Optimizing Directional Change Trading Strategies with Adaptive Thresholds for the Brazilian Stock Market

Lucas E. Mieri[1(✉)] and Sanderson L. Gonzaga de Oliveira[2]

[1] Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos, Brazil
lucasmieri@ita.br
[2] Universidade Federal de São Paulo (UNIFESP), São Paulo, Brazil
sanderson.oliveira@unifesp.br

Abstract. Financial time series analysis has traditionally relied on physical time, where analysts sample prices at fixed intervals. The Directional Change (DC) paradigm replaces this with an event-based representation driven by price magnitude rather than elapsed time. We extend the Multi-Strategy/Threshold Genetic Algorithm Model, which combines eight DC-based trading strategies across multiple thresholds into an ensemble of 70 sub-strategies optimized by a Genetic Algorithm (GA), by introducing volatility-adaptive thresholds that scale to each asset, walk-forward validation with per-fold recalibration, and the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) as an alternative optimizer. This study evaluates the framework on 30 equities from the Brazilian stock exchange across 270 out-of-sample folds spanning from 2017 to 2025. The CMA-ES ensemble achieves a higher Sharpe Ratio than Buy-and-Hold (0.87 vs. 0.64) and nearly doubles its profit factor, while reducing volatility in every fold tested (mean ratio 0.68). CMA-ES and GA achieve comparable out-of-sample performance, but CMA-ES generalizes better, with half the train-to-test Sharpe degradation.

Keywords: Directional Changes · Genetic Algorithm · CMA-ES · Trading Strategies · Brazilian Stock Market · Walk-Forward Validation

1 Introduction

Analysts have largely confined financial time series analysis to physical time, sampling prices at fixed intervals such as daily closes or minute-by-minute snapshots [1,2]. This convention is practical but lossy: price movements that fall between two consecutive sampling points go unrecorded, and can miss significant events [3]. The Directional Change (DC) paradigm offers an event-based alternative. Introduced by Guillaume et al. [1] and later formalized through the scaling laws of Glattfelder et al. [4], DC decomposes a price series into alternating upward and downward movements defined by a threshold θ. A reversal of at least θ from the last extreme confirms a DC event; the continuation beyond that

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026 Workshops, LNCS 16760, pp. 199–214, 2027.
https://doi.org/10.1007/978-3-032-30521-3_13

confirmation point until the next reversal is called the Overshoot (OS). Because events are triggered by price magnitude rather than elapsed time, the resulting representation captures only movements that surpass the threshold, filtering out small fluctuations that may not carry directional information [3].

Salman et al. [5] proposed the Multi-Strategy/Threshold Genetic Algorithm Model (MSTGAM), which combines eight DC-based trading strategies across ten thresholds into an ensemble of 70 sub-strategies with weights optimized by a Genetic Algorithm (GA). Each strategy exploits a distinct statistical regularity of large price movements, so that the ensemble extracts predictive information from the events that the DC paradigm isolates as significant. The model reported superior risk-adjusted returns over both individual DC strategies and Technical Analysis (TA) benchmarks on the New York Stock Exchange (NYSE). However, the authors designed the framework for NYSE data, and several choices limit its applicability to other markets. The thresholds (0.098% to 2.55%) are hard-coded for developed-market volatility. Evaluation relies on a single 80/20 split, with little information about robustness across time. The model fixes the risk-free rate at 2.5%, far below the Brazilian Interbank Deposit Certificate (CDI), the standard risk-free benchmark in Brazil, which has ranged from 2% to 14% over the past decade. Strategy parameters are calibrated once on the full training set, introducing the risk of look-ahead bias; thus, any strategy with genuine predictive capacity should exhibit low train-to-test Sharpe degradation, defined as the difference between Sharpe ratios on the training and test sets. The GA itself requires 50 independent runs per configuration [6] to compensate for its stochastic variability, and CMA-ES [7], widely regarded as state-of-the-art for continuous black-box optimization [8,9], has not been applied to DC ensemble weights.

Three open problems motivate the present work. First, all existing DC trading studies employ fixed thresholds derived manually or from a predefined grid; no prior work adapts thresholds to asset-specific volatility, limiting cross-market applicability. Second, walk-forward evaluation with per-fold recalibration remains uncommon in DC trading studies, where single train-test splits are the prevailing practice. Third, analysts have not applied CMA-ES [7,8] to DC strategy weights, with only isolated uses in computational finance [19].

Notably, analysts have studied all existing DC trading in markets from advanced economies (e.g., FX pairs, NYSE, London Stock Exchange), as well as the Indian stock market. The Brazilian market, although being the largest in Latin America, has not been explored with MSTGAM or any other DC-based trading framework.

This work extends the MSTGAM framework with the following contributions:

1. **Volatility-adaptive threshold calibration**: the model scales DC thresholds to each asset's realized daily volatility; consequently, the same framework applies across assets without manual tuning.

2. **Walk-forward validation with per-fold calibration**: each strategy parameter is recalibrated at each fold using only training data, preventing any future information from entering the optimization.
3. **CDI-adjusted evaluation**: each performance metric employs the daily CDI as the risk-free rate, providing a fair comparison in the Brazilian context.
4. **CMA-ES as an ensemble optimizer**, replacing the multi-run GA with a single optimization per fold.
5. **First application of the DC paradigm to the Brazilian stock exchange (B3)**: this study evaluates the framework on 30 equities spanning multiple sectors over the period from 2015 to 2025.

Based on these extensions, we investigate four hypotheses:

H1. The MSTGAM ensemble with volatility-adaptive thresholds achieves a higher Sharpe Ratio [10] than Buy-and-Hold on B3 equities under out-of-sample walk-forward evaluation.
H2. The DC-based ensemble systematically reduces portfolio volatility and maximum drawdown (MDD) relative to the Buy-and-Hold strategy, operating as a risk filter (a strategy that primarily reduces volatility and drawdown) rather than an alpha generator (a strategy that primarily increases returns above the benchmark).
H3. The advantage of the ensemble over Buy-and-Hold increases with the size of the training window, as exposure to multiple market regimes improves strategy calibration.
H4. CMA-ES exhibits lower train-to-test Sharpe degradation than the GA, indicating better generalization to unseen data.

Section 2 reviews related work on DC-based trading and metaheuristic optimization. Section 3 presents the DC detection algorithm and the eight trading strategies that form the base of the ensemble. Section 4 describes the proposed extensions: volatility-adaptive thresholds, the ensemble optimization with CMA-ES and GA, and the fitness function. Section 5 details the experimental methodology. Section 6 presents and discusses the results. Section 7 concludes the paper.

2 Related Work

The DC paradigm originates from the work of Guillaume et al. [1], who proposed an event-driven approach to summarize price movements in foreign exchange (FX) markets. Glattfelder et al. [4] subsequently identified 12 empirical scaling laws from high-frequency FX data using this framework. Aloud et al. [3] formalized the detection algorithm that decomposes a price series into alternating DC and OS events, given a threshold θ, and Tsang et al. [2] extended the application of DC indicators to equity markets. Tao [11] developed additional DC-based indicators to extract information from financial data. Salman et al. [12] introduced the optimization of DC-based trading strategies using GA, combining seven rule-based strategies on a single threshold into the Multi-Strategy GA Model

(MSGAM). In a subsequent work, Salman et al. [13] extended the approach by applying GA optimization over multiple thresholds for each strategy individually, known as the Multi-Threshold GA Model (MTGAM). The full integration of all eight strategies and thresholds into a single ensemble of 70 sub-strategies, the MSTGAM, was presented in Salman et al. [5], reporting improved Sharpe Ratios over individual strategies and TA benchmarks on 200 NYSE stocks. In short, MSGAM combines multiple strategies on a single threshold, MTGAM optimizes individual strategies across multiple thresholds, and MSTGAM unifies both dimensions into a single ensemble of strategy-threshold pairs.

Beyond GA-based approaches, analysts have applied other metaheuristics to DC trading. Long et al. [14] compared Genetic Programming (GP) under both physical time and DC frameworks, finding that DC-based GP strategies outperform their physical-time counterparts on FX data. Long et al. [15] proposed a multi-objective GP approach that integrates DC signals with a modified Sharpe Ratio as fitness function, using the Non-dominated Sorting Genetic Algorithm II (NSGA-II) to address the single-objective limitation of earlier models. Rayment and Kampouridis [16] applied Deep Reinforcement Learning (DRL) using Proximal Policy Optimization to high-frequency FX trading within a DC sampling framework, and later extended the approach with position-sensitive mechanisms to enhance state representation [17]. Outside the DC paradigm, Chinraj and Anthonysamy [18] combined a GA with the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) in a rule-based moving average strategy to optimize the Indian stock market, using walk-forward validation.

3 Background: DC Paradigm and Trading Strategies

The ensemble relies on two building blocks: a DC detection algorithm and eight trading strategies. We first present the DC detection algorithm in Sect. 3.1 and then summarize the eight trading strategies that serve as the base components in Sect. 3.2.

3.1 Directional Change Detection

As outlined in Sect. 2, the DC paradigm replaces fixed-interval sampling with an event-driven representation in which the price series is dissected solely by the magnitude of price reversals. Given an adjusted close price series $\{p_t\}_{t=1}^{N}$ and a threshold $\theta > 0$, the detection algorithm (Algorithm 1) maintains a running extreme price and the current trend mode. A downturn (DT) is confirmed when the price falls by at least θ from the running high, and an upturn (UT) when it rises by at least θ from the running low. The interval between the extreme point and its confirmation defines the DC event; the continuation beyond the DC Confirmation (DCC) point until the next reversal defines the OS event. Two empirical scaling laws emerge from this decomposition [4]: the average OS price change approximates the threshold ($\langle \Delta p_{OS} \rangle \approx \theta$), reflecting the symmetry of price movements around threshold-defined reversals; and the average OS

duration approximates twice the DC duration ($\langle T_{OS} \rangle \approx 2\langle T_{DC} \rangle$), reflecting that markets typically take longer to exhaust an established trend than to confirm its reversal.

Algorithm 1: Directional Change detection (adapted from Aloud et al. [3]).

Input: prices $\{p_t\}_{t=1}^{N}$, threshold $\theta > 0$, initial mode $\in \{\text{UT}, \text{DT}\}$
Output: list of DC events, per-bar indicators

$p_{ext} \leftarrow p_1$; $mode \leftarrow$ initial mode;
for $t = 2, \ldots, N$ **do**
  if $mode = UT$ **then**
    if $p_t > p_{ext}$ **then**
      $p_{ext} \leftarrow p_t$;
    else if $(p_{ext} - p_t)/p_{ext} \geq \theta$ **then**
      record DT event;
      $mode \leftarrow$ DT;
      $p_{ext} \leftarrow p_t$;
    end if
  else if $mode = DT$ **then**
    if $p_t < p_{ext}$ **then**
      $p_{ext} \leftarrow p_t$;
    else if $(p_t - p_{ext})/p_{ext} \geq \theta$ **then**
      record UT event;
      $mode \leftarrow$ UT;
      $p_{ext} \leftarrow p_t$;
    end if
  end if
  Compute per-bar indicators: OSV_{cur}, TMV_{cur}, trend, durations;
end for

The algorithm starts by setting the extreme price p_{ext} to the first observation and choosing an initial trend mode. Then iterates over each subsequent bar. In upturn mode, if the current price p_t exceeds p_{ext}, the algorithm updates the extreme upward, tracking the running high. If, instead, the price drops by at least θ relative to p_{ext}, the algorithm confirms a downturn event, switches the mode to DT, and resets p_{ext} to p_t. Downturn mode mirrors this logic: the algorithm tracks the running low and confirms an upturn when the price rises by at least θ from it, switching back to UT and resetting p_{ext}. At the end of each iteration, the algorithm computes the per-bar indicators OSV_{cur}, TMV_{cur}, trend direction, and event durations that the trading strategies consume downstream.

At each bar t, the algorithm computes two indicators used by the trading strategies. The Overshoot Value (OSV), defined as $OSV_{cur} = (p_t - p^*_{DCC})/(\theta \cdot p^*_{DCC})$, measures the current price displacement p_t from the theoretical confirmation point p^*_{DCC}. The Total Move Value (TMV), defined as $TMV_{cur} =$

$(p_t - p_{ext})/(\theta \cdot p_{ext})$, measures the displacement from the last extreme point p_{ext}, encompassing the entire move from that pivot [6]. Additional per-bar state includes the trend direction, the number of bars since the last confirmation, and the duration of the last concluded DC event. These arrays, computed in a single sequential pass over the price series, form the input to all subsequent modules.

3.2 Trading Strategies

Salman et al. [12] assembled eight trading strategies, drawn from DC scaling laws, DC indicators, and TA-inspired saturation patterns, into an ensemble framework. The first group (scaling-law strategies) triggers positions based on empirical DC regularities: one monitors total price displacement relative to the threshold, and the other monitors the duration of the current OS event relative to its preceding DC event. The second group comprises four strategies based on DC indicators and statistical ratios: two compare current values of OSV and TMV against calibrated thresholds, one compares the OS-to-DC duration ratio against a training-set average, and one uses the historical ratio of OS-to-DC event counts as a probabilistic entry filter. The third group detects saturation patterns, defined as sequences of consecutive trends with or without overshoot, analogous to ascending and descending triangles in TA. Table 1 summarizes the entry and exit conditions. Each strategy independently emits a Buy, Hold, or Sell recommendation at every bar, and the model does not permit short selling [6]. Individually, these rules are too simple to sustain consistent returns. The ensemble gains its value from combining its signals: agreement produces conviction, disagreement keeps the portfolio in cash.

Table 1. Entry and exit conditions for the eight DC-based trading strategies (adapted from Salman et al. [12]).

Strategy	Type	Buy Signal (in DT)	Sell Signal
1	Scaling law	$\lvert TMV_{cur}\rvert \geq 2.0$	Same in UT
2	Scaling law	$T_{OS} \geq 2 \cdot T_{DC}$	Same in UT
3	Indicator	$\lvert OSV_{cur}\rvert \geq \lvert OSV_{best}^{DT}\rvert$	$\lvert OSV_{cur}\rvert \geq \lvert OSV_{best}^{UT}\rvert$
4	Indicator	$\lvert TMV_{cur}\rvert \geq \lvert TMV_{best}^{DT}\rvert$	$\lvert TMV_{cur}\rvert \geq \lvert TMV_{best}^{UT}\rvert$
5	Ratio	$T_{OS}/T_{DC} \geq RD$	Same in UT
6	Probabilistic	$r \geq RN$ at DCC	Next DCC in UT
7	Saturation	3 consec. UT with OS	Next DCC in DT
8	Saturation	3 consec. DT with OS	Next DCC in UT

In conceptual terms, Strategy 1 assumes that a trend has exhausted its average magnitude, positioning for a reversal; Strategy 2 applies the same logic to duration rather than price. Strategies 3 and 4 learn from training data the overshoot and total-move levels historically preceded reversals, and trigger when the

current market reaches those levels. Strategy 5 indicates trend exhaustion when the OS phase has lasted disproportionately longer relative to its DC phase. Strategy 6 acts as a probabilistic filter, entering positions more often when overshoots are historically frequent. Strategies 7 and 8 detect repeated failed breakouts in one direction.

4 Adaptive Multi-threshold DC Ensemble for the Brazilian Market

Applying DC-based strategies across assets with different volatility profiles requires thresholds that scale with each asset rather than remaining fixed (Sect. 4.1). Once multiple thresholds produce 70 independent signals, a mechanism is needed to aggregate them into a single trading decision (Sect. 4.2). Finally, the fitness function must reflect the cost and interest-rate environment of the target market (Sect. 4.3).

4.1 Volatility-Adaptive Thresholds

The fixed thresholds used by Salman et al. [5] (0.098% to 2.55%) do not transfer directly to markets with different volatility profiles. For example, BOVA11 has a daily volatility of 1.5% while PETR4 has 2.9% (over the 2015–2025 sample). Applying the same thresholds to both produces excessive event density for the volatile asset and insufficient events for the stable one. Inspired by the logarithmic spacing employed by Glattfelder et al. [4] in their scaling law analysis, we calibrate thresholds proportionally to each asset's realized daily volatility $\hat{\sigma}$, spacing them logarithmically between $f_{\min} \cdot \hat{\sigma}$ and $f_{\max} \cdot \hat{\sigma}$:

$$\theta_i = \exp\left(\ln(f_{\min}\hat{\sigma}) + \frac{i-1}{9}\Big[\ln(f_{\max}\hat{\sigma}) - \ln(f_{\min}\hat{\sigma})\Big]\right), \quad i = 1, \ldots, 10, \tag{1}$$

with $f_{\min} = 0.3$ and $f_{\max} = 5.0$. The logarithmic spacing samples low thresholds more finely, where DC events are most frequent. For BOVA11 ($\hat{\sigma} = 1.49\%$), the resulting range is 0.45% to 7.47%; for PETR4 ($\hat{\sigma} = 2.87\%$), 0.86% to 14.3%. The two saturation-pattern strategies are restricted to the five smallest thresholds, as they generate negligible trades at larger values with daily data [6]. The total number of sub-strategies is therefore $6 \times 10 + 2 \times 5 = 70$.

4.2 Ensemble Aggregation

At each bar t, the 70 sub-strategies independently emit a recommendation $r_j \in \{\text{Buy}, \text{Hold}, \text{Sell}\}$. The ensemble aggregates these into a single decision via stateful weighted voting. Given a weight vector $\mathbf{w} = (w_1, \ldots, w_{70})$ with $w_j \geq 0$ and $\sum_j w_j = 1$, the weighted score for each action is

$$S(a) = \sum_{j=1}^{70} w_j \cdot \mathbf{1}[r_j = a], \quad a \in \{\text{Buy}, \text{Hold}, \text{Sell}\}.$$

Following Salman et al. [5], the approach applies a hold-suppression rule: if at least three sub-strategies emit a non-hold recommendation, the approach sets $S(\text{Hold})$ to zero. The ensemble then updates its position only when Buy or Sell achieves the highest score strictly; if neither dominates, the approach maintains the current position. This inertia mechanism avoids unnecessary round-trips when the ensemble signal is ambiguous.

The weight vector $\mathbf{w}$ is optimized using two metaheuristic approaches, a GA and CMA-ES, described in Sect. 5. Since CMA-ES operates in unconstrained $\mathbb{R}^{70}$, the approach must project the raw output $\mathbf{x}$ onto a valid weight vector. We apply L_1 normalization: $w_j = |x_j| / \sum_k |x_k|$, which guarantees $w_j \geq 0$ and $\sum_j w_j = 1$. For the GA, the approach renormalizes weights so that their sum equals 1 after each crossover and mutation operation.

4.3 Fitness Function

The fitness function evaluates a candidate weight vector by simulating the portfolio it produces over the training period. At each bar t, the portfolio is either fully invested in the asset or fully allocated to the risk-free rate. Transaction costs of 3 basis points per side are deducted at each position change, corresponding to the settlement and trading fees charged by the Brazilian stock exchange for retail equity transactions [20]; Salman et al. [6] report a transaction cost of 0.5 basis points per side for NYSE equities, and the B3 cost is approximately six times higher.

Since the approach executes the trading signal at bar t at the close, the portfolio return during that bar still reflects the previous allocation:

$$r_t^{port} = \begin{cases} r_t^{CDI} - c & \text{if entering long at } t \text{ (still in cash during bar)} \\ r_t^{asset} - c & \text{if exiting long at } t \text{ (still in asset during bar)} \\ r_t^{asset} & \text{if maintaining long position} \\ r_t^{CDI} & \text{if out of market} \end{cases}$$

where $c = 0.0003$ is the one-way transaction cost. The fitness is the annualized calendar-based Sharpe Ratio [10] computed over daily excess returns:

$$SR = \frac{\overline{r^{port} - r^{CDI}}}{\text{std}(r^{port} - r^{CDI})} \cdot \sqrt{252} \tag{2}$$

This approach differs from that of Salman et al. [6], who compute the Sharpe Ratio at the trade level. The calendar-based formulation accounts for the opportunity cost of out-of-market intervals, which the trade-level metric ignores. If the standard deviation of excess returns is close to zero, which occurs when the

weight vector produces no trades, and the portfolio remains entirely in CDI, the Sharpe Ratio is substituted by zero in the corresponding metric to avoid numerical instability (see Sect. 5.3 for details).

5 Methodology

This section describes the walk-forward validation scheme (Sect. 5.1), the data sources and optimizer configurations (Sect. 5.2), and the metrics applied to the out-of-sample results (Sect. 5.3).

5.1 Walk-Forward Validation

To validate the out-of-sample data, we adopt an expanding-window walk-forward protocol. For each fold, the training set comprises all available data from 2015 up to year $Y-1$, and the test set covers year Y. A minimum of two training years is required, yielding nine folds ($Y = 2017, \ldots, 2025$). The first fold trains on 2015–2016 ($\sim$494 bars) and tests on 2017 ($\sim$247 bars); the last fold trains on 2015–2024 ($\sim$2 481 bars) and tests on 2025 ($\sim$248 bars).

At each fold the following steps are executed: (*i*) DC thresholds are computed from the training-set volatility using Eq. 1; (*ii*) the indicator-based and ratio-based strategy parameters are calibrated exclusively on training data; (*iii*) the DC transformer and the trading strategies are applied to the full price series, producing a recommendation matrix $\mathbf{R} \in \{\text{Buy}, \text{Hold}, \text{Sell}\}^{n \times 70}$; (*iv*) GA and CMA-ES optimize the weight vector on the training portion of $\mathbf{R}$; and (*v*) the approach applies the optimized weights to the test portion of $\mathbf{R}$ without modification.

5.2 Data and Optimizer Configuration

We collected daily adjusted closing prices for 30 equities listed on B3, spanning sectors including banking, utilities, commodities, energy, retail, construction, and technology. Prices span January 2015 to December 2025 ($\sim$2 700 trading days per asset), and we obtained them using the `yfinance` Python package [21]. The framework runs on Python 3.11. As the risk-free rate, we use the daily Interbank Deposit Certificate (CDI), sourced from the Brazilian Central Bank time series system (series 12) [22]. Transaction costs are set at 3 basis points per side (0.03%), corresponding to the settlement and trading fees for retail equity swing trades [20].

The GA follows the configuration of Salman et al. [6] for direct comparability with the original MSTGAM baseline, with population size 150, 18 generations, two-point crossover with probability 0.95, tournament selection of size 2, and elitism of 1. The first 70 individuals are identity chromosomes, each assigning full weight to a single sub-strategy; the approach initialized the remaining 80 with random, normalized weights. CMA-ES operates in an unconstrained 70-dimensional space with initial step size $\sigma_0 = 0.3$, population size 50, and a maximum of 10 000 function evaluations. The population size exceeds Hansen's [8]

default of $4 + \lfloor 3 \ln 70 \rfloor = 16$ to increase exploration in the high-dimensional weight space.

Salman [6] executed the GA 50 times per configuration and retains the single best chromosome, effectively selecting the most favorable outcome across runs, as the walk-forward protocol already requires 270 independent optimizations across 30 tickers and 9 folds. Both optimizers use a fixed random seed per fold to ensure reproducibility. Each ticker completes in approximately 80 min on an Intel Core i7-10700 workstation (8 cores/16 threads running at 2.9–4.8 GHz) with 16 GB DDR4-2666 RAM, with individual folds ranging from 3 min (the shortest training window) to 15 min (the longest).

5.3 Evaluation Metrics

We evaluated each out-of-sample fold using four metrics. The primary metric is the annualized Sharpe Ratio (Eq. 2), which measures excess return over the CDI per unit of risk. Since the Brazilian risk-free rate has averaged approximately 10% per year over the sample period, positive Sharpe Ratios are less attainable than in lower-rate markets. The remaining metrics are total return, annualized volatility, and maximum drawdown (MDD). When the optimizer produces a weight vector that does not trigger trades, the portfolio remains completely in CDI throughout the fold. In this situation, the excess returns are near-zero with negligible variance, which can produce artificially high Sharpe Ratios. Empirical results from one GA fold showed that the absence of trades resulted in a Sharpe Ratio exceeding 200. To handle such cases, the Sharpe Ratio is substituted by zero in the corresponding metric whenever the standard deviation and the mean of excess returns are close to zero.

6 Results

This section presents the findings, organized by the four tested hypotheses: cross-asset performance (H1) in Sect. 6.1, risk reduction (H2) in Sect. 6.2, training window effect (H3) in Sect. 6.3, and optimizer generalization (H4) in Sect. 6.4.

6.1 Cross-Asset Performance (H1)

Table 2 summarizes the aggregate out-of-sample metrics. Both optimizers outperform Buy-and-Hold in risk-adjusted terms, with CMA-ES achieving a mean Sharpe Ratio of 0.868 compared to 0.637 for the passive benchmark. The profit factor nearly doubles that of Buy-and-Hold (8.47 vs. 4.43), indicating that the ensemble generates substantially larger gains in relation to its losses. The advantage holds across 26 of 30 tickers. Among the four in which Buy-and-Hold dominates (BBAS3, PETR4, SBSP3, SUZB3), three fall within commodity-linked sectors, where sustained directional trends over the sample period favor uninterrupted exposure. The per-ticker mean Sharpe Ratio for CMA-ES ranges from

Table 2. Aggregate out-of-sample metrics across 30 equities and 270 folds (2017–2025). Values in parentheses denote standard deviations across folds.

Metric	CMA-ES	GA	B&H
Mean Sharpe Ratio	0.87 (0.93)	0.85 (1.05)	0.64 (0.96)
Folds > B&H Sharpe	60%	59%	–
Tickers > B&H Sharpe	87%	83%	–
Profit factor ($\sum r^+ / \sum \lvert r^- \rvert$)	8.47	6.26	4.43
Vol. ratio ($\sigma_{ens}/\sigma_{B\&H}$)	0.68	0.64	1.00
Folds with lower MDD	95%	95%	–
MDD ratio ($MDD_{ens}/MDD_{B\&H}$)	0.68	0.65	1.00
Mean degradation ($SR_{train} - SR_{test}$)	0.54 (0.99)	1.08 (0.96)	–

0.12 (SUZB3) to 1.58 (VIVT3), with an interquartile range of 0.60–1.15, which suggests the aggregate result is robust to outliers.

Figure 1 illustrates this for BOVA11, the Ibovespa ETF. Over nine concatenated out-of-sample years, CMA-ES triples the initial capital while Buy-and-Hold roughly doubles it. The gap opens during the 2020 and 2024 corrections, where the ensemble moves to CDI and avoids the worst of the drawdown.

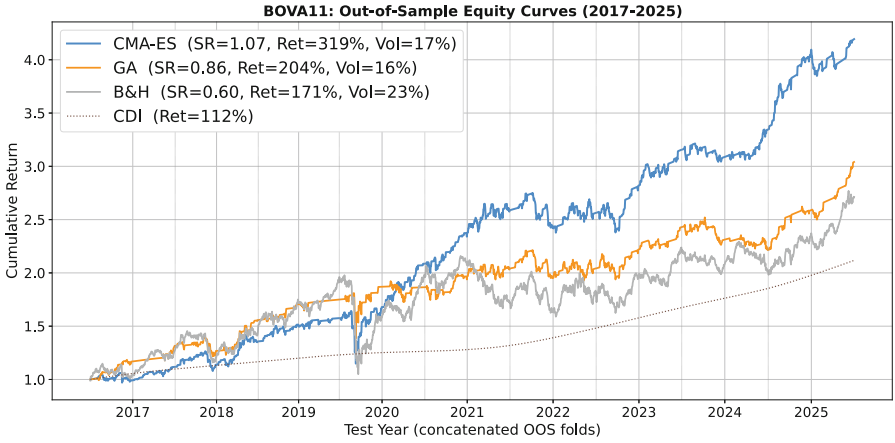


Fig. 1. BOVA11 out-of-sample equity curves (2017–2025), concatenated across nine folds. Legend reports cumulative Sharpe Ratio, return, and annualized volatility.

6.2 Risk Reduction (H2)

Figure 2 shows the annualized volatility of each fold under CMA-ES and GA against Buy-and-Hold. All folds lie below the diagonal for both optimizers: the

ensemble never increases volatility. The mean ratio is 0.68, a reduction of approximately 32%. Maximum drawdown follows the same pattern, with 95% of folds showing lower drawdowns under CMA-ES and GA.

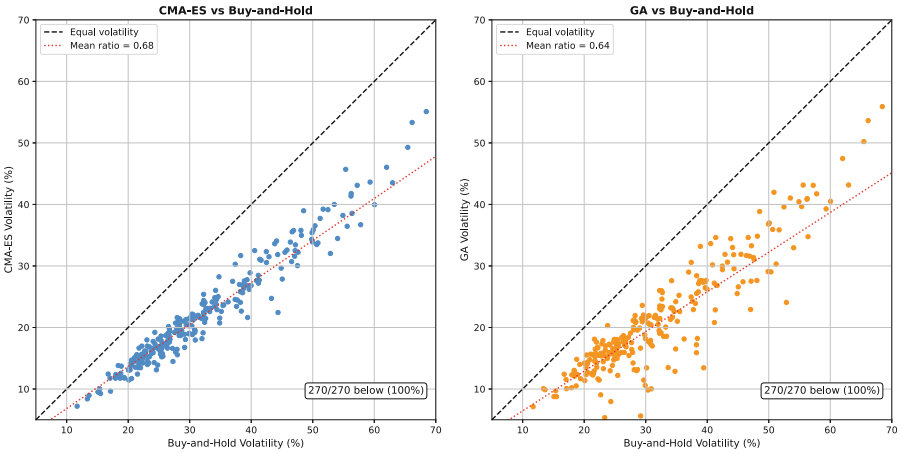


Fig. 2. Annualized volatility per fold: CMA-ES (left) and GA (right) versus Buy-and-Hold. All 270 points lie below the diagonal.

The mechanism behind this result is the ensemble's tendency to allocate the portfolio to CDI whenever the sub-strategies disagree. This filtering removes the asset's return during periods of ambiguous signal. Volatility reduction is therefore universal across tickers and folds.

6.3 Training Window Effect (H3)

Figure 3 shows the Sharpe difference between CMA-ES and Buy-and-Hold as a function of the training window length. Folds trained on shorter windows (2–5 years, $n = 120$) show a mean advantage of +0.10, while folds trained on longer windows (6–10 years, $n = 150$) show +0.33. The positive trend suggests that exposure to more market regimes, including the 2020 crisis, improves both strategy calibration and ensemble weight optimization.

6.4 Optimizer Generalization (H4)

CMA-ES and GA produce indistinguishable out-of-sample Sharpe Ratios (0.868 vs. 0.852), but differ in how they reach this result. Figure 4 shows the distribution of train-to-test Sharpe degradation for both methods. CMA-ES degrades by 0.54 on average (std = 0.99), while the GA degrades by 1.08 (std = 0.96), approximately twice as much. The GA achieves higher training Sharpe Ratios but fails to transfer this advantage to unseen data. While the GA explores the

search space through population-based selection and recombination without an explicit model of the underlying distribution, CMA-ES iteratively estimates and refines a full covariance matrix that parametrizes the search distribution itself. Thus, CMA-ES explores the 70-dimensional weight space more efficiently and converges to solutions that generalize better.

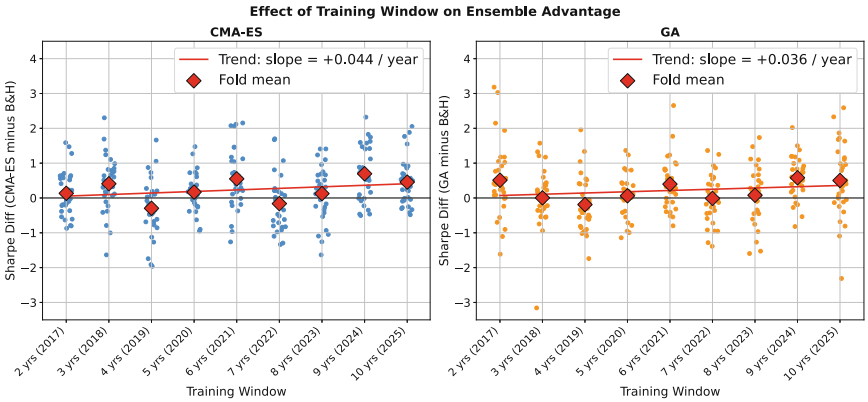


Fig. 3. Sharpe difference (CMA-ES minus B&H) versus training window length. Red diamonds indicate fold means; the red line is the linear trend. (Color figure online)

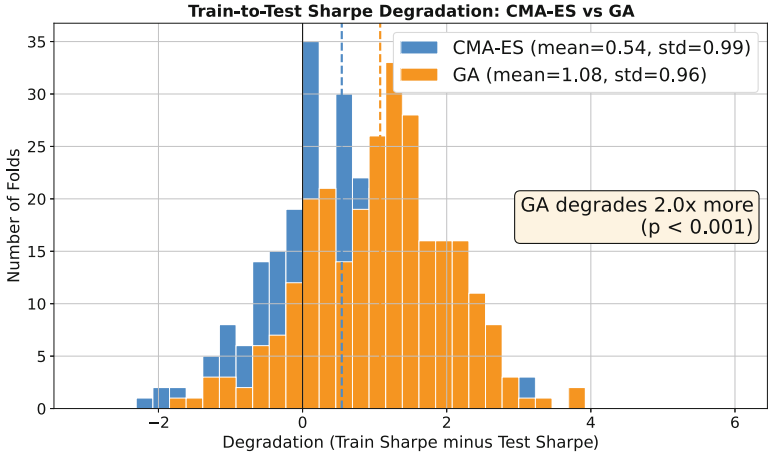


Fig. 4. Train-to-test Sharpe degradation across 270 folds. CMA-ES (blue) concentrates near zero; GA (orange) shifts toward higher degradation. (Color figure online)

7 Conclusion

We extended the MSTGAM framework for the Brazilian market by replacing its fixed thresholds with volatility-adaptive ones, introducing walk-forward validation, and adding CMA-ES as a second optimizer alongside the original GA. The results confirm the viability of DC-based trading in a high-interest-rate emerging market. The CMA-ES ensemble outperforms Buy-and-Hold in risk-adjusted terms (Sharpe 0.87 vs. 0.64, H1), with volatility reduced in every fold and drawdown in 95% of them (H2). Longer training windows strengthen this advantage (H3), suggesting the ensemble benefits from exposure to a wider range of market conditions. CMA-ES and GA deliver similar out-of-sample Sharpe Ratios, but CMA-ES degrades half as much from training to test (H4), making it the more reliable optimizer for this 70-dimensional weight space.

The most relevant limitation is the annual rebalancing cycle. Markets shift regimes within a calendar year, and the current protocol cannot react until the next fold boundary. A related issue is that the ensemble relies exclusively on DC-derived signals: it has no direct representation of the prevailing market regime and cannot reliably discern whether the market is trending or range-bound. Incorporating strategies less correlated with DC, such as classical TA indicators, could diversify the signal base and improve robustness during regimes where DC events are sparse.

On the deployment side, the framework offers structural advantages that simplify operation. Daily data refresh and per-fold recalibration are computationally inexpensive, allowing the model to incorporate new observations and retrain quickly between trading sessions. The modular structure of the ensemble accommodates new strategies with minimal changes: any rule that emits Buy, Hold, or Sell signals can be added as a sub-strategy, enabling the optimizer to assign its weight. These properties make the framework suitable for real-time operation on equities, where end-of-day decisions can be computed and queued for the closing auction.

The most promising direction for future work is regime-conditional weight optimization. We expect that certain DC strategies perform well in specific market conditions but poorly in others. Providing the optimizer with regime information as an additional input, rather than relying on a single static weight vector per fold, could allow the ensemble to alternate between strategy configurations as conditions change.

Acknowledgments. The second author acknowledges the Brazilian funding agency CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) for financial support with grant 402599/2023-3. The authors also thank the anonymous referees for their careful reading, insightful comments, and suggestions.

Disclosure of Interests. The authors declare that they have no competing interests.

References

1. Guillaume, D.M., Dacorogna, M.M., Davé, R.R., Müller, U.A., Olsen, R.B., Pictet, O.V.: From the bird's eye to the microscope: a survey of new stylized facts of the intra-daily foreign exchange markets. Finance Stochast. **1**(2), 95–129 (1997)
2. Tsang, E.P.K., Tao, R., Serguieva, A., Ma, S.: Profiling high-frequency equity price movements in directional changes. Quant. Finan. **17**(2), 217–229 (2017)
3. Aloud, M., Tsang, E.P.K., Olsen, R., Dupuis, A.: A directional-change event approach for studying financial time series. Economics **6**(1), 1–17 (2012)
4. Glattfelder, J.B., Dupuis, A., Olsen, R.B.: Patterns in high-frequency FX data: discovery of 12 empirical scaling laws. Quant. Finan. **11**(4), 599–614 (2011)
5. Salman, O., Melissourgos, T., Kampouridis, M.: A genetic algorithm for the optimization of multi-threshold trading strategies in the directional changes paradigm. Artif. Intell. Rev. **59**(1), 2 (2026)
6. Salman, O.: Trading Strategies Optimization by Genetic Algorithm Under the Directional Changes Paradigm. PhD Thesis, University of Essex (2024)
7. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evol. Comput. **9**(2), 159–195 (2001)
8. Hansen, N.: The CMA Evolution Strategy: A Tutorial. arXiv preprint arXiv:1604.00772 (2016)
9. Vermetten, D., de Nobel, J., Wang, H., Doerr, C., Bäck, T.: Comparison of high-dimensional Bayesian optimization algorithms on BBOB. ACM Trans. Evol. Learn. Optim. **4**(3), 1–31 (2024)
10. Sharpe, W.F.: Mutual fund performance. J. Bus. **39**(1), 119–138 (1966)
11. Tao, R.: Using Directional Change for Information Extraction in Financial Market Data. PhD Thesis, University of Essex (2018)
12. Salman, O., Kampouridis, M., Jarchi, D.: Trading strategies optimization by genetic algorithm under the directional changes paradigm. In: IEEE Congress on Evolutionary Computation (CEC), pp. 1–8 (2022)
13. Salman, O., Melissourgos, T., Kampouridis, M.: Optimization of trading strategies using a genetic algorithm under the directional changes paradigm with multiple thresholds. In: IEEE Congress on Evolutionary Computation (CEC), pp. 1–8 (2023)
14. Long, X., Kampouridis, M., Kanellopoulos, P.: An in-depth investigation of genetic programming under physical time and directional change frameworks for algorithmic trading. IEEE Access **13**, 151001–151015 (2025)
15. Long, X., Kampouridis, M., Papastylianou, T.: Multi-objective genetic programming-based algorithmic trading, using directional changes and a modified Sharpe ratio score for identifying optimal trading strategies. Artif. Intell. Rev. **59**(2), 39 (2025)
16. Rayment, G., Kampouridis, M.: High frequency trading with deep reinforcement learning agents under a directional changes sampling framework. In: 2023 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 387–394 (2023)
17. Rayment, G., Kampouridis, M.: Enhancing high-frequency trading with deep reinforcement learning using advanced positional awareness under a directional changes paradigm. In: 2024 International Conference on Machine Learning and Applications (ICMLA), pp. 63–70 (2024)
18. Chinraj, K.F., Anthonysamy, S.S.: A hybrid framework integrating mathematical genetic algorithm and multi-criteria decision making for robust rule-based stock market strategy optimization. Math. Comput. Sci. **6**(4), 89–100 (2025)

19. Yu, K., Cheng, C.: Dynamic Portfolio Optimization Using Evolution Strategy. Stanford University, CS229 Final Report (2017)
20. B3 – Brasil, Bolsa, Balcão: Tarifas de ações e fundos de investimento – mercado à vista. Available at: https://www.b3.com.br/pt_br/produtos-e-servicos/tarifas/listados-a-vista-e-derivativos/renda-variavel/tarifas-de-acoes-e-fundos-de-investimento/a-vista/. Accessed March 2026
21. Aroussi, R.: yfinance: Download market data from Yahoo! Finance API. Python package. https://pypi.org/project/yfinance/ (2017)
22. Banco Central do Brasil: Sistema Gerenciador de Séries Temporais (SGS), série 12 – Taxa de depósito interfinanceiro (CDI). https://www3.bcb.gov.br/sgspub/ (accessed March 2026)