, P. (2013). Detecting iceberg orders. *Market Microstructure Knowledge Base*.

[4] Cont, R., Stoikov, S., & Talreja, R. (2010). A stochastic model for order book dynamics. *Operations Research*, 58(3), 549–563.

[5] Cont, R., Kukanov, A., & Stoikov, S. (2014). The price impact of order book events. *Journal of Financial Econometrics*, 12(1), 47–88.

[6] Frey, S., & Sandås, P. (2017). The impact of iceberg orders in limit order books. *Review of Finance*, 21(2), 773–800.

[7] Glosten, L. R. (1994). Is the electronic open limit order book inevitable? *Journal of Finance*, 49(4), 1127–1161.

[8] Kyle, A. S. (1985). Continuous auctions and insider trading. *Econometrica*, 53(6), 1315–1335.

[9] Moallemi, C. C., & Yuan, K. (2016). A model for queue position valuation in a limit order book. *Working Paper*, Columbia University.

[10] Moro, E., Vicente, J., Moyano, L. G., Gerig, A., Farmer, J. D., Vaglica, G., Lillo, F., & Mantegna, R. N. (2009). Market impact and trading profile of hidden orders in stock markets. *Physical Review E*, 80(6), 066102.

[11] Rosu, I. (2009). A dynamic model of the limit order book. *Review of Financial Studies*, 22(11), 4601–4641.

[12] Zotikov, D., & Antonov, A. (2019). CME iceberg order detection and prediction. *arXiv preprint arXiv:1909.09495*.

2.5 Code Template

2.5.1 Intensity Estimation

```
import numpy as np
import pandas as pd

class IntensityEstimator:
    def __init__(self, lookback_window=60):
        """
        lookback_window: seconds of history for rate estimation
        """
        self.lookback_window = lookback_window
        self.event_buffer = []

    def update(self, event_type, timestamp):
        """
        event_type: 'limit', 'cancel', 'trade'
        timestamp: float (seconds since epoch)
        """
        self.event_buffer.append((event_type, timestamp))
        # Prune old events
        cutoff = timestamp - self.lookback_window
        self.event_buffer = [(t, ts) for (t, ts) in self.event_buffer
                             if ts >= cutoff]

    def estimate_intensities(self):
        """
        Returns: (lambda_L, lambda_C, lambda_M) in events/second
        """
        if len(self.event_buffer) == 0:
            return (0, 0, 0)

        counts = {'limit': 0, 'cancel': 0, 'trade': 0}
        for event_type, _ in self.event_buffer:
            counts[event_type] = counts.get(event_type, 0) + 1

        lambda_L = counts['limit'] / self.lookback_window
        lambda_C = counts['cancel'] / self.lookback_window
        lambda_M = counts['trade'] / self.lookback_window

        return (lambda_L, lambda_C, lambda_M)
```

2.5.2 Probe Pair Submission

```
import time
from ib_insync import IB, LimitOrder

class MDLTProbeExecutor:
    def __init__(self, ib_connection, gap_ms=50):
        self.ib = ib_connection
        self.gap_ms = gap_ms
        self.rho_history = []

    def submit_probe_pair(self, symbol, bid_price):
        """
        Returns: (T1, T2, D_obs, V_gap, rho)
        """
        # Step 2: Submit P1
        contract = Stock(symbol, 'SMART', 'USD')
        order_p1 = LimitOrder('BUY', 1, bid_price)
        trade_p1 = self.ib.placeOrder(contract, order_p1)

        # Step 3: Wait for fill, record T1
        while not trade_p1.isDone():
            self.ib.sleep(0.001)
        T1 = trade_p1.log[-1].time.timestamp()

        # Step 4: Sleep delta
        time.sleep(self.gap_ms / 1000.0)

        # Step 5: Submit P2
        order_p2 = LimitOrder('BUY', 1, bid_price)
        trade_p2 = self.ib.placeOrder(contract, order_p2)

        # Step 6: Record T2
        while not trade_p2.isDone():
            self.ib.sleep(0.001)
        T2 = trade_p2.log[-1].time.timestamp()

        # Step 7-8: Scan market tape (requires market data subscription)
        D_obs = self.scan_market_activity(symbol, T1, T2, bid_price)
        V_gap = self.calculate_gap_volume(symbol, 0, self.gap_ms / 1000, bid_price)

        # Step 9: Compute rho
        if V_gap > 0:
```

```
            rho = (D_obs / V_gap) - 1
        else:
            rho = 0

        # Step 10: Update rolling average
        self.rho_history.append(rho)
        if len(self.rho_history) > 10:
            self.rho_history.pop(0)

        return (T1, T2, D_obs, V_gap, rho)

    def scan_market_activity(self, symbol, T1, T2, price):
        """
        Query tick data for [T1, T2], sum trades and cancels at price
        Returns: D_obs (int)
        """
        # Placeholder: requires market data subscription
        # In practice: query historical ticks from IB or LOBSTER
        return 100  # dummy value

    def calculate_gap_volume(self, symbol, t_start, t_end, price):
        """
        Sum visible limit order adds during [t_start, t_end] at price
        Returns: V_gap (int)
        """
        # Placeholder
        return 50
```

2.5.3 Queue Position Prediction

```
class QueuePredictor:
    def __init__(self, alpha=0.1):
        self.alpha = alpha
        self.rho_smooth = 0

    def update_rho(self, rho_new):
        """
        Exponential smoothing
        """
        self.rho_smooth = (1 - self.alpha) * self.rho_smooth + self.alpha * rho_new

    def predict_queue_position(self, Q_visible, lambda_M, lambda_C):
        """
```

```
    Returns: (Q_MDLT, expected_wait_time)
    """
    Q_MDLT = Q_visible * (1 + self.rho_smooth)
    mu_depletion = lambda_M + lambda_C
    if mu_depletion > 0:
        wait_time = Q_MDLT / mu_depletion
    else:
        wait_time = float('inf')
    return (Q_MDLT, wait_time)
```

2.6 Proofs

This section presents the various proofs of the different concepts mentioned in this book.

2.6.1 Conservation Law

Queue at price p^* evolves as:

$$Q(t) = Q(0) + \int_0^t dN_L(s) - \int_0^t dN_M(s) - \int_0^t dN_C(s)$$

For probe P_1 at position q_1, execution at T_1 implies:

$$\int_0^{T_1} (dN_M(s) + dN_C(s)) = q_1$$

For P_2 at position $q_2 = q_1 + V_{\text{gap}} + H_{\text{gap}}$:

$$\int_0^{T_2} (dN_M(s) + dN_C(s)) = q_2 = q_1 + V_{\text{gap}} + H_{\text{gap}}$$

Subtracting:

$$\int_{T_1}^{T_2} (dN_M(s) + dN_C(s)) = V_{\text{gap}} + H_{\text{gap}}$$

But LHS is observable:

$$D_{\text{obs}} = \int_{T_1}^{T_2} dN_M(s) + \int_{T_1}^{T_2} dN_C(s)$$

Hence:

$$H_{\text{gap}} = D_{\text{obs}} - V_{\text{gap}}$$

2.6.2 Unbiasedness Under Poisson Assumptions

Theorem: If $N_M(t), N_C(t)$ are Poisson with constant rates λ_M, λ_C, and icebergs refill uniformly in time, then $\mathbb{E}[\hat{H}_{\text{gap}}] = H_{\text{gap}}$.

Proof:

$$\mathbb{E}[D_{\text{obs}}] = \mathbb{E}\left[\int_{T_1}^{T_2} dN_M(s) + dN_C(s)\right] = (\lambda_M + \lambda_C)\mathbb{E}[T_2 - T_1]$$

By definition, $T_2 - T_1$ is the time to deplete q_2:

$$\mathbb{E}[T_2 - T_1] = \frac{q_2}{\lambda_M + \lambda_C} = \frac{V_{\text{gap}} + H_{\text{gap}}}{\lambda_M + \lambda_C}$$

Substituting:

$$\mathbb{E}[D_{\text{obs}}] = (\lambda_M + \lambda_C) \cdot \frac{V_{\text{gap}} + H_{\text{gap}}}{\lambda_M + \lambda_C} = V_{\text{gap}} + H_{\text{gap}}$$

Hence:

$$\mathbb{E}[\hat{H}_{\text{gap}}] = \mathbb{E}[D_{\text{obs}} - V_{\text{gap}}] = (V_{\text{gap}} + H_{\text{gap}}) - V_{\text{gap}} = H_{\text{gap}}$$