

# Exchange and Algorithmic Trading — Part 1:

## Understand the Market

---

This document summarizes practical points for building trading algorithms and simple smart order routing for US equities (NASDAQ & NYSE).

---

## Part 1 — Market basics (US equities: NASDAQ & NYSE)

### Market hours (ET)

- Regular trading hours (RTH): 09:30 — 16:00.
- Extended hours (pre/post): typically 04:00 — 09:30 (pre) and 16:00 — 20:00 (post). Rules and liquidity vary by venue.

### Opening & closing auctions

- Opening auction: single-price match at 09:30 using orders submitted during the pre-open. Exchanges publish imbalance indicators before open.
- Closing auction: single-price match at 16:00. Indicative prices and imbalance info are provided ahead of the close.
- Allocation and tie-breaks vary by exchange (often price-time priority). Check NASDAQ/NYSE rulebooks for exact details.

### Extended-hours trading

- Fewer order types accepted (often limit-only). Liquidity is thinner and spreads are wider.
- Price discovery is more fragile; auctions generally don't run the same way as during RTH.

### Common order types & how matching works

- Common order types: Market, Limit, Stop, Stop-limit, IOC, FOK, GTC/GTD, MOO/MOC/LOO/LOC, pegged (midpoint/primary), iceberg/reserve, hidden, post-only, sweep-to-fill.
  - Matching: continuous book uses price priority, then time priority. Auction crosses match at a single price using venue-specific allocation rules.
- 

## Part 2 — Alternative venues & smart routing

### Dark pools — why use them?

- Dark pools offer anonymity and can reduce visible market impact for large orders. They may provide price improvement but come with tradeoffs.

### How dark pools work

- Alternative Trading Systems (ATS) match non-displayed interest using methods like midpoint crosses, reference-price matches, or periodic crosses. Matching logic and allocation rules differ across venues.

## Risks with dark pools

- Adverse selection, toxic flow, and regulatory scrutiny. Price discovery on the lit market can be weaker if too much liquidity is hidden.

## Smart Order Routing (SOR)

- SOR addresses fragmented liquidity by routing child orders across venues to optimize execution based on price, size, latency, and fees.
  - A typical SOR:
    - Ingests live top-of-book and venue state.
    - Estimates fill probability and expected cost.
    - Splits parent orders into child orders and sends them to selected venues.
    - Monitors fills and re-routes partial fills or failures.
  - SOR can be simple rule-based or enhanced with predictive models for fill probability and adverse-selection avoidance.
- 

## Part 3 — Agency algos (practical guidance)

General points for all algos:

- Be careful around open/close auctions — pause or adapt behavior.
- Have policies for leftover quantity and low liquidity (e.g., escalate aggression, route to dark pools, or delay).
- Monitor fills and performance continuously.

## Short descriptions of common algos

### 1. Whisper

- Small, randomized limit orders near the BBO or midpoint to probe for hidden liquidity with minimal signaling.

### 2. TWAP (Time-Weighted Average Price)

- Splits the parent order evenly over a time window.

### 3. VWAP (Volume-Weighted Average Price)

- Allocates execution proportional to an intraday volume profile to match the VWAP benchmark.

### 4. Decipher (predictive)

- Adjusts aggression, price, and venue using short-term signals (order book slope, imbalance, trade flow) from a trained model.

### 5. Iceberg

- Posts a visible slice and hides the rest; replenishes the visible slice as it executes.
-

## Concise pseudocode (practical, with edge-case handling)

### Whisper

```
parent = Q
slice_size = small
while parent > 0 and time_left:
    sleep(random_short())
    price = pick_near(BBO_or_mid, offset)
    id = send_limit(price, slice_size)
    wait(short_timeout)
    if filled(id): parent -= filled
    else: cancel(id)
    if low_fill_rate: escalate_or_route_to_dark()
    if auction_imminent: pause_or_prepare_auction_orders()
```

### TWAP

```
parent = Q
intervals = N
for i in 0..N-1:
    wait_until(scheduled_time(i))
    qty = base_slice_or_adjusted()
    send_limit_passive(qty)
    wait(fill_timeout)
    handle_partial_fill(remaining => retry_or_escalate_per_policy)
    if auction_imminent: pause_or_submit_MOC_if_configured()
```

### VWAP

```
parent = Q
profile = load_intraday_volume_profile()
for interval in profile_between(start, end):
    expected_vol = live_or_profile(interval)
    qty = round(Q * expected_vol / total_expected_vol)
    decide_aggression_based_on(liquidity, time_left)
    send_order(qty, type_based_on_aggression)
    handle_partials(route_to_other_venues_or_increase_aggression)
```

### Decipher (predictive)

```
parent = Q
model = trained_short_term_model()
while parent > 0 and time_left:
    state = sample_orderbook()
    pred = model.predict(state)
```

```

action = policy_from(pred) # price, size, venue, order_type
send_order(action)
monitor_and_update_model_with_outcome()
if adverse_signal_or_auction: pause_or_adjust()

```

## Iceberg

```

parent = Q
display = D
reserve = Q - D
while parent > 0:
    post_limit(price, display, reserve_visible=False)
    wait_until(display_filled_or_timeout)
    if display_filled:
        parent -= display
        display = min(D, reserve); reserve -= display
    else:
        cancel_and_adjust_or_escalate()
        handle_auction_periods_as_configured()

```

Edge-case policies for all algos:

- Order incompleteness: report remainders; convert to market or escalate based on policy at deadline.
- Inadequate liquidity: increase aggression within risk limits, route to more venues, or delay execution per policy.
- Auctions: pause continuous execution near auction windows unless specifically targeting the auction.

---

## Part 3 (continued) — Broker differences & useful metrics

How broker implementations may differ

- Parameter tuning (slice sizes, participation rates)
- Aggression escalation rules and thresholds
- Venue access and routing preferences
- Fee/rebate models and latency considerations
- Use of predictive models vs deterministic rules
- Auction handling and reporting granularity

Key metrics to track

Execution quality:

- Implementation Shortfall (arrival price slippage)
- VWAP/TWAP benchmark deviation
- Price improvement vs NBBO midpoint

Fill and latency metrics:

- Fill rate, time-to-completion, partial-fill frequency
- Venue-level fill rates and latency

Impact & adverse selection:

- Immediate market impact and short-term adverse moves after fills

Operational metrics:

- Child messages and cancel rates (message efficiency)
- Re-routes and failure counts

Risk & consistency:

- Variance of slippage, tail-risk percentiles (95/99)

Track these as daily KPIs per strategy and broker. Use percentiles and time series to spot trends and regressions.

---

## References & next steps

- Review NASDAQ and NYSE rulebooks and FIX specs for exact details on orders and flags.
- Backtest algos on historical L1/L2 data, including auctions and low-liquidity periods.
- Build monitoring dashboards for the metrics above and automate alerts for regressions.