Kaihong Deng

Quantitative Developer & Researcher

Guangzhou, China (UTC+8) | kh.deng@foxmail.com | arXiv:2512.22476

Summary

- Seeking **Quant/Backtesting Engineer Intern (Remote)**; building an auditable backtesting + auto-tuning workflow for crypto perpetual futures.

- Focus: strict execution semantics (t→t+1), realistic trading costs (fees/slippage/funding), walk-forward evaluation, reproducible artifacts.

- Delivery: can implement scoped modules (indicator/cost model/slippage model/validator/report) with tests and reproducible commands.

- Availability: Full-time (40+ hrs/week), start immediately (UTC+8).

Highlights

- Strict backtesting with auditability: deterministic runs (seed), environment self-check signatures binding code+cost profile, and machine-readable artifacts.

- Realistic cost model per bar: fees + slippage + funding; multiple versioned cost profiles (e.g., conservative/high-leverage) for replay and sensitivity.

- WFE (walk-forward evaluation): sliding windows with typical defaults of 2000-bar windows and 500-bar steps; consolidated ranking outputs (CSV/Excel).

- Scale of search runs (from internal logs): 300–900 trials for alpha legs and 400–1200 trials for hedge legs; multi-process workers commonly set to 12.

- Documented performance in project notes: vectorized backtesting (2.6× speedup; ~3.2–3.7s per sample) and Optuna TPE search efficiency (3–5× vs. LHS/random).

- Market/data & validation integration: Binance Futures historical data; Gate.io perpetual testnet connectivity and live-style monitoring/report artifacts (no mainnet funds).

Skills

Languages	Python (3.10+), English (Technical reading/writing), SQL (basic), PowerShell (basic)
Backtesting	strict t→t+1 semantics, cost modeling, WFE/robustness gates, metrics/reporting
Optimization	Optuna (TPE), Bayesian search, grid/random baselines
Data	pandas, NumPy, CSV/Parquet, time-series alignment, reproducibility controls
Quality	Git, pytest, type hints, ruff, pre-commit/CI gates, artifact/version traceability

Selected Project

Aochuang (Proprietary) — Crypto Perpetual Futures Backtesting & Auto-Tuning System *2025–2026*
Python; strict execution engine + realistic costs + WFE + audit gates + Optuna

- Implemented a strict execution engine enforcing t→t+1 semantics to reduce look-ahead bias; supports multi-timeframe signal pipelines.

- Integrated real-world cost modeling (fee, slippage, funding) and produced a cost breakdown for audits and sensitivity checks.

- Added walk-forward evaluation (sliding windows) and robustness gates to reduce *best-of-N* overfitting risk.

- Automated parameter search with Optuna (TPE) and generated structured reports (Excel/CSV) for candidate ranking and validation.

- Designed reproducibility controls: self-check signatures bind code + cost profile; CI gates catch semantic drift (schema + no-lookahead tests).

- Built testnet-oriented validation and monitoring/report artifacts to evaluate execution constraints beyond pure backtests.

Experience

Independent Research Engineer (Quant)   *Remote*
End-to-end ownership (design → implementation → verification → audit)

- Built and iterated a research workflow that prioritizes correctness and reproducibility before scaling experiments.

- Maintained an append-only decision log and traceable artifacts for each experiment run.

Education

Guangzhou College of Commerce (GCC)   *2023–2027 (expected)*
B.S. Candidate in Artificial Intelligence

Work Samples

- Preprint: AutoQuant: An Auditable Expert-System Framework for Execution-Constrained Auto-Tuning in Cryptocurrency Perpetual Futures (arXiv:2512.22476).

- Technical documentation, sample reports (Excel/CSV), and code walkthroughs can be provided upon request.