

K. S. Sushanth

+91 63791 19460 — k.s.sushanth06@gmail.com — linkedin.com/in/sushanthks — github.com/SushanthKS06

CORE SKILLS

Languages: Python, C++, Bash

Quantitative: Probability, Statistics, Linear Algebra, Optimization

Markets: Limit Order Books, Market Microstructure, Liquidity, Slippage, Transaction Costs

Systems: Event-Driven Architecture, Latency Profiling (p50/p95/p99), Cache-Aware Memory Design

Machine Learning: Linear Models, Tree-Based Models, Regularization, Time-Series Cross-Validation, Regime Detection

Data Tools: Time-Series Analysis, Feature Engineering, NumPy, Pandas, Matplotlib, Linux/Unix, Git

Backtesting: Execution-Aware Backtesting, Transaction Cost Modeling

PROJECTS

Deterministic Exchange Simulator (Limit Order Book)

Project Developer

Market Microstructure

- Built a fully deterministic, event-driven exchange simulator modeling limit and market orders, partial fills, cancellations, and strict price-time priority.
- Implemented replayable simulations using fixed seeds and ordered event queues, guaranteeing identical outcomes across runs.
- Reduced median per-event latency by 38% (4.1 ms to 2.5 ms) on a single-threaded pipeline by eliminating per-event object creation, improving cache locality, and batching events.
- Identified simulator failure modes under liquidity stress including queue starvation and spread collapse, and calibrated matching logic to preserve realistic execution behavior.

Execution-Aware Statistical Arbitrage Strategy

Project Developer

Market Microstructure

- Designed a mean-reversion strategy using volatility-normalized z-score signals across multiple liquid instruments.
- Explicitly modeled transaction costs, order book impact, and execution delays, eliminating optimistic backtest bias.
- Stress-tested signals across rolling windows and randomized regimes; rejected two statistically significant but non-stationary signals despite positive historical performance.
- Evaluated behavior across five volatility regimes, prioritizing drawdown control and tail stability over peak returns.

Low-Latency Market Data Processing Pipeline

Project Developer

Performance Optimization

- Built a real-time tick-data ingestion and aggregation pipeline sustaining approximately one million events per minute on a single-core, single-process setup under steady load.
- Reduced end-to-end processing latency by approximately 32% (p50) by minimizing allocations, using fixed-size rolling buffers, and simplifying parsing paths.
- Measured latency distributions (p50 and p95) and identified Python object creation as the dominant bottleneck under bursty conditions.
- Designed modular components reusable for live strategy prototyping and offline research, with explicit migration paths to C++ for lower-latency requirements.

Machine Learning-Based Signal Filtering & Regime Detection

Project Developer

Statistics

- Applied regularized linear models and tree-based models to filter noisy trading signals rather than predict prices directly.
- Engineered microstructure-aware features including volatility, spread dynamics, and order imbalance, validated using time-series cross-validation.
- Used ML outputs as confidence and regime filters, reducing false-positive trades by approximately 20% without increasing turnover.
- Explicitly avoided overfitting by rejecting unstable features and validating performance across multiple market regimes.

Low-Level Order Book Core

Project Developer

Performance-Oriented Systems

- Implemented a lightweight limit order book core in C++ with deterministic matching and explicit memory ownership.
- Used contiguous, cache-friendly data structures to minimize cache misses during high-frequency insert and cancel workloads.
- Benchmarked order insert and cancel operations, achieving sub-millisecond median latency with stable tail behavior on commodity hardware.
- Identified correctness bugs under edge-case ordering scenarios and fixed them by enforcing strict invariants and defensive checks.

EDUCATION

Sri Krishna College Of Engineering And Technology

Bachelor of Technology (B.Tech) in Information Technology; CGPA: 8.11/10

Coimbatore, India

Sep 2023 – May 2027 (Expected)

- Relevant Coursework: Probability, Stats, Algorithms, Linear Algebra, OS, Optimization