Evolving Adaptive Market Making Strategies with Compressed Input Representations and CMA-ES

ONAT YARICILAR · CS + FINANCE · ADVISED BY PROF. JIM O'CONNOR

MOTIVATION

Market makers post buy and sell limit orders continuously to provide liquidity. Designing a robust policy is hard: the agent must balance profitability against inventory risk across diverse market conditions. This project asks whether **evolutionary optimization** can discover such policies automatically, using a compact numerical representation of the limit order book as input.

APPROACH

The experiment runs inside **ABIDES**, a high-fidelity agent-based market simulator. ABIDES populates a continuous double auction with value, noise, momentum, and execution agents, fed with real **Nordic limit-order-book data**.

The market-making agent is parameterized by an **8-dimensional genome** controlling participation rate, quote sizing, spread smoothing, cancel delay, inventory aversion, and max inventory. **CMA-ES** searches this space by maintaining a multivariate Gaussian over genomes, updating each generation to favor higher-scoring policies.

FITNESS & ROBUSTNESS

Each evaluation scores a genome by log-return minus inventory and drawdown penalties. The training objective is **CVaR of the worst 25% of training scenarios**, so a genome must survive adverse conditions to rank well. Datasets split 60/20/20 (train/val/test); the best genome replays on held-out sets post-optimization.

RESULTS

Across 12 generations and 144 candidate evaluations, CMA-ES converged to a genome with **positive mean PnL** on train and validation, controlled inventory (within the 3x cap), and a stable Sharpe trajectory. The best genome maintained profitable spreads under high-noise and momentum-driven regimes that eliminated naive baselines.

WHAT I BUILT

End-to-end in Python: CMA-ES harness, parallelized evaluation runner, terminal TUI streaming per-worker progress, optional DearPyGui dashboard, and an auto-generated Markdown report with train/val statistics and convergence plots. ABIDES forked and patched for modern Pandas compatibility.

GENOME (8 DIMS)

Parameter	Range	Role
pov	0.5-35%	Participation rate
min_order_size	50-500	Quote size (shares)
spread_alpha	0.01-0.30	Spread smoothing
cancel_delay	10-200 ms	Order lifetime
inv_aversion	0.01-0.50	Inventory penalty
inv_limit	500-5000	Max inventory cap
skew	-0.5 to 0.5	Bid/ask asymmetry
ladder_spacing	>=6 ticks	Quote ladder step

BEST GENOME

```
pov: 1.5%
min_order_size: 391
spread_alpha: 0.10
cancel_delay: 92 ms
inv_aversion: 0.10
inv_limit: 3,128
```

EXPERIMENT SETUP

Population	12 genomes
Generations	10 + 2 warmup
Total evals	144
Train seeds	1-6, 20200603
Val seeds	7-8, 20200603
Test seeds	9-10, 20200604
Objective	CVaR (worst 25%)
Val blend	0.25 weight
Loss penalty	10.0x

STACK

Python · NumPy · Pandas · CMA · ABIDES · Matplotlib · DearPyGui