Algorithmic Trading for Reversion and Trend-Following

Author: *Kecheng Shi*

Abstract

We develop and evaluate a fully specified, research-to-execution stack for long-only trading in BTC/USD on a fixed 5-minute, right-closed/right-labeled UTC grid. Two causal signal modules are implemented: an EMA–ADX trend follower with an optional price-action gate and a mid-price z-score reversion model gated by ADX/RSI. Research and pseudo-live paths share identical state machines and an explicit friction model—fees f, symmetric slippage ε, and exchange quantization $(\delta p, \delta q)$—with self-financing wealth $W_t = W_{t-1} + Q_{t-1}(C_t - C_{t-1})$ and discrete jumps for slippage and fees. Live routing via Alpaca Crypto uses conservative, single-quote GTC limit placement at offset η with two-threshold hysteresis for replace/cancel, affordability haircuts on buys, and inventory-aware sells; risk is enforced out-of-process using a one-day parametric $VaR_\alpha \approx z_\alpha \hat{\sigma}_{min} \sqrt{1440} |Q_t| P_t$.

On the UTC window ¿ the trend leg attains net total return 51.91%, Sharpe $\widehat{SR} = 2.13$, and $MDD = -13.59\%$, while the reversion leg achieves 37.29%, $\widehat{SR} = 2.65$, and $MDD = -7.55\%$; a buy-and-hold reference records $\widehat{SR} \approx 1.65$ with $MDD \approx -31.6\%$. Stress programs perturb exactly the deployed knobs (limit offset η, hysteresis, freshness gates, and risk limits) and report acceptance over operational envelopes. The system is containerized (Docker) with a PostgreSQL/TimescaleDB backend, idempotent order flow, and a data-quality guard, yielding a reproducible, auditable pipeline. Results are in-sample and instrument-specific; we outline a roadmap for walk-forward validation, queue-reactive placement, EVT-based tail risk, and multi-asset, cross-venue generalization.

1. Introduction

This paper presents a fully specified, research-to-execution stack for systematic trading in BTC/USD on a fixed 5-minute grid. The stack couples two simple, complementary signal modules—a long-only EMA–ADX trend follower with an optional price-action gate, and a long-only mid-price z-score reversion model gated by ADX/RSI—to an execution layer that is conservative by design. The backtester and the pseudo-live executor share the same causal state machines, identical bar policy (right-closed/right-labeled UTC bars), and an explicit friction model (fees, slippage, and exchange quantization), so that research and run-time paths remain path-consistent by construction. Accounting follows the usual self-financing wealth recursion $W_t = W_{t-1} + Q_{t-1}(C_t - C_{t-1})$ with discrete jumps for slippage and fees, while position updates occur only on the algebraic crossings that define each module's state.

The implementation is environment-driven and auditable. Code is organized into a batch backtester (`part1_backtest.py`), a pseudo-live trader that routes via Alpaca Crypto (`part2_pseudo_live.py`), a real-time data-guard for bar integrity (`part3_data_guard.py`), an order-update listener (`part3_order_listener.py`), and an out-of-process risk worker (`part3_risk_worker.py`), with a Docker composition and a PostgreSQL/TimescaleDB store (`docker-compose.yml`, `000_init.sql`). Order flow is idempotent (stable client IDs), order hygiene is hysteretic (replace only on sufficient drift and age; cancel on flips or max lifetime), and sizing is equity-proportional under explicit caps, affordability checks, and venue ticks/lots. All timestamps are in UTC; configuration, limits, and defaults are surfaced as environment variables and logged, yielding runs that are reproducible from $(\text{code}, \text{config}, \text{data}, \text{seed})$.

Empirically, on a single contiguous UTC window at 5-minute resolution, the trend follower captured more of the directional move but with higher drawdown, while the reversion leg produced a higher risk-adjusted profile with lower drawdown. For the representative run reported later, the trend module attained a net total return of 51.91% with $\widehat{SR}=2.13$ and $MDD=-13.59\%$, whereas the reversion module delivered 37.29% with $\widehat{SR}=2.65$ and $MDD=-7.55\%$. A buy-and-hold reference over the same window posted $\widehat{SR}\approx 1.65$ and $MDD\approx -31.6\%$. These figures are net of the backtest frictions used throughout and reflect timing skill rather than leverage or omitted costs.

The study is intentionally narrow in scope—one symbol, one bar size, one venue route, and fixed hyperparameters chosen ex ante—and its statistics are descriptive rather than inferential. Limitations include stylized research fills versus live limit placement, Gaussian parametric VaR in the risk layer, and residual data risk despite integrity checks. To quantify robustness where it matters operationally, we couple results to stress programs that perturb exactly the deployed knobs: limit offsets, repricing hysteresis, freshness gates, affordability and inventory constraints, and risk limits. The roadmap, detailed later, prioritizes purged/embargoed walk-forward with serial-correlation–adjusted and deflated Sharpe reporting, queue-reactive placement and state-dependent hysteresis, EVT-based tail risk, deterministic replay for reconciliation tests, and multi-asset, cross-venue generalization under explicit portfolio budgets.

2. Data & Preprocessing

2.1 Universe and Horizon

The empirical universe is the BTC–USD cross (BTC/USD). All analyses use a single, fixed sampling interval of 5 minutes; no other bar sizes are employed. Let P_t denote the bar-close price and define continuously compounded returns by $r_t = \ln P_t - \ln P_{t-1}$. The study evaluates a single contiguous backtest window $[T_0, T_1)$ without train/validation/test splits; all statistics are therefore descriptive of the in-sample period. Where running estimates are required (e.g., EMAs, rolling moments), they are computed causally using only information available up to time t to avoid look-ahead in the state evolution, while acknowledging that the absence of a hold-out window precludes out-of-sample validation.

2.2 Vendors and Access

Market connectivity and order routing use a broker REST interface exposed through an official SDK for authenticated access to accounts, orders, and market data. All timestamps are localized to UTC, which serves as the reference clock for logging, aggregation, and evaluation. When daily aggregates are reported, the day boundary is taken at $00:00\ UTC$ to eliminate daylight-saving artifacts and ensure calendar consistency.

2.3 Cleaning and Resampling

Raw trades and quotes are normalized to a unified schema keyed by exchange timestamp and then aggregated to the native 5-minute grid. OHLC feasibility is enforced for every bar via $H_t \geq max(O_t, C_t)$ and $L_t \leq min(O_t, C_t)$, with non-negative volume and spreads. Bars are right-closed and right-labeled to prevent double counting across adjacent intervals. Because only one bar size is used, no downsampling is required; when coarser summaries are displayed, they are derived by composition-preserving aggregation,

$$O_t^{(k)} = O_{t-k+1}^{(1)}, C_t^{(k)} = C_t^{(1)}, H_t^{(k)} = \max_{i=0}^{k-1} H_{t-i}^{(1)}, L_t^{(k)} = \min_{i=0}^{k-1} L_{t-i}^{(1)},$$

which preserves price-path identities. Short gaps of length at most g_{max} bars are imputed conservatively using last-observation-carried-forward with $O=H=L=C$ and zero volume; longer gaps are flagged and excluded from estimation. Outliers are mitigated by robust filters on r_t and on spreads, with optional winsorization at symmetric quantiles.

2.4 Transaction Model

Backtests internalize explicit fees, spread, and slippage. Let q_t be the signed base-asset quantity, spread_t the quoted spread, and f a proportional fee rate. Executions are simulated at

$$P_t^{exec} = P_t^{mid} + 1/2\, sign(q_t)\, \text{spread}_t + \lambda\, sign(q_t) |q_t|^{\alpha},$$

with cash cost $C_t = f |q_t| P_t^{exec}$. Orders are quantized to exchange constraints by a price tick δp and a lot step δq. A latency proxy Δt_{lat} (network plus queueing) shifts fills to reference market data from $t + \Delta t_{lat}$. Impact parameters (λ, α) and frictions (f, spread_t) are held fixed within a run to preserve path consistency.

2.5 Quality Checks

Automated validations run at ingest and after each aggregation step. The UTC time index must be strictly monotone with the expected bar count of 288 per calendar day for 5-minute sampling in $24/7$ crypto trading. OHLC identities and non-negativity constraints are reasserted, and derived metrics satisfy $\hat{\sigma}_t^2 \geq 0$. Cross-instrument sanity checks (e.g., futures–spot basis where applicable) guard against economically infeasible prints. Structural changes are annotated via change-point scans on r_t and on spread_t. Storage-level constraints enforce primary-key uniqueness by $(\text{symbol}, \text{timestamp}_{UTC}, \text{bar size})$, preventing duplicates and overlaps. Within these controls, all reported results attribute performance to reproducible inputs on a single, UTC-clocked 5-minute bar stream.

3. Mathematical Models

This section defines the signals, gates, and state transitions used by the two strategies. Time is a fixed, right-closed grid indexed by integers $t \in Z$ (e.g., 5-minute bars). Each bar has OHLC tuple (O_t, H_t, L_t, C_t) and we write $P_t \equiv C_t$. The indicator $1[\cdot]$ takes value 1 when its predicate is true and 0 otherwise. All recursions are causal, using only information available at time t; a warm-up of length equal to the largest lookback is discarded before any statistic is read.

3.1 Trend-Following

The core trend estimate is an EMA crossover filtered by Wilder's trend-strength. Fast and slow exponential moving averages of the close are

$$E_t^{(f)} = \alpha(\tau_f) C_t + (1 - \alpha(\tau_f)) E_{t-1}^{(f)}, E_t^{(s)} = \alpha(\tau_s) C_t + (1 - \alpha(\tau_s)) E_{t-1}^{(s)},$$

with smoothing factor $\alpha(\tau) = 2/\tau + 1$ and initialization $E_{t_0}^{(\cdot)} = C_{t_0}$ at the end of warm-up. Wilder's framework supplies a bounded strength proxy. The true range is

$$TR_t = max\{H_t - L_t, |H_t - C_{t-1}|, |L_t - C_{t-1}|\},$$

and directional movements are

$$+DM_t = max\{H_t - H_{t-1}, 0\} 1[H_t - H_{t-1} > L_{t-1} - L_t], -DM_t = max\{L_{t-1} - L_t, 0\} 1[L_{t-1} - L_t > H_t - H_{t-1}].$$

Wilder's smoothed (exponential) moving average with length n is defined by

$$WEMA_n(x_t) = WEMA_n(x_{t-1}) + 1/n (x_t - WEMA_n(x_{t-1})),$$

seeded by the simple mean on the first n points. Using this, the Average True Range is $ATR_t = WEMA_n(TR_t)$ and the directional indices are

$$+DI_t = 100 \frac{WEMA_n(+DM_t)}{ATR_t}, -DI_t = 100 \frac{WEMA_n(-DM_t)}{ATR_t}.$$

Instantaneous strength is

$$DX_t = 100 \frac{|+DI_t - -DI_t|}{+DI_t + -DI_t},$$

and the Average Directional Index is $ADX_t = WEMA_n(DX_t)$, which by construction lies in $[0, 100]$.

An optional price–action confirmation uses the real-body fraction

$$\phi_t = \frac{max(C_t - O_t, 0)}{max(H_t - L_t, \varepsilon_{den})},$$

with a tiny $\varepsilon_{den} > 0$ guarding flat candles; hence $0 \le \phi_t \le 1$. Define the crossover residual $\Delta_t = E_t^{(f)} - E_t^{(s)}$. The long-only position state $S_t \in \{0,1\}$ changes only on zero-crossings of Δ_t subject to gates:

$$\text{enter } (S_t = 1) \Leftrightarrow \Delta_{t-1} \le 0, \Delta_t > 0, ADX_t \ge \acute{A}, \text{and, if enabled}, \phi_t \ge \beta,$$

$$\text{exit } (S_t = 0) \Leftrightarrow \Delta_{t-1} \ge 0, \Delta_t < 0,$$

and otherwise $S_t = S_{t-1}$. The parameters are $\Theta_{\text{trend}} = (\tau_f, \tau_s, n_{ADX}, \acute{A}, \beta)$ with $n_{ADX} \in Z_{+¿¿}$, $\acute{A} \in [0, 100]$, and $\beta \in [0, 1]$.

3.2 Mean-Reversion

The contrarian signal standardizes short-horizon deviations of the mid-price under weak-trend regimes. Let the mid-price be

$$M_t = 1/2 (H_t + L_t).$$

For a window length $n \in Z_{+¿¿}$, write μ_t and σ_t for the rolling mean and standard deviation of $\{M_{t-n+1}, \ldots, M_t\}$ (computed causally), and define the standardized deviation

$$z_t = \frac{M_t - \mu_t}{\sigma_t}, \sigma_t > 0.$$

A regime gate excludes strong trends via $ADX_t < \gamma$ with $\gamma \in (0, 100)$, and a short-horizon momentum filter admits entries only when the Relative Strength Index indicates local exhaustion, $RSI_t < \rho_\ell$ with $\rho_\ell \in (0, 100)$.

The RSI uses Wilder smoothing on up/down closes. With

$$u_t = max(C_t - C_{t-1}, 0), d_t = max(C_{t-1} - C_t, 0),$$

define $\overline{U}_t$ and $\overline{D}_t$ by the $WEMA_{n_{RSI}}$ recursion (seeded by simple means over the first n_{RSI} points), then

$$RS_t = \frac{\overline{U}_t}{\overline{D}_t}, RSI_t = 100 - \frac{100}{1 + RS_t} = 100 \frac{\overline{U}_t}{\overline{U}_t + \overline{D}_t},$$

with the numerical conventions $RSI_t = 100$ if $\overline{D}_t = 0$ and $RSI_t = 0$ if $\overline{U}_t = 0$, ensuring $RSI_t \in [0, 100]$.

The long-only state $S_t \in \{0,1\}$ is governed by three thresholds $0 < \theta_{\text{exit}} < \theta_{\text{entry}} < \theta_{\text{stop}}$ applied to z_t. Let "crosses down through a'' mean $z_{t-1} > a$ and $z_t \le a$; "crosses up through a'' means $z_{t-1} < a$ and $z_t \ge a$. Then

$$\text{enter } (S_t = 1) \Leftrightarrow z_t \text{ crosses down } -\theta_{\text{entry}}, ADX_t < \gamma, RSI_t < \rho_\ell,$$

$$\text{exit–reversion } (S_t = 0) \Leftrightarrow z_t \text{ crosses up } -\theta_{\text{exit}} \text{ under } ADX_t < \gamma,$$

$$\text{exit–stop } (S_t=0) \Leftrightarrow z_t < -\theta_{\text{stop}}.$$

Outside these first-arrival events the state is held constant, so trades align with discrete reversions rather than per-bar rebalancing. The parameters are $\Theta_{\text{rev}} = (n, \theta_{\text{entry}}, \theta_{\text{exit}}, \theta_{\text{stop}}, n_{ADX,z}, \gamma, n_{RSI}, \rho_\ell)$ with $n, n_{ADX,z}, n_{RSI} \in Z_{+¿¿}$ and $\theta_{\text{entry}} > \theta_{\text{exit}} > 0$, $\theta_{\text{stop}} > \theta_{\text{entry}}$.

3.3 Properties, initialization, and domains

All recursions are causal and bounded: $ADX_t \in [0,100]$, $RSI_t \in [0,100]$, and $\phi_t \in [0,1)$ by definition; $z_t \in R$ is unbounded but well-defined whenever $\sigma_t > 0$. The initial values for $WEMA_n(\cdot)$ statistics use simple means over their first windows, and EMAs initialize at the close at the end of warm-up. The trend and reversion state machines are long-only, $S_t \in \{0,1\}$, and change state exclusively on the algebraic crossing conditions shown above; no execution, sizing, or fee assumptions are part of the model definitions in this section.

4. Strategy Specifications

Both strategies execute an identical loop—fresh data ingestion, signal/state update, order hygiene, equity-proportional sizing, order submission, and risk enforcement—diverging only in the construction of the signal block. At each polling instant a reference price is selected from the latest trade if fresh; otherwise the bar close is used. If both references are stale, the iteration is skipped. Deterministic rounding at the price and quantity granularities ensures venue-admissible orders and prevents floating-point drift.

4.1 Trend-Following

Trend-Following (EMA–ADX long-only). The entry condition is an upward crossover of the fast over the slow EMA, gated by a minimum ADX and, optionally, a simple price-action filter; the exit condition is the symmetric downward crossover. Backtests apply frictional fills at $C_t(1 \pm \varepsilon)$ with per-trade fee f. Pseudo-live execution submits one patient GTC limit at an offset η from the chosen reference price and employs hysteresis: existing quotes are repriced only when drift exceeds δ_{rp} *and* age exceeds $T_{¿rp}$; orders are canceled at $\overline{T}_{life}$ or immediately on signal flip when cancel-on-flip is enabled. Sizing is equity-proportional: when $S_t = 1$, the notional target is $N_t = min(L_{max} W_t, \overline{N})$, quantized to $Q_t^\star = round_q(N_t / P_t)$; otherwise $N_t = 0$. Submissions enforce a per-trade cap, minimum notional $N_¿$, and quantization $(\delta p, \delta q)$; buys preserve a USD reserve and sells leave a small base-asset reserve. Risk overlays estimate one-day VaR from one-minute volatility via $VaR_\alpha \approx z_\alpha \hat{\sigma}_{min} \sqrt{1440} |Q_t| P_t$ with a variance floor; on breach the system cancels outstanding orders, flattens exposure, and enters cooldown. Crossing tests reference only adjacent bars and rely on $\Delta_t = E_t^{(f)} - E_t^{(s)}$ as defined in §3.

```
# Trend loop
S ← 0; last_signal ← None
every POLL_SEC:
  bars ← fetch_1m(); ref_px, fresh ← choose_ref_price(bars); if !
fresh: continue
```

```
  # SIGNAL (trend): EMA cross + ADX (+ optional PA)
  ema_f, ema_s, adx ← precompute_ema_adx(bars)
  cross_up   ← (ema_f[-2] ≤ ema_s[-2]) and (ema_f[-1] > ema_s[-1])
  cross_down ← (ema_f[-2] ≥ ema_s[-2]) and (ema_f[-1] < ema_s[-1])
  adx_ok ← adx[-1] ≥ ADX_MIN
  pa_ok  ← (¬USE_PA_FILTER) or price_action_ok(bars[-1], β)
  if S==0 and cross_up and adx_ok and pa_ok: S ← 1
  elif S==1 and cross_down:                  S ← 0
  if last_signal is None or S ≠ last_signal:
     if CANCEL_ON_FLIP: cancel_all_orders(); last_signal ← S

  # Sizing, hygiene, submission, risk
  W ← equity(); Q ← position_qty()
  N_tgt ← min(L_max*W, N_cap) if S==1 else 0
  Q* ← round_q(N_tgt/ref_px); ΔQ ← Q* - Q
  if |ΔQ| < max(2·10^{-QDEC}, N_min/ref_px): continue
  side ← BUY if ΔQ>0 else SELL
  if manage_open_order(side, ref_px): continue
  ΔQ ← clamp_trade_qty(MAX_Q_PER_TRADE, ΔQ, Q)
  if side==SELL: ΔQ ← min(ΔQ, free_qty_minus_reserve()); if ΔQ ≤ 0:
continue
  if |ΔQ|·ref_px < N_min: continue
  place_limit_order(side, |ΔQ|, ref_px, η)
  if breach_var_or_drawdown(): cancel_all_orders(); flatten_all();
cooldown()
```

4.2 Z-Score Reversion

Z-Score Reversion. Signals and state transitions follow §3.2 exactly. Execution mirrors the trend module (same frictional fills and quoting protocol), with cancel-on-flip being particularly important to avoid chasing momentum bursts in a contrarian regime. Sizing is the same equity-proportional scheme with $\left(L_{max}, \overline{N}, N_{¿}\right)$ and $(\delta p, \delta q)$; exits fully liquidate and any rounding residue is cleared on the next compatible signal. Risk overlays combine the structural stop at $-\theta_{stop}$ with the same VaR/exposure/drawdown framework. Implementation enforces a strictly positive variance floor $\sigma_{¿} > 0$ and allows optional symmetric winsorization of z_t to stabilize execution without altering §3.2 definitions.

```
# Reversion loop (same skeleton; different signal block)
S ← 0; last_signal ← None
every POLL_SEC:
  bars ← fetch_1m(); ref_px, fresh ← choose_ref_price(bars); if !
fresh: continue

  # SIGNAL (reversion): mid z, ADX/RSI gates
  mid ← (H+L)/2
  μ ← roll_mean(mid, Z_N); σ ← roll_std(mid, Z_N)   # σ floored in
code
```

```
  z ← (mid-μ)/σ
  adx ← compute_adx(bars, Z_ADX_LEN)
  rsi ← compute_rsi(C, RSI_LEN)

  th_e = -|Z_ENTRY|; th_x = -|Z_EXIT|; th_st = -|Z_STOP|
  x_dn(a) ← (z[-2] > a) and (z[-1] ≤ a);  x_up(a) ← (z[-2] < a) and (z[-1] ≥ a)
  adx_ok ← (adx[-1] < ADX_SKIP) or is_na(adx[-1])
  rsi_ok ← (rsi[-1] < RSI_LOWER) or is_na(rsi[-1])

  if S==0 and x_dn(th_e) and adx_ok and rsi_ok:             S ← 1
  elif S==1 and ( (x_up(th_x) and adx_ok) or x_dn(th_st) ): S ← 0
  if last_signal is None or S ≠ last_signal:
     if CANCEL_ON_FLIP: cancel_all_orders(); last_signal ← S

  # Steps identical to trend: sizing, hygiene, submission, risk
  W ← equity(); Q ← position_qty()
  N_tgt ← min(L_max*W, N_cap) if S==1 else 0
  Q* ← round_q(N_tgt/ref_px); ΔQ ← Q* - Q
  if |ΔQ| < max(2·10^{-QDEC}, N_min/ref_px): continue
  side ← BUY if ΔQ>0 else SELL
  if manage_open_order(side, ref_px): continue
  ΔQ ← clamp_trade_qty(MAX_Q_PER_TRADE, ΔQ, Q)
  if side==SELL: ΔQ ← min(ΔQ, free_qty_minus_reserve()); if ΔQ ≤ 0: continue
  if |ΔQ|·ref_px < N_min: continue
  place_limit_order(side, |ΔQ|, ref_px, η)
  if breach_var_or_drawdown(): cancel_all_orders(); flatten_all(); cooldown()
```

4.3 Numerical Techniques Used

Numerical Techniques. All rolling statistics use single-pass recursions with $O(T)$ time and $O(1)$ memory. EMA weights use $\alpha(\tau)=2/(\tau+1)$, and Wilder smoothing uses $\alpha=1/n$. Standardization employs stable updates with a floor $\sigma>0$ and optional symmetric winsorization. Any resampling (for summaries) respects OHLC composition and uses right-closed/right-labeled timestamps. Deterministic operators $round_p(\cdot)$ and $round_q(\cdot)$ enforce tick/lot admissibility and prevent floating-point accumulation.

```
function round_p(x, PRICE_DEC): return round(x, PRICE_DEC)
function round_q(x, QTY_DEC):   return max(0, round(x, QTY_DEC))
function floor_q(x, QTY_DEC):   step = 10^{-QTY_DEC}; return max(0, floor((x+1e-12)/step)*step)
function desired_limit(side, ref_px, η):
    off = 1-η if side==BUY else 1+η
    return round_p(ref_px * off, PRICE_DEC)

function clamp_trade_qty(max_per_trade, ΔQ, pos_qty):
```

```
    if ΔQ > 0: return min(ΔQ, max_per_trade)                      # 
BUY
    if ΔQ < 0: return min(|ΔQ|, max_per_trade, max(0,pos_qty))    # 
SELL
    return 0

function breach_limits():
    # checks VaR, exposure, rolling drawdown; returns True if any 
breached
    ...
```

5. Backtesting Methodology

5.1 Sample Definition and Bar Policy

The evaluation is a single fixed-parameter historical simulation on a contiguous window $T=$¿ using 5-minute OHLC bars. Let (O_t, H_t, L_t, C_t) denote observations on this grid and set $P_t \equiv C_t$. No training/validation/test split and no walk-forward are employed; hyperparameters are chosen ex ante and held constant over T. To suppress start-up transients, a warm-up of n_{max} bars—equal to the largest lookback across EMA/ADX/RSI and the rolling statistics used in the reversion leg—is discarded; all statistics are computed on T¿ with $|W| = n_{max}$. Bars are right-closed/right-labeled to prevent overlap and to ensure unambiguous state updates.

5.2 Transaction Costs and Slippage

Frictions are modeled explicitly. With proportional fee rate f and symmetric slippage ε, buys fill at $P_t^{buy} = C_t(1+\varepsilon)$ and sells at $P_t^{sell} = C_t(1-\varepsilon)$, and fee cash flows equal $f|q_t|C_t$ on notional $|q_t|C_t$. When enabled, an impact term augments the execution reference by $\lambda\, sign(q_t)|q_t|^\alpha$ so that

$$P_t^{exec} = P_t^{mid} + 1/2\, sign(q_t)\, spread_t + \lambda\, sign(q_t)|q_t|^\alpha,$$

which makes realized PnL size-dependent. Exchange admissibility is enforced by a price tick δp and lot step δq, and an optional latency proxy Δt_{lat} shifts the market reference from t to $t + \Delta t_{lat}$.

5.3 Position Sizing and Leverage

Sizing is equity-proportional with explicit caps. For equity W_t, last price P_t, and leverage bound $L_{max} \geq 1$, the target notional under a long signal is

$$N_t = min(L_{max} W_t, \overline{N}), Q_t^\star = round_q\left(\frac{N_t}{P_t}\right),$$

and the submitted delta is $\Delta Q_t = Q_t^\star - Q_{t^-}$. Submissions respect a per-trade clamp, a minimum notional N¿ to suppress micro-orders, a small USD reserve on buys to accommodate fees/rounding, and a few base-asset ticks retained on sells to avoid dust from quantization.

5.4 Execution Model and Accounting

Research fills are frictional "market-like" per §5.2. Between trades the wealth process evolves as

$$W_t = W_{t-1} + Q_{t-1}\left(C_t - C_{t-1}\right),$$

with discrete jumps at fills for slippage, fees, and rounding. Pseudo-live logic (reported separately) uses patient GTC limits at a relative offset $\eta > 0$ from the last trade, maintained by hysteresis: replace only if relative drift ¿δ_{rp} and order age ¿T_{rp}; cancel at $\overline{T}_{life}$ or on signal flip. This preserves path consistency between signal and execution layers.

5.5 Hyperparameter Policy

No cross-validation, grid search, or walk-forward selection is performed. All hyperparameters—e.g., $\left(\tau_f, \tau_s, n_{ADX}, \acute{A}, \beta\right)$ for the trend leg and $\left(n, \theta_{entry}, \theta_{exit}, \theta_{stop}, n_{ADX,z}, \gamma, n_{RSI}, \rho_\ell\right)$ for the reversion leg—are fixed a priori and logged. Any sensitivity sweeps shown are diagnostic and do not alter the headline configuration.

5.6 Benchmarks and Reporting

Results are presented in absolute terms (cumulative PnL, annualized volatility and Sharpe from 5-minute log-returns, turnover, and drawdown). For context a buy-and-hold of BTC/USD on the same grid and with identical friction assumptions is reported; a cash baseline is trivial with $r_t \equiv 0$. Annualizations use $N_{yr} = 525,600/5$ bars per year, so that $\hat{\sigma}_{ann} = \hat{\sigma}_{bar}\sqrt{N_{yr}}$ and $\widehat{SR}_{ann} = \sqrt{N_{yr}}\acute{r}_{bar}/\hat{\sigma}_{bar}$. *Buy-and-hold is evaluated on the same bar set with the same frictions and quantization.*

5.7 Timezone Convention

All timestamps are in UTC. Aggregations, freshness checks, and calendar-level summaries therefore reference midnight at $00:00$ UTC; expected bar counts follow a 24×7 schedule, and deviations trigger data-quality alarms. The UTC convention is applied uniformly to signals, execution, and reporting so that latencies and boundaries are measured on a single, consistent clock.

6. Results

We evaluate both components on BTC/USD at a 5-minute sampling frequency over the UTC window ¿. Unless stated otherwise, all figures are **net** of the backtest frictions described in §5 (fee $f = 10$ bps on notional; symmetric slippage $\varepsilon = 2$ bps per side), with bars right-closed/right-labeled. Performance statistics are computed from bar returns and annualized using $N_{yr} = 525,600/5 = 105,120$ bars per year so that $\widehat{SR} = \sqrt{N_{yr}}\acute{r}_{bar}/\hat{\sigma}_{bar}$.

6.1 Trend-Following (EMA–ADX with price-action gate)

Parameters used in the run are $\left(\tau_f, \tau_s, n_{ADX}, \acute{A}, \beta\right) = \left(13, 150, 7, 26.0, 0.5\right)$ with the price–action filter enabled; global run controls are $\text{initial cash} = \$100,000$, $\text{min notional} = \$5$, $\text{price\_dec} = 2$, $\text{qty\_dec} = 6$. Results over the evaluation window are $N_{\text{bars}} = 100,382$, $N_{\text{trades}} = 467$, total return

¿51.91%, Sharpe $\widehat{SR}=2.13$, and maximum drawdown $MDD=-13.59\%$. The buy-and-hold reference on the same window attains 91.05% with $\widehat{SR}_{BH}=1.64$ and $MDD_{BH}=-31.58\%$. The trade count implies an average inter-trade spacing of about $100,382/467\approx 215$ bars (≈ 18 hours at 5-minute resolution), consistent with a regime-based rather than high-turnover profile. Trend-following backtest summary

6.2 Z-Score Reversion (mid-price standardization with ADX/RSI gates)

Parameters are $(n,\theta_{entry},\theta_{exit},\theta_{stop},n_{ADX,z},\gamma,n_{RSI},\rho_\ell)=(288,2.4,0.0,3.5,14,30.0,14,30.0)$; $RSI_{upper}=70$ is configured but unused in the long-only entry. The same global controls apply ($\text{initial cash}=\$100,000$, $\text{min notional}=\$5$, $\text{price\_dec}=2$, $\text{qty\_dec}=6$). Results are $N_{\text{bars}}=100,382$, $N_{\text{trades}}=164$, total return ¿37.29%, Sharpe $\widehat{SR}=2.65$, and $MDD=-7.55\%$. The buy-and-hold comparator records 91.54% with $\widehat{SR}_{BH}=1.65$ and $MDD_{BH}=-31.58\%$. The average spacing is about $100,382/164\approx 610$ bars (~ 2 days), reflecting selective entries at z-threshold crossings under weak-trend and locally oversold conditions.Z-score reversion backtest summary

6.3 Comparative interpretation

Across the common window the trend follower captures a larger share of the directional move (total return 51.91% versus 37.29%) but with higher drawdown, whereas the reversion leg delivers the superior risk-adjusted profile ($\widehat{SR}=2.65$ versus 2.13) and roughly halves the drawdown (-7.55% versus -13.59%). In both runs, improvements over buy-and-hold in $\widehat{SR}$ persist despite explicit frictions and quantization, indicating that timing rather than leverage or fee omission drives the results.

7. Stress Testing

Stress testing interrogates the sensitivity of realized performance to deviations from the implementation choices present in the codebase. The canonical input is the UTC, right-closed/right-labeled 5-minute OHLC stream $\{(O_t,H_t,L_t,C_t)\}_{t=1}^{T}$; research frictions match the backtester with proportional fees $f=0.1$ bps and symmetric slippage $\varepsilon=0.02$ bps per side, while price/quantity admissibility is enforced by ticks $(\delta p,\delta q)$ implied by $\text{PRICE\_DEC}=2$ and $\text{QTY\_DEC}=6$. The pseudo-live loop adopts a freshness gate: the reference price is the latest trade when its age does not exceed $\text{TRADE\_STALE\_SEC}=180$ seconds, otherwise the most recent bar is used provided its age is below $\text{BAR\_FRESH\_SEC}=180$ seconds; the loop skips entirely when both are stale. A stress experiment applies a transformation S_θ to these gates or to the input path, replays the state machines, and yields stressed summaries $\{W_T^{(\theta)},MDD^{(\theta)},\widehat{SR}^{(\theta)},VaR_\alpha^{(\theta)}\}$.

7.1 Microstructure and execution stresses

Execution realism is studied by perturbing the very controls that govern order placement in pseudo-live. The limit offset $\eta=\text{LIMIT\_AWAY\_PCT}$ (default 0.001) is scaled to $\eta'=\kappa_\eta\eta$ with $\kappa_\eta\in[0.5,3]$ to traverse the adverse-selection versus non-execution trade-off. Order-maintenance hysteresis is tightened or loosened by adjusting the minimum replace age

$\mathrm{MIN\_REPRICE\_SEC}$ (default 45 s), the relative drift threshold $\mathrm{REPRICE\_PCT}$ (default 5 bps), and the lifetime cap $\mathrm{MAX\_LIFETIME\_SEC}$ (default 900 s). Data freshness is stressed by widening or shrinking the trader's thresholds relative to the data-guard service, which uses $\mathrm{TRADE\_STALE\_SEC}=1800$ s and $\mathrm{BARS\_FRESH\_SEC}=180$ s, thereby quantifying the consequence of acting on increasingly stale references. Episodes of insufficient balance are emulated by increasing the side-specific cooldown $\mathrm{INSUFF\_COOLDOWN\_SEC}$ (default 60 s), after which the resulting missed entries and the induced change in time-under-water are recorded. Backtest-side sensitivity to symmetric slippage is summarized by the elasticity

$$E_\varepsilon=\frac{\partial W_T}{\partial \varepsilon}\frac{\varepsilon}{W_T},$$

estimated via finite differences at $\varepsilon'=\kappa_\varepsilon \varepsilon$ with $\kappa_\varepsilon \ge 1$. Robustness is reported as the smallest κ_ε for which $\widehat{SR}^{(\kappa_\varepsilon)}<1$ or $MDD^{(\kappa_\varepsilon)}<-25\%$. Because the live loop models execution implicitly through $(\eta,\mathrm{MIN\_REPRICE\_SEC},\mathrm{REPRICE\_PCT},\mathrm{MAX\_LIFETIME\_SEC})$ rather than an explicit spread term, slippage stress in live mode is effected by those knobs rather than by multiplying a quoted spread.

7.2 Data quality, gaps, and liquidity droughts

The data-guard worker supplies crypto-specific integrity rules with defaults $\mathrm{LOOKBACK\_MIN}=120$, $\mathrm{GAP\_TOL\_MIN}=10$, $\mathrm{GAP\_RECENT\_MIN}=60$, $\mathrm{GAP\_PCT\_WARN}=50$, $\mathrm{GAP\_PCT\_FAIL}=80$, $\mathrm{SPREAD\_BPS\_CAP}=50$, $\mathrm{JUMP\_WARN\_BPS}=300$, and $\mathrm{JUMP\_FAIL\_BPS}=1500$. Stress is administered by tightening these thresholds to reject more sessions or by loosening them to admit noisier ones, then measuring the consequences for trading halts, time-under-water, and drawdown. Liquidity droughts and discontinuities are simulated by injecting controlled gaps into closes, $C'_t=C_t(1+J_t)$ with heavy-tailed J_t, enforcing OHLC feasibility by $O'_t=C'_{t-1}$, $H'_t\ge max(O'_t,C'_t)$, and $L'_t\le min(O'_t,C'_t)$. Matching halts are emulated by setting a null fill-hazard over selected intervals, during which marketable flow experiences additional delay and patient limits expire under the live hysteresis.

7.3 Parameter and regime stresses

Brittleness with respect to calibration is evaluated by perturbing precisely the hyperparameters consumed by the code. For the trend follower, $(\tau_f,\tau_s,n_{ADX},\acute{A},\beta)$ are shocked around $(13,150,7,26.0,0.5)$. For the reversion leg, $(n,\theta_{entry},\theta_{exit},\theta_{stop},n_{ADX,z},\gamma,n_{RSI},\rho_\ell)$ are shocked around $(288,2.4,0.0,3.5,14,30.0,14,30.0)$. The robustness frontier is summarized by

$$R(\Delta)=\min_{\|\delta\|\le\Delta}\widehat{SR}^{(\delta)},MDD_{max}(\Delta)=\max_{\|\delta\|\le\Delta}\vee MDD^{(\delta)}\vee,$$

with $\|\cdot\|$ defined on natural parameter scales (span differences for EMAs and ADX; absolute threshold changes for z). Regime sensitivity is probed by scaling the minute-volatility proxy used by the risk worker, $r_t^{(1m)}\mapsto\kappa_\sigma r_t^{(1m)}$, and recording how often the ADX and, for reversion, the RSI gates bind as κ_σ varies, together with the resulting changes in turnover and drawdown.

7.4 Resampling, bootstrap, and model uncertainty

Sampling uncertainty and serial dependence are quantified by a stationary (geometric-block) bootstrap of bar returns $r_t = \ln C_t - \ln C_{t-1}$. Each resample replays the discrete state machine to produce $\{\widehat{SR}^{¿}, MDD^{¿}\}$ and percentile bands; Sharpe uncertainty is additionally reported with a serial-correlation adjustment. Extreme-tail risk is estimated by a peaks-over-threshold model: exceedances above u are fit with a generalized Pareto distribution with parameters (ξ, β), yielding the expected shortfall

$$ES_\alpha = \frac{q_\alpha}{1-\xi} + \frac{\beta - \xi u}{1-\xi}, \xi < 1,$$

where q_α denotes the α-quantile of the fitted tail law.

7.5 Risk-worker and combined shocks

Risk governance follows the deployed worker, which computes a one-day parametric value-at-risk from minute bars with window $\mathrm{VAR\_WL} = 1440$ and confidence $\mathrm{VAR\_CONF} = 0.95$, and enforces $\mathrm{MAX\_VAR\_PCT} = 0.20$, $\mathrm{MAX\_DRAWDOWN\_PCT} = 0.30$, optional $\mathrm{MAX\_EXPOSURE\_USD}$ (default 0, i.e., disabled), with $\mathrm{ENFORCE} = 1$ and $\mathrm{CANCEL\_OPEN\_BEFORE\_FLATTEN} = 1$. Combined scenarios stack execution and data shocks with regime shifts by concurrently modifying η, the hysteresis triplet $(\mathrm{MIN\_REPRICE\_SEC}, \mathrm{REPRICE\_PCT}, \mathrm{MAX\_LIFETIME\_SEC})$, the freshness gates $(\mathrm{TRADE\_STALE\_SEC}, \mathrm{BAR\_FRESH\_SEC})$, and the volatility scale κ_σ. Acceptance in a scenario is recorded when no forced flattening occurs; if a breach triggers enforcement, the residual drawdown accrued until cooldown is bounded by a preset tolerance and included in the stress ledger.

7.6 Reporting and acceptance criteria

Results are reported as deltas to the baseline, $\Delta \widehat{SR}(\theta) = \widehat{SR}^{(\theta)} - \widehat{SR}$, $\Delta MDD(\theta) = MDD^{(\theta)} - MDD$, and changes in time-under-water and data-guard halt counts. For a deployment envelope Θ tied to actual configuration keys—$\eta \in [0.0005, 0.003]$, $\mathrm{MIN\_REPRICE\_SEC} \in [15, 120]$, $\mathrm{REPRICE\_PCT} \in [0.0002, 0.001]$, $\mathrm{MAX\_LIFETIME\_SEC} \in [300, 1800]$, $\mathrm{TRADE\_STALE\_SEC} \in [60, 600]$, $\mathrm{BAR\_FRESH\_SEC} \in [60, 600]$, $\mathrm{SPREAD\_BPS\_CAP} \in [20, 100]$, $\mathrm{JUMP\_FAIL\_BPS} \in [800, 1500]$, and parameter perturbations around the baselines above—acceptance requires

$$\inf_{\theta \in \Theta} \widehat{SR}^{(\theta)} \geq 1, ¿\theta \in \Theta \vee MDD^{(\theta)} \vee \leq 0.25, ¿\theta \in \Theta \frac{VaR^{(\theta)}_{0.95}}{W_0} \leq 0.20,$$

together with reconciliation errors $|\delta Q_t|$ bounded by one lot tick during intentional websocket drops and reconnect back-offs. Because stresses operate through the same environment variables and state machines as the backtester, pseudo-live trader, and monitoring workers, the resulting robustness frontier directly informs conservative production settings for η, hysteresis, freshness gates, and risk limits prior to capital scaling or multi-asset generalization.

8. Part II — Pseudo-Live Trading & Broker API

8.1 Integration and Time Base

The pseudo-live executor interfaces with Alpaca Crypto via `alpaca-py` (≥ 0.21), separating market data (`CryptoHistoricalDataClient`) from order flow (`TradingClient`) to reduce coupling and simplify failure isolation. The process is fully environment-driven and runs 24×7 without market-hours gating. Paper routing is the default with `ALPACA_PAPER=1`; setting `ALPACA_PAPER=0` enables live submission. Symbols are normalized deterministically: data endpoints use slashed forms such as `"BTC/USD"`, trading endpoints use unslashed forms such as `"BTCUSD"`, with `symbol_for_data` and `symbol_for_trade` providing a single source of truth. All timestamps are in UTC, and the loop cadence is governed by `POLL_SEC` (default 5 s).

8.2 Reference Price and Data Freshness

Each cycle refreshes a short bar window of length `BAR_LOOKBACK_MIN` minutes and queries the latest trade. A freshness gate selects the reference price: the latest trade is used when its age is at most `TRADE_STALE_SEC` (default 180 s); otherwise the most recent bar close is used when the bar age is at most `BAR_FRESH_SEC` (default 180 s); when both are stale the iteration is skipped. This policy prevents actions on degraded feeds while keeping indicators aligned with the backtest.

8.3 Signal Parity with Research

Signal computation mirrors the backtest exactly to preserve path consistency. The trend leg implements an EMA crossover gated by ADX with an optional price–action confirmation; the reversion leg computes a mid-price z-score with ADX and RSI gates. Defaults are supplied via the environment: `EMA_FAST=20`, `EMA_SLOW=50`, `ADX_LEN=14`, `ADX_MIN=25.0`, `USE_PA_FILTER=1`, `PA_MIN_BODY_FRAC=0.5`, `Z_N=50`, `Z_ENTRY=2.0`, `Z_EXIT=0.5`, `Z_STOP=3.0`, `ADX_SKIP=30.0`, `Z_ADX_LEN` falling back to `ADX_LEN`, and `RSI_LEN=14` with `RSI_LOWER=30.0`. The active component is selected by `STRATEGY∈{ema,zscore}`.

8.4 Order Submission Model

Under normal operation only `GTC` limit orders are used; time-in-force is set by `DEFAULT_TIF` and defaults to `GTC`. For a reference price P_t, limits are placed symmetrically at

$$L_t^{buy} = P_t(1 - \mathrm{LIMIT\_AWAY\_PCT}),\ L_t^{sell} = P_t(1 + \mathrm{LIMIT\_AWAY\_PCT}),$$

with $\mathrm{LIMIT\_AWAY\_PCT} = 10^{-3}$ by default. All submissions carry idempotent client identifiers `psl-<symbol>-<epoch>-<side>` for safe retries and clean reconciliation. Partial fills are handled natively; the remaining quantity is $q_{rem} = max(0, q_{sub} - q_{filled})$ and is honored in subsequent alignment steps. Market orders are reserved for emergency flattening in the risk worker, not for routine execution.

8.5 Sizing, Quantization, and Affordability

Sizing is equity-proportional, capped, and quantized to venue constraints. For long/flat signals $s_t \in \{0,1\}$ the target notional is

$$N_t = min\left(s_t \cdot \text{MAX\_LEVERAGE} \cdot equity_t, \text{USD\_NOTIONAL\_CAP}\right), q_t = \frac{N_t}{P_t}.$$

Prices are rounded to `PRICE_DEC` and quantities to `QTY_DEC`. Orders with notional below `MIN_NOTIONAL_USD` are suppressed, and per-trade size is bounded by `MAX_QTY_PER_TRADE`. On sells, available inventory is computed from broker-reported `qty_available` (falling back to total when absent) less the **remaining** quantity on open sell orders; a small reserve of `BASE_QTY_RESERVE_TICKS` prevents dust after flooring. On buys, an affordability haircut is applied: available USD is reduced by `AVAILABLE_SAFETY_HAIRCUT`, outstanding buy notional, and a `USD_CASH_RESERVE`, then floored to lot size before submission. These checks precede tick/lot rounding so that all orders are admissible.

8.6 Order Hygiene and Hysteresis

Open orders are maintained with explicit hysteresis to avoid churn. States that reserve funds or quantity—“new”, “partially_filled”, “accepted”, “pending_new”, “accepted_for_bidding”, “pending_replace”, and “held”—are treated as open. Repricing occurs only when the relative drift $\Delta = ¿ L_{have} - L_{want} \vee ¿ max\left(L_{want}, \varepsilon\right)$ exceeds `REPRICE_PCT` and the order age exceeds `MIN_REPRICE_SEC`; defaults are 5 bps and 45 s. In-place replacement is preferred; otherwise the order is canceled and reissued. A practical staleness rail `CANCEL_AFTER_SEC` and an absolute lifetime cap `MAX_LIFETIME_SEC` retire orders that drift or age excessively. Single-sided discipline is enforced: only one healthy same-side order may work at a time, and all resting orders are purged on signal flips when `CANCEL_ON_FLIP=1`.

8.7 Connectivity, Listeners, and Idempotence

The executor is an idempotent REST poller augmented by an event-driven order-update stream. The listener subscribes to trade updates, deduplicates messages on $(\text{order\_id}, \text{event}, \text{ts})$, and reconnects with exponential backoff capped by `RECONNECT_MAX_SEC`. In-flight submissions are tracked for `INFLIGHT_TTL_SEC` to prevent duplicate placements across transient disconnects. The main loop remains time-paced, which limits sensitivity to microbursts and broker latencies.

8.8 Exceptions and Reconciliation

Broker responses are parsed for actionable diagnostics. The “insufficient balance” condition (code `40310000`) triggers a side-level cooldown keyed by $(\text{symbol}, \text{side})$ for `INSUFF_COOLDOWN_SEC`, preventing immediate resubmission against reserved funds. Position reconciliation prefers `qty_available` when present; otherwise it uses total quantity and subtracts still-open same-side remainder so that resultant sells never exceed free inventory.

8.9 Risk and Health Workers

Risk and data quality are enforced in dedicated processes to reduce live/backtest divergence. The data-guard monitors minute-bar integrity, trade and bar freshness, spreads, jump magnitudes, and infeasible candle geometry using crypto-appropriate thresholds such as `TRADE_STALE_SEC=180` and `BARS_FRESH_SEC=180`. The risk worker estimates one-day value-at-risk from recent one-minute returns and scales by $\sqrt{1440}$, tracks exposure and rolling drawdown, and—when `ENFORCE=1`—first cancels open orders and then flattens the position if $VaR_{\alpha}/W_t > \mathrm{MAX\_VAR\_PCT}$, if absolute exposure exceeds `MAX_EXPOSURE_USD`, or if drawdown breaches `MAX_DRAWDOWN_PCT`. Emergency flattening respects the same tick/lot rounding and may pre-cancel resting orders with `CANCEL_OPEN_BEFORE_FLATTEN=1`.

8.10 Safety Modes and Opt-In Live Routing

Live routing is explicitly opt-in. Paper trading is enabled by default, and `DRY_RUN=1` prints the intended action—side, quantity, limit, time-in-force—without touching the broker. Conservative limit placement via $\mathrm{LIMIT\_AWAY\_PCT}$, two-threshold hysteresis, minimum notional checks, side-level cooldowns, affordability haircuts, and single-sided order discipline collectively ensure that, even when live submission is enabled, orders remain conservative, quantized, and rate-limited by both time and price drift.

9. Part III — Deployment, Monitoring & Risk

9.1 Containerization and orchestration

The system is deployed as a multi-service Docker composition comprising a PostgreSQL database, a batch backtester, the pseudo-live trader, an order-update listener, a risk worker, a data-quality guard, and a dashboard. Each service runs in its own container, inherits configuration from a shared project-level `.env`, and binds a versioned schema applied at boot via SQL migrations. Application containers depend on the database healthcheck and join an internal bridge network; only the dashboard is published externally. Secrets are injected through environment variables or Docker secrets rather than image layers, structured logs are written to bind-mounted volumes, and stateless processes can scale horizontally, whereas event-ordered components (e.g., the order listener) are kept singleton to preserve causality.

9.2 Positions tracking and reconciliation

Inventory is triangulated from periodic broker account snapshots, the real-time trade-update stream, and an internal event-sourced ledger keyed by stable client order identifiers. If Q_t^{acct} denotes broker-reported holdings at time t and O_t^{open} is the set of active sell orders with remaining quantities $\{q_{t,i}^{rem}\}$, then the free-to-sell quantity is

$$Q_t^{free} = Q_t^{acct} - \sum_{i \in O_t^{open}} q_{t,i}^{rem}.$$

Independently, deterministic replay of fills yields Q_t^{replay}, and reconciliation computes $\delta Q_t = Q_t^{acct} - Q_t^{replay}$. Small $¿\delta Q_t \vee ¿$ within one lot are tolerated as quantization dust; persistent discrepancies escalate from forced re-sync, to a trading freeze with order cancellations, and, if policy permits, to a quantized corrective trade that flattens the mismatch. All transitions are durably recorded for audit.

9.3 Risk telemetry and dashboard

Risk and performance are rendered from the relational store as time-aligned series with coherent definitions. Mark-to-market profit and loss is

$$PnL_t = \sum_{s \le t} Q_{s^-} \left(C_s - C_{s-1} \right) - \sum_{\text{fills} s \le t} \text{fees}_s ,$$

with equity $W_t = W_0 + PnL_t$ and instantaneous drawdown $DD_t = 1 - \frac{W_t}{max_{u \le t} W_u}$. Turnover annualizes from traded notional as

$$\widehat{TO} = \frac{N_{yr}}{T} \sum_{t=1}^{T} \frac{¿ \Delta Q_t \vee C_t}{\overline{W}}, N_{yr} = \frac{525,600}{m},$$

where m is the bar size in minutes and $\overline{W}$ is average equity. Sharpe is computed from bar returns r_t via

$$\widehat{SR} = \sqrt{N_{yr}} \frac{\acute{r}}{\hat{\sigma}_r} .$$

A one-day parametric value-at-risk is displayed as

$$VaR_{\alpha,t} \approx z_\alpha \hat{\sigma}_{min,t} \sqrt{1440} \vee Q_t \vee C_t ,$$

with $\hat{\sigma}_{min,t}$ estimated from recent one-minute returns and z_α the one-sided standard normal quantile. Breaches of exposure budgets, drawdown ceilings, or $VaR_{\alpha,t} / W_t$ limits are surfaced and mirrored into the enforcement loop.

9.4 Scheduling, liveness, and health

Operational cadence follows cron-style scheduling with jitter to avoid phase locking. The trader's control loop is time-paced and idempotent; the order listener consumes and persists trade-update events; the risk worker recalculates limits on a short interval (e.g., every 60 s); and an account snapshotter runs more frequently (e.g., every 30 s). Health is asserted by database availability and by application-level freshness checks on the last market tick, websocket subscription status, database write latency, and event backlogs. Degradations trigger exponential-backoff restarts; trading is suspended when freshness or reconciliation guarantees are not satisfied. Dependencies ensure application services start only after the database is healthy, providing a deterministic boot graph.

9.5 Risk enforcement

Portfolio-level brakes run out-of-process to decouple execution and control. Limits include a cap on VaR_α / W_t, an absolute exposure ceiling, and a rolling peak-to-trough drawdown bound. With enforcement enabled, a breach triggers a deterministic sequence: cancel resting orders (optionally via `CANCEL_OPEN_BEFORE_FLATTEN=1`), flatten the position with a single market order sized to venue ticks and lots, and apply a cooldown. The VaR engine estimates minute-scale volatility $\hat{\sigma}_{min,t}$ over a sliding window and scales it by $\sqrt{1440}$; the one-sided confidence level determines z_α. All thresholds and windows are externally configurable and logged alongside actions to preserve auditability.

9.6 Data-quality rules and suspend logic

A dedicated data-guard process enforces algebraic and economic invariants in real time. OHLC geometry must satisfy $H_t \geq max(O_t, C_t)$ and $L_t \leq min(O_t, C_t)$ with non-negative volume and spreads; timestamps must be strictly increasing at the configured bar size. Freshness checks assert that the latest trade is no older than a prescribed threshold and that bars have arrived within a short grace period. Gap detection counts consecutive missing minutes and the fraction of missing minutes over a recent window; spread and jump monitors operate in basis points with distinct warn/fail bands. Violations escalate from warn (telemetry only), to hold (block new entries while permitting exits), to freeze (cancel and flatten), ensuring no trading proceeds on ambiguous or infeasible inputs. All evaluations use UTC so that clocks, boundaries, and latencies are consistent across backtest and live operation.

9.7 End-to-end guarantees

Path consistency arises from using the same state machine in research and pseudo-live, explicit modeling of frictions and exchange quantization, conservative limit placement with two-threshold hysteresis, and idempotent order flow keyed by stable client IDs. Centralized, durable storage of bars, orders, fills, snapshots, and run metadata—paired with deterministic reconciliation—yields an auditable pipeline in which every live decision and reported metric is attributable to reproducible inputs and a fully specified configuration.

10. Limitations

10.1 Scope and market microstructure

The empirical evidence is restricted to a single instrument, BTC/USD, sampled exclusively on a 5-minute grid and routed through a single broker path. Crypto liquidity is fragmented and venue microstructure differs materially across exchanges; spreads, depth, and matching semantics on the routed venue need not coincide with a consolidated mid or with alternative books. Bar aggregation implicitly treats all prints within an interval as exchangeable; during thin or jumpy periods this smoothing can understate path risk and overstate fillability. The pipeline normalizes symbols and timestamps consistently across components, but the underlying route remains venue-specific and therefore non-representative of cross-venue execution.

10.2 Execution model approximation

The research engine prices frictional “market-like” fills at $C_t(1 \pm \varepsilon)$ with proportional fees f and optional impact $\lambda \vee q_t ¿^\alpha$, whereas the pseudo-live executor submits a single GTC limit per side at $P_t(1 \mp \eta)$ and maintains it via two-threshold hysteresis. Neither queue position, fill hazard, nor adverse selection are modeled explicitly in backtest; as a result, price improvement realized by resting limits may be overstated in simulation and understated in stress. Exchange quantization is enforced through $(\delta p, \delta q)$, but flooring at the lot granularity can interact with fee debits to leave small residual inventories between signal transitions. The latency proxy Δt_{lat} approximates network and broker delays yet cannot capture state-dependent throttling, congestion, or exchange halts. In contrast to the limit-only main loop, the risk worker is permitted to flatten via market orders upon breaches; this asymmetry is intentional for safety but introduces a gap between research fills and emergency live behavior.

10.3 Statistical interpretation and uncertainty

All results are descriptive of a single fixed-parameter historical run on a contiguous window with no train/validation/test split and no walk-forward. Hyperparameters are selected ex ante, computed causally, and then held constant; consequently no out-of-sample inference is claimed. Sharpe ratios are computed from bar returns and annualized under square-root-of-time scaling; the classical IID approximation yields $SE(\widehat{SR}) \approx \sqrt{(1+1/2\,\widehat{SR}^2)/T}$ for T bars, which is optimistic in the presence of serial correlation and conditional heteroskedasticity. Any bootstrap or sensitivity figures should therefore be read as exploratory diagnostics rather than formal confidence intervals.

10.4 Regime dependence and threshold brittleness

Signal behavior is regime-contingent. The EMA–ADX module underperforms in range-bound regimes where the difference $\Delta_t = E_t^{(f)} - E_t^{(s)}$ changes sign frequently, increasing whipsaws. The z-score reversion module underperforms in persistent trends when z_t remains negative without reverting to $-\theta_{exit}$. Gating via $ADX_t \geq \acute{A}$ for trend and $ADX_t < \gamma$ with $RSI_t < \rho_\ell$ for reversion mitigates but does not eliminate these exposures, and any fixed threshold becomes brittle as the volatility scale shifts. Although standardization on the mid-price $M_t = 1/2(H_t + L_t)$ reduces close-to-close microstructure noise, it does not immunize the strategy against structural regime changes.

10.5 Risk model simplifications

Portfolio brakes rely on a parametric value-at-risk proxy,

$$VaR_\alpha \approx z_\alpha \hat{\sigma}_{min} \sqrt{1440} \vee Q_t \vee P_t,$$

with $\hat{\sigma}_{min}$ estimated from recent one-minute returns. This Gaussian scaling downplays fat tails, clustered jumps, and liquidation cascades that characterize crypto markets. Forced unwinds, partial fills under droughts, and fee tier changes propagate nonlinearly through realized PnL and are not fully captured by the proxy. Drawdown statistics condition on the realized path and do not bound prospective losses under halts or extreme spread inflation.

10.6 Data quality and timekeeping

All timestamps are in UTC and bars are right-closed/right-labeled, which standardizes aggregation and avoids daylight-saving artifacts; nonetheless, exchange clock skew, late prints, and out-of-order ticks can survive resampling. Algebraic filters that enforce $H_t \geq max(O_t, C_t)$, $L_t \leq min(O_t, C_t)$, non-negative volume, and winsorized returns reduce but do not eradicate data risk; they may also attenuate true extremes. Gap handling imputes short outages conservatively and suspends on longer outages, yet residual ambiguity remains at the boundary between imputation and exclusion.

10.7 Operational and integration risks

Broker APIs can reorder or drop websocket events, return transient $5xx$ errors, or accept cancels after fills, producing reconciliation gaps $¿\delta Q_t \vee ¿0$ even under idempotent client order IDs. The executor's side-level cooldowns, affordability haircuts, cancel-on-flip behavior, and in-flight deduplication reduce but cannot eliminate race conditions between polling, event ingestion, and broker state changes. Paper trading differs from live in matching logic, rebates, throttle limits, and error surfaces; measured behavior under paper does not guarantee linear extrapolation to production.

10.8 Generality and extensibility

Findings pertain to one symbol, one fee/slippage model, one route, and a long-only hyperparameter family. The system does not model shorts, perpetual funding, borrow costs, cross-venue latency, or multi-asset interactions. Extending to portfolios would require explicit treatment of cointegration, basis and funding dynamics, venue selection, and cross-asset risk budgeting; absent those elements, portability should be regarded as a hypothesis to be tested rather than a conclusion.

11. Further Development

11.1 Multi-asset, cross-venue generalization

A natural extension is to move from a single routed symbol to a consolidated, multi-asset stack. Let $P_t = \left(P_t^1, \ldots, P_t^k\right)^\top$ denote venue/asset closes and $r_t = \Delta \ln P_t$ their log-returns. Mean-reversion can be posed on stationary spreads $s_t = w^\top \ln P_t$ with w estimated via Johansen or ridge-penalized GLS under the budget $1^\top w = 0$, while trend acts on $w^\top P_t$ subject to per-asset caps. Derivatives support should incorporate basis $b_t = F_t - S_t$ and funding-aware carry for perpetuals with $E\left[r_{t+1}\right] \approx \mu_t + \text{funding}_t \Delta t$, so that optimal Q_t internalizes expected overnight costs and borrow constraints. Broker adapters should expose a uniform order interface while preserving venue-specific ticks, lots, and fee tiers already enforced in the code.

11.2 Queue-reactive placement and state-dependent hysteresis

The current executor (see `part2_pseudo_live.py`) uses a single GTC limit per side at a fixed offset η with two-threshold hysteresis (REPRICE_PCT, MIN_REPRICE_SEC, MAX_LIFETIME_SEC). A queue-reactive policy can replace fixed η by $\eta_t^\star$ learned from top-of-book depth $\left(B_t, A_t\right)$ and arrival intensities $¿$. With microprice $P_t^{¿} = \frac{A_t}{A_t + B_t} \text{ask}_t + \frac{B_t}{A_t + B_t} \text{bid}_t$ and imbalance $I_t = \frac{B_t - A_t}{B_t + A_t}$, choose

$$\eta_t^\star = \underset{\eta \geq 0}{arg\,max}\, E\left[\text{improvement}\left(\eta\right)\right] - c_{\text{miss}} P\left[\text{no fill} \mid \eta, \lambda_t^{\pm}\right],$$

and promote δ_{rp} from a constant to $\delta_{rp}(I_t, \lambda_t^{\pm})$ so that replaces occur only when queue risk warrants churn. Survival models for fill hazard can be trained offline and embedded as lightweight predictors online.

11.3 Model combination with online allocation

The two legs can be combined by an online meta-policy with weights $w_t \in [0,1]$ driven by regime features x_t (e.g., realized volatility, ADX, spread), $w_t = \sigma(\theta^\top x_t)$, producing $\pi_t = w_t \pi_t^{trend} + (1 - w_t) \pi_t^{rev}$. An experts-with-exponential-weights variant sets $w_t^{(i)} \propto \exp\{\eta \hat{g}_{1:t}^{(i)}\}$ for $i \in \{trend, rev\}$ and a gain proxy $\hat{g}$ regularized by drawdown. Position targets should remain quantized and capped as in the current sizing path, with portfolio budgets $\sum_a |Q_t^a| P_t^a \le L_{max} W_t$ and per-asset notional ceilings.

11.4 Validation and uncertainty quantification

The repository currently performs a single fixed-parameter run (no split/walk-forward in `part1_backtest.py`). To harden inference, implement purged, embargoed walk-forward and block bootstrap for serial dependence. Report Sharpe with serial-correlation adjustment and deflated Sharpe ratios to control multiple testing. Model confidence set procedures and superior predictive ability tests can be layered onto walk-forward segments to distinguish sampling variability from genuine skill.

11.5 Risk extensions beyond Gaussian VaR

The risk worker estimates a one-day parametric VaR from minute returns with $VaR_\alpha \approx z_\alpha \hat{\sigma}_{min} \sqrt{1440} |Q_t| P_t$. Extending to extreme-value methods on exceedances over a threshold u yields generalized-Pareto CVaR,

$$CVaR_\alpha = \frac{\beta + \xi(u - q_\alpha)}{1 - \xi} (\xi < 1),$$

which better reflects tail clustering and gap risk. Liquidity-aware stress should couple spread inflation, partial fills, and forced unwinds so that capital-at-risk limits bind on *executible* exposure, not just marked exposure.

11.6 Infrastructure, observability, and exactly-once

The multi-service composition (Docker + TimescaleDB) can be strengthened with an event-sourced outbox for broker actions and exactly-once ingestion of fills and order updates (see `part3_order_listener.py`). Health SLOs—tick freshness, websocket uptime, DB latency—should drive a circuit breaker that sets $S_t = 0$ and cancels working orders when any SLO violates thresholds. A deterministic replay harness that reprocesses raw bars and broker events enables differential testing between research and live binaries, closing the gap surfaced by the separate backtest (`part1_backtest.py`) and executor (`part2_pseudo_live.py`).

11.7 Data enrichment and slippage priors

Augmenting bars with depth D_t, spread s_t, funding/borrow rates, and cross-venue quotes enables microstructure-aware slippage priors, e.g.,

$$\varepsilon_t = \alpha_0 + \alpha_1 s_t + \alpha_2 D_t^{-1},$$

in place of a flat ε. These priors can feed both backtest realism and live affordability checks, harmonizing with the existing affordability haircut and USD reserve logic.

12. Conclusion

This study presented a fully specified, auditable stack for long-only systematic trading on BTC/USD built around two complementary components: an EMA–ADX trend follower with an optional price-action gate and a mid-price z-score reversion model gated by ADX/RSI. Both strategies share a single execution skeleton—freshness-guarded price selection, causal signal updates, order hygiene with two-threshold hysteresis, equity-proportional sizing under explicit caps, and out-of-process risk enforcement—implemented consistently in research and pseudo-live code paths with deterministic rounding and venue-admissible orders.

On a fixed 5-minute, UTC-clocked window, the trend follower delivered a net total return of 51.91% with $\widehat{SR}=2.13$ and $MDD=-13.59\%$, while the reversion leg produced 37.29% with $\widehat{SR}=2.65$ and $MDD=-7.55\%$. Against a buy-and-hold reference whose Sharpe was ≈ 1.6 and drawdown $\approx -31.6\%$, both modules improved risk-adjusted performance despite explicit frictions and quantization. The comparative pattern is clear: trend captures more of the directional move; reversion compresses drawdown and raises Sharpe, with lower turnover consistent with selective, regime-conditioned entries.

Stress programs tied directly to deployed knobs—limit offset η, hysteresis thresholds, freshness gates, affordability and inventory checks, and the risk worker's VaR/DD budgets—indicate that performance is resilient within a realistic operating envelope, while making explicit where deterioration first appears (non-execution from aggressive offsets, drawdown under spread inflation, or enforcement-triggered flattening). The separation of duties (trader, listener, data guard, risk) and idempotent, client-ID–keyed order flow provide path consistency and operational safety that are uncommon in backtest-only studies.

At the same time, the evidence is descriptive and in-sample. Results reflect one symbol, one route, one bar size, and fixed hyperparameters; backtest fills are stylized relative to live, and VaR is Gaussian. Data controls reduce but do not eliminate residual feed risk. Regime dependence persists: the trend leg can whipsaw in ranges; the reversion leg can underperform in persistent trends. These limitations bound external validity and argue for cautious interpretation of point estimates.

The immediate roadmap is concrete: unify bar policy across research and live; add purged, embargoed walk-forward with serial-correlation–adjusted and deflated Sharpe reporting; replace fixed η with queue-reactive placement η_t^{*} and state-dependent replacement bands; extend the risk layer with EVT-based $CVaR$; and introduce an event-sourced outbox with exactly-once ingestion and deterministic replay to close any remaining research/live gaps. Medium-term, lift the single-asset constraint by adding cross-venue consolidation, depth/funding enrichment, and portfolio-level allocation $w_t = \sigma\left(\theta^\top x_t\right)$ to adaptively blend trend and reversion under explicit risk budgets.

Under conservative sizing, hard brakes, and strict freshness/data-quality gates, the stack is production-capable for staged rollout (shadow → paper → small-notional live). The right interpretation of the present statistics is as *engineering telemetry*: they demonstrate that the design achieves path consistency and robust execution under realistic frictions. Moving from promising in-sample behavior to durable production performance now depends on completing the validation and infrastructure steps above and re-testing as regimes, liquidity, and fee surfaces evolve.

13. References

[1] Wilder, J. W. (1978). *New Concepts in Technical Trading Systems*. Trend Research (RSI, ADX).

[2] Brown, R. G. (1959). *Statistical Forecasting for Inventory Control*. McGraw–Hill (exponential smoothing/EMA).

[3] Johansen, S. (1991). Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. *Econometrica*, 59(6), 1551–1580.

[4] Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67.

[5] Avellaneda, M., & Stoikov, S. (2008). High-frequency trading in a limit order book. *Quantitative Finance*, 8(3), 217–224.

[6] Cont, R., Kukanov, A., & Stoikov, S. (2014). The price impact of order book imbalance. *SSRN Working Paper*.

[7] Cartea, Á., Jaimungal, S., & Penalva, J. (2015). *Algorithmic and High-Frequency Trading*. Cambridge University Press.

[8] Hasbrouck, J. (2007/2018). *Empirical Market Microstructure: The Institutions, Economics, and Econometrics of Securities Trading*. Oxford University Press.

[9] Bouchaud, J.-P., Bonart, J., Donier, J., & Gould, M. (2018). *Trades, Quotes and Prices: Financial Markets Under the Microscope*. Cambridge University Press.

[10] Politis, D. N., & Romano, J. P. (1994). The stationary bootstrap. *Journal of the American Statistical Association*, 89(428), 1303–1313.

[11] Newey, W. K., & West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3), 703–708.

[12] White, H. (2000). A reality check for data snooping. *Econometrica*, 68(5), 1097–1126.

[13] Hansen, P. R. (2005). A test for superior predictive ability. *Journal of Business & Economic Statistics*, 23(4), 365–380.

[14] Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The model confidence set. *Econometrica*, 79(2), 453–497.

[15] Bailey, D. H., & López de Prado, M. (2014). The deflated Sharpe ratio. *The Journal of Portfolio Management*, 40(5), 94–102.

[16] López de Prado, M. (2018). *Advances in Financial Machine Learning*. Wiley (purged K-fold CV, embargo, walk-forward).

[17] Embrechts, P., Klüppelberg, C., & Mikosch, T. (1997). *Modelling Extremal Events for Insurance and Finance*. Springer.

[18] McNeil, A. J., Frey, R., & Embrechts, P. (2015). *Quantitative Risk Management* (2nd ed.). Princeton University Press.

[19] Pickands III, J. (1975). Statistical inference using extreme order statistics. *The Annals of Statistics*, 3(1), 119–131. (GPD for POT)

[20] Balkema, A. A., & de Haan, L. (1974). Residual life time at great age. *The Annals of Probability*, 2(5), 792–804. (GPD limit)

[21] Kelly, J. L. (1956). A new interpretation of information rate. *Bell System Technical Journal*, 35(4), 917–926.

[22] MacLean, L. C., Thorp, E. O., & Ziemba, W. T. (Eds.). (2011). *The Kelly Capital Growth Investment Criterion: Theory and Practice*. World Scientific.

[23] Alpaca. (2025). *alpaca-py SDK & Crypto API Documentation*. Alpaca Securities LLC.

[24] Timescale. (2025). *TimescaleDB Documentation*. Timescale, Inc.

[25] Docker Inc. (2025). *Docker Documentation*. Docker, Inc.

Appendices

A. Parameter Spaces — Run Configuration Used

This appendix records the **exact** configuration used in the experiments. Secrets are kept as environment variables and are **omitted** from the report for security.

Secrets / Broker (omitted).
$\text{ALPACA\_API\_KEY\_ID}=\langle \text{redacted} \rangle, \text{ALPACA\_API\_SECRET\_KEY}=\langle \text{redacted} \rangle, \text{ALPACA\_PAPER}=1.$

Global / Common.
Symbol BTC/USD with crypto asset class; price and quantity quantization correspond to

$$\text{SYMBOL}=BTC/USD, \text{ASSET\_CLASS}=\text{CRYPTO}, \delta p=10^{-2}(\text{PRICE\_DEC}=2), \delta q=10^{-6}(\text{QTY\_DEC}=6),$$

Strategy selector: $\text{STRATEGY}=\text{ema}.$

Trend-Following (EMA–ADX with price–action gate).
Fast/slow spans, ADX gate, and candle filter are

$$\tau_f=\text{EMA\_FAST}=13, \tau_s=\text{EMA\_SLOW}=150, n_{ADX}=\text{ADX\_LEN}=7, \acute{A}=\text{ADX\_MIN}=26, \text{PA}=\text{USE\_PA\_FILT}$$

Z-Score Mean Reversion (long-only fade).
(Configured but not selected in this run.)

$$n = \text{Z\_N} = 288,\ z_{\text{entry}} = \text{Z\_ENTRY} = 2.4,\ z_{\text{exit}} = \text{Z\_EXIT} = 0.0,\ z_{\text{stop}} = \text{Z\_STOP} = 3.5,$$

$$n_{ADX,z} = \text{Z\_ADX\_LEN} = 14,\ \gamma = \text{ADX\_SKIP} = 30,\ n_{RSI} = \text{RSI\_LEN} = 14,\ \rho_{\ell} = \text{RSI\_LOWER} = 30,\ \rho_{u} = \text{RSI\_UPPE}$$

Sizing / Caps (shared).

$$L_{max} = \text{MAX\_LEVERAGE} = 1.0,\ N_{\text{¿}} = \text{MIN\_NOTIONAL\_USD} = \$5.$$

Live Trader (pseudo-live).
Good-'til-canceled limit placement with symmetric offset:

$$\text{DEFAULT\_TIF} = GTC,\ \eta = \text{LIMIT\_AWAY\_PCT} = 0.005\,(0.5\%),\ \text{BAR\_LOOKBACK\_MIN} = 240,$$

$$N_{max} = \text{USD\_NOTIONAL\_CAP} = \$5 \cdot 10^{5},\ Q_{max} = \text{MAX\_QTY\_PER\_TRADE} = 9.9999999 \times 10^{7},$$

$$t_{\text{cancel}} = \text{CANCEL\_AFTER\_SEC} = 300,\ \text{DRY\_RUN} = 0,\ \text{CANCEL\_ON\_FLIP} = 1.$$

Backtest.
One-year window with 5-minute bars, initial equity $\$100,000$, proportional fees and symmetric slippage:

$$\text{DURATION\_DAYS} = 365,\ m_{\text{bar}} = \text{TF\_MIN} = 5\,\text{minutes},\ W_{0} = \text{INITIAL\_CASH} = \$100,000,$$

$$f = \text{FEE\_BPS} = 0.10\%\,(10\text{ bps per notional}),\ \varepsilon = \text{SLIPPAGE\_BPS} = 0.02\%\,(2\text{ bps per side}).$$

Data Guard (crypto-friendly).
Gap and staleness tolerances, spread/jump caps:

$$\text{LOOKBACK\_MIN} = 120,\ \text{GAP\_TOL\_MIN} = 10,\ \text{GAP\_RECENT\_MIN} = 60,\ \text{GAP\_PCT\_WARN} = 50,\ \text{GAP\_PC}$$

$$\text{TRADE\_STALE\_SEC} = 1800,\ \text{BARS\_FRESH\_SEC} = 180,\ \text{SPREAD\_BPS\_CAP} = 50,\ \text{JUMP\_WARN\_BPS} = 30$$

$$\text{ONE\_SHOT} = 0,\ \text{STRICT} = 0.$$

Order Listener.

$$\text{BACKFILL\_HOURS} = 24,\ \text{SAFETY\_OVERLAP\_SEC} = 300,\ \text{SINK\_JSONL} = 1,\ \text{SINK\_CSV} = 0,\ \text{COST\_RATE}$$

Risk Worker.
One-day, one-sided parametric VaR with enforcement and drawdown cap:

$$\text{VAR\_WL} = 1440\,\text{minutes},\ \text{VAR\_CONF} = 0.95,\ \text{MAX\_VAR\_PCT} = 0.20,\ \text{MAX\_EXPOSURE\_USD} = 0,$$

$$\text{MAX\_DRAWDOWN\_PCT} = 0.30,\ \text{ENFORCE} = 1,\ \text{CANCEL\_OPEN\_BEFORE\_FLATTEN} = 1,\ \text{VAR\_REFRES}$$

Database.

$$\text{DB\_DSN} = \text{postgresql://postgres:postgres@db:5432/trading}.$$

Notes.
If a trading endpoint requires an unslashed symbol (e.g., $BTCUSD$), override SYMBOL per service in the composition while keeping data services on BTC/USD. All **data indices and timestamps are UTC** (right-closed/right-labeled). Secrets remain in environment and are not printed in this document.

B. Reproducibility & Environment

This section pins the runtime, determinism settings, and step-by-step commands to reproduce results and figures. Let the bar size be m minutes; annualization uses $N_{yr}=525,600/m$.

B.1 Environment (containerized). Python 3.11 (CPython), CPU-only. Container OS: Debian slim; host: Docker ≥ 24 on Linux/macOS/Windows. Core libs: `numpy`, `pandas`, `scipy`, `matplotlib` (Agg), `alpaca-py`, `SQLAlchemy`, `psycopg2-binary`, `tzdata`; database: PostgreSQL 16 (+ TimescaleDB). Time zone: `UTC`; bars are right-closed/right-labeled.

B.2 How to run (Docker Compose).

```
# Build images
docker compose build

# Start DB (and optionally initialize schema)
docker compose up -d db
# docker compose exec -T db psql -U postgres -d trading -f
docker/sql/000_init.sql

# Backtest (honors STRATEGY and backtest vars in .env)
docker compose run --rm backtest

# Pseudo-live (paper by default unless ALPACA_PAPER=0)
docker compose up -d data_guard risk_worker order_listener live_trader
dashboard

# Logs
docker compose logs -f backtest
docker compose logs -f data_guard
docker compose logs -f live_trader
docker compose logs -f risk_worker
```

B.3 — Using the Dashboard (Reproduce Figures)

Export the §6 figures (trend and z-score equity/drawdown) directly from the dashboard in `app/dashboard/Home.py`. Let the bar size be m minutes; annotate captions with (f,ε) and timezone.

1) Launch services

```
# From project root
docker compose build
docker compose up -d db dashboard          # start DB + dashboard UI
```

2) Open the UI - Visit http://localhost:8501 (or the port mapped in docker-compose.yml). - Use the Select backtest run dropdown at the top.

3) Choose run & verify parameters - Source: csv . - Symbol: BTC/USD. - Strategy: ema for Figure 6.1; zscore for Figure 6.2. - Confirm the Parameters panel matches your .env (e.g., EMA_FAST=13, EMA_SLOW=150, FEE_BPS=0.10, SLIPPAGE_BPS=0.02, bar size $m=5$).

4) Export figures

- Scroll to the equity chart with buy-and-hold overlay.
- Use the chart's download button (camera/save icon) to export images.
- Save with consistent names:
- report/figs/06_results/trend_equity_dd.png
- report/figs/06_results/rev_equity_dd.png
- If a separate drawdown panel exists, export it too (e.g., *_dd_only.png).

5) Re-run with on-screen params (optional) - Click Re-run with these params to enqueue a deterministic backtest. - Monitor progress: `bash   docker compose logs -f backtest`

```
-    When the new run appears in the dropdown, repeat Export figures.
```

C. Code & Data Inventory

C.1 File layout & entrypoints. ```python app/ dashboard/ Home.py # dashboard UI (reports/inspection) part1_backtest.py # entrypoint: backtest engine part2_pseudo_live.py # entrypoint: pseudo-live trader (paper by default) part3_data_guard.py # entrypoint: data-quality sentry part3_order_listener.py # entrypoint: websocket order/fill listener part3_risk_worker.py # entrypoint: risk & flatten logic docker/ sql/ 000_init.sql # schema/migrations for PostgreSQL + TimescaleDB docker-compose.yml # service orchestration (db, backtest, live stack) Dockerfile # container image (Python 3.11-slim + deps) out/ # run artifacts (logs, CSVs, figures) requirements.txt # pinned Python dependencies

C.2 Entrypoints.

Backtest:`docker compose run --rm backtest`(or python app/part1_backtest.py).

Trader: `docker compose up -d live_trader` (depends on data_guard, order_listener, risk_worker).

Dashboard: `docker compose up -d dashboard` (serves UI for runs).

Database: `docker compose up -d db`(schema via docker/sql/000_init.sql).