Evolutionary Optimization of an Intraday Strategy: A Genetic Algorithm Approach to the Opening Range Breakout in U.S. MidCap Markets

Jorge Moncada Gutiérrez[1]

[1]BSc in Economics and Finance, Universidad Autónoma de Madrid

May 2026

Abstract

This dissertation evaluates the portability and performance of an evolutionary Opening Range Breakout (ORB) strategy within the highly efficient U.S. equity market, specifically testing the CME E-mini S&P 400 MidCap (EMD) futures. Building upon the framework established by Wu et al. (2021), the study employs Genetic Algorithms (GA) coupled with a Walk-Forward Analysis over a ten-year historical period (2009–2018) to dynamically optimize entry thresholds and protective closing mechanisms. The empirical findings reveal a structural divergence from previous literature: whereas the reference model relies heavily on fixed stop-losses for stability, the GA applied to the U.S. MidCap market systematically rejects hard stops due to the high level of intraday noise. The results obtained demonstrate that algorithmic momentum strategies are not entirely transferable across markets and that their internal logic must undergo rigorous evolutionary adaptation, shifting from simple loss-cutting to dynamic profit conservation, in order to maintain a competitive edge in more liquid and efficient markets.

Keywords: Algorithmic Trading, Opening Range Breakout (ORB), Genetic Algorithms, Walk-Forward Analysis, Intraday Momentum, E-mini S&P 400.

Contents

1 Introduction **3**

2 Motivation **8**

3 Objetives **10**

3.1 General Objectives 10

3.2 Specific Objectives 10

3.2.1 Technical Design 10

3.2.2 Evolutionary Optimization 11

3.2.3 Comparative Evaluation 11

4 Conceptual Framework **12**

4.1 From Market Efficiency to Adaptative Markets 12

4.2 Genetic Algorithms 13

4.3 Microstructure and Intraday Momentum Anomalies 15

4.3.1 The Opening Range Phenomenom 15

4.3.2 Contraction-Expansion Principle and Intraday Mass Psychology . . 15

4.4 The Opening Range Breakout system propsed by Wu et al. 16

5 Methodology **20**

5.1 Parameter optimization process with Genetic Algorithms 20

5.2 Walk-Forward Configuration 25

6 Results **28**

6.1 Analysis Roadmap 28

6.2 Optimal Parameters and the 2D Results 29

6.3 3D Results 31

6.4 Comparisons against Wu et al.'s Results and Buy & Hold benckmark . . . 32

7 Conclusions **36**

Appendix **41**

1. Introduction

In the early 1970s, shortly after the inception of NASDAQ as the world's first electronic market, the New York Stock Exchange (NYSE) introduced the *Designated Order Turnaround (DOT)* system, creating an electronic routing mechanism for orders from brokerage firms directly to the trading floor. This surpassed the physical delivery of paper tickets and led the rudimentary manual methods to an end.

Thanks to the integration of advanced trading technologies, increasingly leveraged by computational intelligence, the financial system has fostered the adoption of complex and automated mechanisms that enhance and projects new ways of trading (Wu et al., 2021). This change in structure enabled traders to rapidly assess and roll out strategies grounded in massive amounts of historical data, marking a departure from markets run solely by traditional long-horizon institutional approaches (Zarattini, Barbon, and Aziz, 2024).

The rise of rapid buying and selling — broadly defined as financial trading — has fundamentally transformed market participation. This evolution is perhaps most evident in the behavior of day traders; although they represent a mere 1% of participants, they command a disproportionately large share of market activity[1](Lundström, 2018). Operating within this high-volume environment, these traders frequently compete directly against sophisticated *high-frequency trading* (HFT) algorithms (Zarattini, Barbon, and Aziz, 2024). This collective intensity results in a pronounced concentration of trading volume within short timeframes, reflecting the contemporary dominance of intraday strategies (Wu et al., 2021). Ultimately, this shift towards temporal density has entailed the development of advanced computational models, thereby catalyzing the widespread adoption of algorithmic trading.

Considering the vast existence of research concerning construction, analysis and testing of algorithmic trading strategies, this work will benchmark the academic research taken in 2021 by Wu et al. (2021), due to its mathematical rigor in the deployment of their momentum-type strategy and their unique use of Genetic Algorithms to solve the parameter-heavy optimization of their strategy parameters.

[1]Due to their high frequency of execution, day traders are estimated to account for approximately 20% to 50% of the total traded volume, depending on the specific marketplace and measurement period.

Even though they used the Taiwan Index Futures (TXF) as their underlying asset, this dissertation seeks to test the model's resilience in a significantly more efficient and competitive environment: the U.S. futures market. Consequently, we have chosen the CME E-mini S&P MidCap 400 futures — identified in the platform as EW/EMD[2]. Using this instrument preserves an important methodological feature of reference study, since the empirical exercise is conducted on exchange-traded futures rather than broker CFDs or ETFs.

From a practical standpoint, EMD is one of the most operationally consistent U.S. index future readily available, and even though we could have used the ES for this dissertation, which replicates the S&P 500 futures market, both futures serve as a reasonable proxy of the US market.

While the EMD and the ES track different segments of the market, they share a very high positive correlation. As shown in the chart, both indices follow the same directional trends and react simultaneously to the same economic events. Although the EMD tends to be more volatile, its price behavior remains structurally consistent with the E-mini S&P 500 Futures. For this reason, the EMD provides a valid environment as well to test the robustness of our methodology.

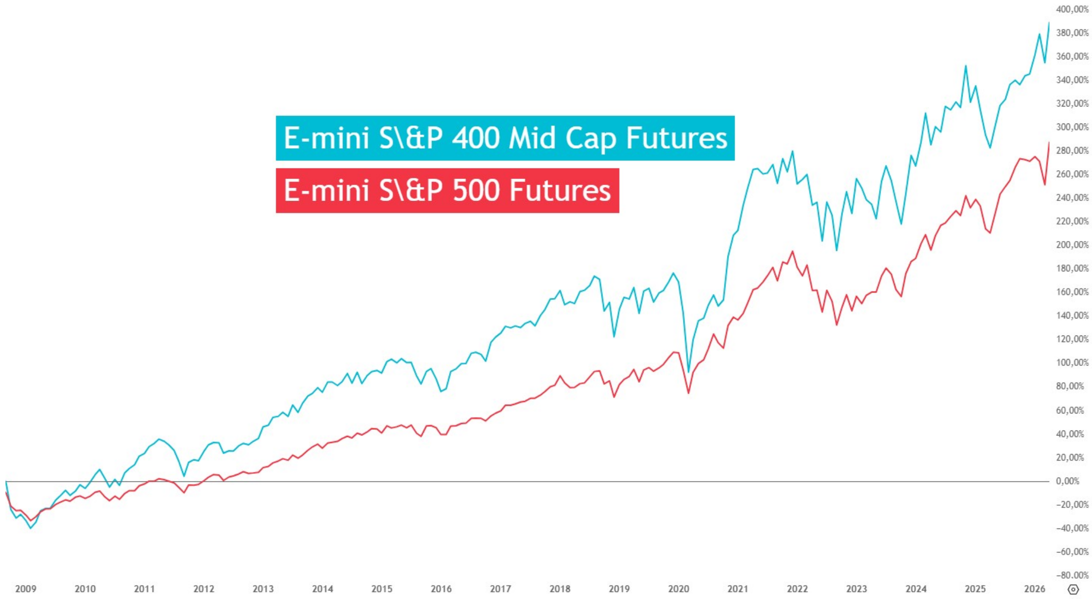

Figure 1.1: *Graphical view from Tradingview of the comparison between EMD and the ES, with the same % scale.*

[2]It is worth mentioning that EW does not stand for Equal-Weighted but for End-of-the-Week. It is a Market-Cap Weighted instrument, and the official ticker in the CME for this instrument is EMD.

Another pillar of this dissertation, and one of the biggest milestones, is the replication and analysis of the Evolutionary-ORB model created by Wu et al. (2021). What this group of Taiwanese engineers and mathematicians constructed corresponds to a family of momentum driven strategies called *Opening Range Breakout*, or as we tend to call them, ORB strategies. This is a widely recognized technical trading style, employed primarily as a day trading methodology to capitalize on intraday momentum. Fundamentally, the ORB system operates on the premise that the initial price activity immediately following the market open — referred to as the opening range — is the most dynamic period of the trading day, and states that this early volatility and institutional supply or demand imbalance projects the trajectory for the remainder displacement of the session (Zarattini, Barbon, and Aziz, 2024).

Historically, the foundation for the ORB technique was laid by *Crabel* (1990), who viewed the strategy as a method for identifying and profiting from directional days, or days of expansion, distinct from calmer, contractionary days (Holmberg, Lönnbark, and Lundström, 2013). As a directional strategy, ORB posits that if the asset's price exceeds a predetermined threshold above or below the opening price, the probability favors the continuation of that directional move until the market closes for the day (Holmberg, Lönnbark, and Lundström, 2013). By this means, traders using ORB establish clear upper and lower bounds based on the opening price, and a position (either long or short) is entered only if the price decisively breaks out of this range.

The successful implementation and sustained profitability of the ORB strategy hinge entirely upon the precise calibration of numerous interconnected variables, such as the parameters for measuring the opening range, the time of the range, the entry and the closing signals, etc. At its core, the system will always require setting the optimal range or threshold (ρ), which must be dynamically narrow enough to capture genuine intraday momentum signals while simultaneously being wide enough to filter out random market noise (Lundström, 2018). In contrast, modern ORB systems typically incorporate mechanisms like trailing stops, stop-losses, and take-profits to manage risk and lock in gains, making the configuration process increasingly complex. Given the dimensionality and possible interdependence among these inputs, optimizing the ORB strategy is far from straightforward; relying on intuitive or discretionary judgment is inherently unreliable in capturing the nuanced balance between risk and reward. Furthermore, leveraging brute force methods, such as an exhaustive grid search, quickly becomes computationally prohibitive, which could lead to over-fitted models if analyzing large amounts of data (Wu et al., 2021). Consequently, realizing the potential profitability of ORB requires moving beyond simple methods toward highly efficient computational intelligence algorithms tailored for navigating this extensive parameter space.

Addressing these computational challenges requires a transition towards evolutionary techniques. Specifically, this project adopts the use of *Genetic Algorithms* (GAs), as Wu et al. (2021) followed in their research. GAs are known for simulating natural selection to efficiently evolve parameter sets, ensuring that only the most profitable and robust configurations survive and proliferate. The iterative procedure allows the algorithm to systematically explore a significantly wider and more complex solution space, making it highly effective for solving non-convex multi-peak optimization problems encountered in algorithmic trading (Wu et al., 2021). Consequently, GAs efficiently derive rational, optimal parameters for both threshold determination and protective closing strategies, as we will discuss later on.

Furthermore, to ensure a rigorous comparative analysis with the original study by Wu et al. (2021), this replication is tested over an extended historical period spanning from 2009 to 2018. This specific time-frame subjects the strategy to various macroeconomic cycles and volatility regimes, allowing for a robust validation of the algorithm's adaptability.

Finally, in order to bridge the gap between academic theory and institutional-grade execution, *StrategyQuantX* (SQX) is utilized as the central research ecosystem. Although SQX is a popular choice among sophisticated retail investors, it functions as a high-level "strategy factory", capable of producing professional and corporate-grade algorithmic systems. Its *AlgoWizard* module allows for the deep, granular construction of the complex logic described by Wu et al. (2021), while its proprietary *Optimizer* provide a robust environment to evolve strategies with a depth that exceeds standard back-testing platforms.

The process of creation and optimization that we will explain in the methodology begins, accordingly, with the construction of the system, and ends with the optimization, including the parametrization of both the Genetic Algorithm and its Walk-Forward rolling window. However, the process of landing a theoretical paper into a production-ready software environment presented several challenges and technical hurdles. Specifically, replicating certain academic nuances — such as the sigmoid fitness function or the specific hybridization of genetic selection procedures mentioned in the source literature of Wu et al. (2021) — required a series of creative engineering adaptations. These adaptations, while seemingly minor, were essential to reconcile the paper's experimental logic with the practical requirements of an algorithmic engine as SQX. As will be shown, while the evolutionary pathway might differ slightly from the original study, the final output remains a high-fidelity and robust realization of the model.

2. Motivation

Whereas Wu et al. (2021) tested the evolutionary ORB model on Taiwan Index Futures, the present dissertation applies the same methodological logic to the CME E-mini S&P MidCap 400 futures (EMD), as mentioned before. In this sense, this paper should be understood as a methodological replication with the possibility of external-market validation rather than an exact replication of the original underlying. This choice is defensible because it preserves the use of exchange-traded futures, because EMD was the most operationally consistent U.S. index future available in StrategyQuant, and because its standardized CME contract structure makes its implementation more coherent than moving to non-equivalent vehicles such as CFDs or ETFs. At the same time, keeping the study within a U.S. index-futures context allows the dissertation to lean on other works that studied the ORB in the S&P500.

Worth to mention that Lundström (2020) showed ORB returns on S&P 500 futures vary materially across volatility states, Tsai et al. (2014) documented ORB profitability across Taiwan, Hong Kong, and the U.S., and Zarattini et al. (2024) used the S&P 500 as a central benchmark when comparing active intraday strategies with passive exposure. For that reason, and as we have mentioned before in the introduction, the role of EMD should be framed carefully: it is not the S&P 500 itself, since the S&P MidCap 400 and the S&P 500 represent different segments of the U.S. equity market, but it is still a reasonable U.S. futures proxy for testing whether the ORB framework can transfer to a liquid and competitive equity-index market.

While analytical efforts concerning the Opening Range Breakout strategy have traditionally focused predominantly on defining the optimal entry threshold or range, relying solely on the market close as the exit price is fundamentally precarious (Lundström, 2018). Such conventional methods often lack the necessary robustness, exposing the trader to significant losses if market momentum fails or signals prove unreliable. Indeed, assuming position clearance only at the end of the trading day may misrepresent true profit potential, as real-life systems must incorporate stop-loss measures to limit intraday drawdowns (Holmberg, Lönnbark, and Lundström, 2013).

In this context, the framework proposed by Wu et al. (2021) emphasises not only the opening range, but also the rigorous exit management — a factor that provides a primary justification for replicating their methodology. By extension, these intraday strategies offer a crucial defensive advantage in today's volatile climate by eliminating exposure to overnight risks, such as unpredictable price gaps. Because positions are deliberately initiated and liquidated within a single session, the trader avoids the "gap risk" inherent in multi-day holdings (Zarattini, Barbon, and Aziz, 2024). The relevance of analyzing such momentum-based systems is further underscored by the fact that their profitability often correlates positively with the underlying asset's volatility.

Studying intraday systems applied to highly liquid instruments — specifically proxies of the S&P 500 Future — is essential, as these markets offer substantial profit potential for professional day traders. Critically, effective intraday strategies often demonstrate minimal correlation with overall market movements, frequently maintaining a near-zero beta coefficient (Zarattini, Barbon, and Aziz, 2024). This positioning transforms them into vital tools for modern risk management, providing an important source of uncorrelated alpha in a diversified portfolio (Zarattini, Barbon, and Aziz, 2024).

Ultimatelly, another important motivation for this research is to provide an academic, rigorous and advanced validation of StrategyQuant as a professional-grade commercial tool. There is a genuine interest in exploring whether platforms that are accessible to the broader trading and research community can actually handle the weight of complex scientific papers and adapt their findings to new scenarios.

By landing the theoretical Evolutionary-ORB model into this ecosystem, we are essentially testing the portability of academic knowledge. As we have transitioned from the specific market conditions of the original study to a more global and efficient context, using this software allows us to see if advanced financial engineering can be successfully moved from isolated labs into versatile, production-ready platforms. Proving that we can faithfully replicate and even "evolve" a sophisticated model within such a tool confirms that these platforms are robust enough to serve as reliable quantitative laboratories, effectively bridging the gap between high-level theory and practical, real-world implementation.

3. Objetives

3.1 General Objectives

The general objective of this dissertation is to design, optimize, and evaluate an intraday Opening Range Breakout (ORB) strategy on the E-mini S&P 400 MidCap (EMD), using evolutionary optimization (Genetic Algorithms) implemented through StrategyQuantX, and using the evolutionary ORB framework with protective closing strategies proposed by Wu et al. (2021) as the main methodological reference.

Thus, the main goal is to test whether the ORB logic proposed in the reference paper can remain competitive when transferred from the original Taiwanese futures market to a highly liquid U.S. equity index futures environment, with risk controls treated as a core component rather than as an afterthought.

3.2 Specific Objectives

3.2.1 Technical Design

The replication of their core logic system is not an easy job, since it must be fully specified, executable, and therefore testable afterwards. In practice, that means moving from a general description ("trade breakouts after the open") to a rule set where every relevant decision is defined in an unambiguous way: what exactly counts as the opening range, how the upper and lower breakout levels are computed, what constitutes a valid breakout, and what happens when price action becomes inconsistent with the original breakout hypothesis. After reaching the end-to-end project, it's fair to mention that this part supposed more than half of the working hours.

As mentioned before, we will create the strategy using StrategyQuant's AlgoWizard module, because it forces the logic to be written as a sequence of explicit conditions rather than as an interpretation. This replication step is not a formality: if the trading rules are even slightly vague, the backtest results become impossible to interpret, since small implementation choices can silently change the entire behavior of an intraday system. For this reason, we will treat the implementation itself as part of the research contribution: the ORB logic will be modeled so it can be repeated, audited, and modified in a controlled way while preserving the structure of the reference approach.

3.2.2 Evolutionary Optimization

Another specific objective of this dissertation is to let the strategy's parameters be selected through an evolutionary process rather than through intuition or manual trial-and-error. Concretely, I will use the Genetic Optimizer built into SQX to search for the parameter combination that best replicates both *GA_ORB_ret* and *GA_ORB_sharpe* -both metrics exposed in Wu et al. dissertation, that we will explain later on-. Using an evolutionary optimizer is crucial for treating parameter selection as a structured research step. The point is not to claim that a GA finds a "true" optimum in any absolute sense, but to use a method that is designed to explore a large, irregular search space efficiently and to converge toward configurations that score well under a chosen fitness criterion. This is directly aligned with the logic of the reference framework, where selecting ORB thresholds and protective closing parameters is presented as a non-trivial optimization problem and evolutionary computation is used to derive rational parameter sets without relying on exhaustive grid search.

3.2.3 Comparative Evaluation

Our fourth and final objective is to empirically validate the ORB strategy by benchmarking its performance against a passive Buy & Hold approach in the EMD index. By contrasting the optimized versions of the strategy against each other and against the market benchmark, the study aims to determine if the evolutionary approach consistently generates alpha and/or provides superior drawdown protection. Ultimately, these results will be cross-referenced with the findings of Wu et al. to assess whether the observed behaviors align with established literature.

4. Conceptual Framework

4.1 From Market Efficiency to Adaptative Markets

The *Efficient Market Hypothesis* (EMH), developed by Fama in 1970, suggests that asset prices fully reflect all available information and evolve as *random walks*, theoretically making it impossible to consistently profit from historical price data (Fama, 1965); (Fama, 1970). However, while markets may trend toward efficiency in the long run, extensive day trading research reveals exploitable inefficiencies at the short-term and intraday levels (Holmberg, Lönnbark, and Lundström, 2013). These intraday deviations from a random walk — such as momentum anomalies — are frequently driven by the cognitive biases of irrational investors, including herd instinct, confirmation bias, and market overreaction or underreaction to news (Holmberg, Lönnbark, and Lundström, 2013). Additionally, temporary pricing anomalies can be exacerbated by microstructural frictions, such as short-term volatility clustering or even scheduled lunch breaks in certain exchanges (Holmberg, Lönnbark, and Lundström, 2013).

To reconcile the traditional EMH with these behavioral realities, Andrew Lo (2004) proposed the *Adaptive Markets Hypothesis* (AMH). Rather than viewing financial markets as rigid, immutable physical systems, the AMH conceptualizes them as dynamic, evolutionary environments where participants adapt using simple heuristics. Under this biological framework, markets continuously shift between periods of efficiency and periods of relative inefficiency, depending on changing business conditions and the influx of new competitors (Lo, 2004); (Macedo, Godinho, and Alves, 2020). Because of this, investment strategies undergo natural life cycles of returns and losses. A trading strategy might be born and function profitably during a specific market regime, but as investors recognize and exploit the opportunity, their adaptive behavior alters the overall market structure. Eventually, the specific inefficiency is arbitraged away, and the strategy becomes totally or partially obsolete (Tsai et al., 2014).

In this context, the evolutionary nature of the financial markets highlights a critical flaw in traditional algorithmic trading: relying on a strategy with fixed, static parameters is unsustainable, as its performance will inevitably degrade when market conditions evolve

(J. R. Hill, Pruitt, and L. Hill, 2000). To mitigate this risk, Genetic Algorithms (GA) become essential.

4.2 Genetic Algorithms

A Genetic Algorithm is a process based on evolutionary computation that employs metaheuristic methods for solving optimization problems (Wu et al., 2021). The roots of the idea go back to Charles Darwin's theory of natural selection, which describes how certain traits become more common because they offer an advantage for survival. Formally introduced by John Holland in 1975 (Allen and Karjalainen, 1999), GAs maintain a population of possible solutions (often called individuals or chromosomes) to a given problem, gradually improving them by imitating biological reproduction.

The quality of each solution candidate is evaluated according to a problem-specific fitness function, which measures the quality of the solutions corresponding to its genetic structure. Chromosomes with higher fitness scores are more likely to be carried over into the subsequent generation. The process of evolution involves three fundamental operations:

- **Selection:** Solutions with higher fitness are more likely to be chosen for the next generation. Methods such as *Roulette Wheel Selection* assign survival chances based on proportional fitness.
- **Crossover (recombination):** The main way the algorithm explores new possibilities by combining portions of two parent chromosomes to form a new offspring, ideally keeping the useful traits from each parent.
- **Mutation:** Introduces small, random alterations to chromosomes to keep the population diverse and avoid getting stuck in local optima with overly similar solutions.

In the context of this research, the parameter selection is driven by two distinct fitness functions: one focused on maximizing raw profitability (*GA_ORB_Ret*), and another prioritizing risk-adjusted stability through the Sharpe ratio (*GA_ORB_Sharpe*). To systematically evaluate the impact of risk management, the optimization is divided into three progressive layers for each function: a naive version (focusing solely on adaptive entry thresholds), a version incorporating the stop-loss mechanism (SL), and a complete version that adds the dynamic take-profit (SL+TP). Consequently, this methodology yields six distinct optimized models, allowing for a robust comparison between aggressive momentum capture and stringent risk management.

The selection of Genetic Algorithms as the core optimization methodology is strictly justified by the inherent characteristics of financial markets; financial data is notoriously noisy and presents non-convex, multi-peak optimization landscapes with irregular or non-differentiable objective functions. In such environments, traditional calculus-based methods (like gradient descent) frequently fail or get trapped in local optima (Wu et al., 2021). To overcome this, the evolutionary mechanisms of GAs — specifically crossover and mutation — inject necessary randomness into the population, ensuring a broad exploration of the solution space and effectively preventing the algorithm from premature convergence on suboptimal rules (Macedo, Godinho & Alves, 2020).

Furthermore, unlike exhaustive grid searches that suffer from the curse of dimensionality and an elevated risk of overfitting when the parameter space is vast, GAs provide a highly efficient metaheuristic alternative. Through a process called "implicit parallelism", GAs can evaluate a vast number of virtual "what-if" scenarios — sometimes trillions of combinations — simultaneously, requiring little to no prior operator bias regarding the underlying market model (Noever and Baskaran, 1994). This efficiency allows them to easily embody the multi-dimensionality of trading problems, meaning they can simultaneously optimize both the structure of the solution (e.g., selecting which protective strategies or indicators to activate) and their specific numerical parameters at the same time (Macedo, Godinho, and Alves, 2020); (Allen and Karjalainen, 1999).

Unlike more opaque artificial intelligence approaches, such as Neural Networks, GA-based models offer high interpretability. They do not just provide a decision, but they allow the extraction of robust, transparent trading rules from historical data (Allen and Karjalainen, 1999); (Noever and Baskaran, 1994). Ultimately, this optimal trade-off between the exploration of new, untested possibilities and the exploitation of successful historical traits makes GAs uniquely equipped to navigate changing economic climates. By utilizing rolling training windows to periodically re-evaluate and evolve critical variables — such as entry thresholds and protective stop-loss levels (Wu et al., 2021) -, GAs ensure that a trading model remains resilient and aligned with the market's current evolutionary stage, effectively avoiding the obsolescence that plagues rigid strategies.

4.3 Microstructure and Intraday Momentum Anomalies

4.3.1 The Opening Range Phenomenom

Beyond this optimization process of genetically evolving the best sets of parameters within a strategy, the effectiveness of such algorithmic systems is strictly and fundamentally governed by the quality of the signal they process, which in our specific system is found in the phenomenon of the opening range. This interval (the opening range) represents a critical period of price discovery where financial markets actively process and assimilate information accumulated overnight. Although markets cannot respond in real time to events that occur after trading hours, this contained information is rapidly priced during the earliest stages of the market open (Tsai et al., 2014). Consequently, the heightened volatility observed in the first few minutes of a session is not merely random market noise, but rather a highly reliable directional signal (Tsai et al., 2014). This early price action reveals valuable insights regarding the underlying institutional supply and demand imbalances that are likely to dictate the market's direction throughout the day (Zarattini, Barbon, and Aziz, 2024). By accurately analyzing the dynamics of this initial volatility, traders can systematically extract market information to forecast the probable price trajectory for the remainder of the session (Tsai et al., 2014).

4.3.2 Contraction-Expansion Principle and Intraday Mass Psychology

This mechanics of market volatility can be further explained by the *Contraction-Expansion* principle, initially established by Toby Crabel in 1990 and extensively analyzed by researchers like Holmberg et al. (2013). According to this theoretical foundation, financial markets inherently alternate on an intraday basis in a cyclical manner between regimes of contraction — characterized by calm periods of modest price movements and market equilibrium — and regimes of expansion, which feature high volatility and strong directional trends (Crabel, 1990). Within this context, the Opening Range Breakout strategy is not conceptualized as a predictive forecasting tool, but rather as a precise mechanism for identifying these inevitable intraday shifts into expansion (Holmberg, Lönnbark, and Lundström, 2013). By setting predetermined price thresholds around the opening range - as we will see in the upper and lower bounds of this daily range - the strategy waits for the market to decisively break out of its normal bounds, theoretically allowing traders to recognize and profit from only days of expansion.

Once a period of expansion is successfully identified, the prevailing market behavior over the intraday horizon shifts distinctly towards momentum rather than mean reversion (Holmberg, Lönnbark, and Lundström, 2013). This intraday momentum is theoretically driven by explainable market participant behavior, rooted in the mass psychology of the traders behind the price action (Zarattini, Barbon, and Aziz, 2024), alongside the cognitive biases of irrational investors (Holmberg, Lönnbark, and Lundström, 2013). As highlighted by Sönnert (2015), when irrational investors observe a breakout, behavioral phenomena create a tangible gap in the random walk, pushing prices further in the direction of the initial trend. This herding effect provides a rational explanation of why prices deviating from equilibrium tend to keep trending, enabling momentum-based systems to capture substantial profits before the market closes (Sönnert, 2015).

Consequently, Opening Range Breakout strategies stand out as one of the most common intraday trading approaches that operationalize these intraday momentum dynamics. While mostly based on technical analysis, certain variants may also employ fundamental methodologies (Zarattini, Barbon, and Aziz, 2024).

4.4 The Opening Range Breakout system propsed by Wu et al.

Theoretically, the main concern of the ORB is the establishment of both upper and lower bounds. In the original and most basic model, these boundaries are defined by the highest and lowest prices in the first half hour of the trading day, though multiple variants soon emerged (Tsai et al., 2014). Once the price breaks one of these bounds, the model triggers a long or short entry accordingly (Lundström, 2018). However, standard ORB rules, which typically mandate liquidation only at the end of the trading session, can often fall short on strength, as they tend to overlook specific market traits and intraday risks (Wu et al., 2021).

To address this, the research conducted by Wu et al. (2021) identifies a fundamental structural fragility in traditional ORB systems: their inherent lack of robustness against market uncertainties and the absence of mathematically defined closing rules. To mitigate these risks, the authors proposed the *Evolutionary ORB model* ($GAORB$), which utilizes a Genetic Algorithm to optimize entry thresholds and integrate protective closing strategies. This optimized model would derive from their initial ORB idea: the *Threshold Adjusting* ($TAORB$) mechanism. This $TAORB$ model fundamentally calculates dynamic boundaries (both upper and lower boundaries, B_u and B_l, respectively) as a function of early-session volatility (σ):

$$B_u = h + \epsilon_1 \sigma \tag{1}$$

$$B_l = l - \epsilon_2 \sigma \tag{2}$$

In this formulation, h and l represent the high and low of the range, σ stands for the standard deviation of the opening range, and ϵ_1 and ϵ_2 serve merely as discrete coefficients that multiply the effect of σ. By shifting these thresholds outward during periods of high volatility within the opening range (since σ would exacerbate both upper and lower bounds), the system effectively filters out intraday noise, requiring a more decisive breakout to trigger an entry.

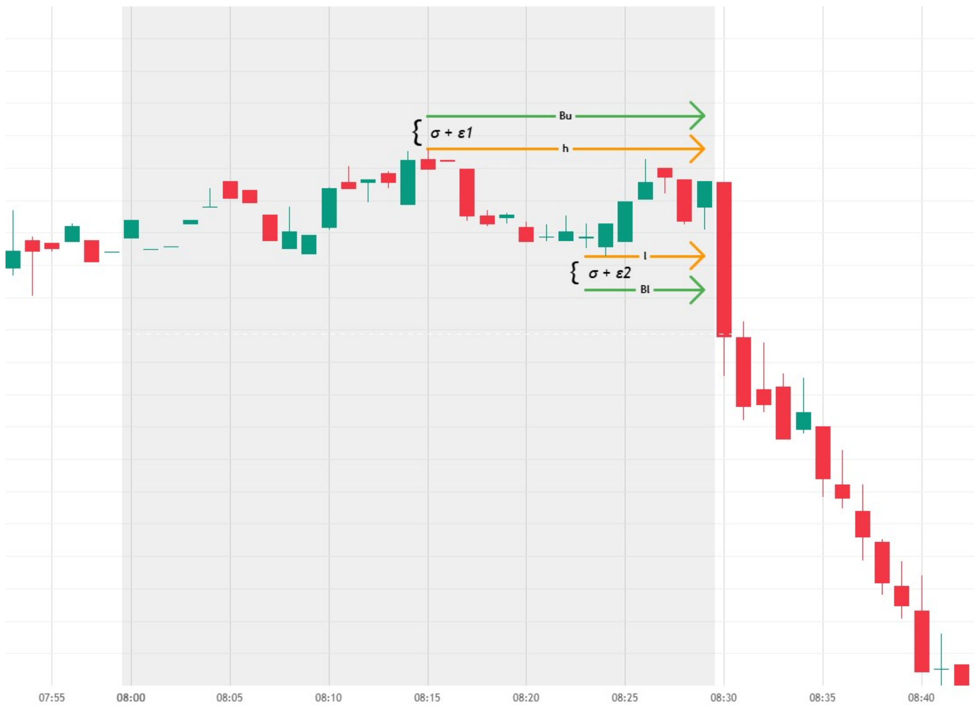

Figure 4.1: *Visual example in TradingView of the upper and lower bounds. In this singular example, the red candle at 8:30 would surpass the lower bound, activating a short position.*

However, accurately defining the entry thresholds solves only half of the algorithmic equation. As Wu et al. (2021) indicate, the $GAORB$ framework integrates two mathematically defined protective closing strategies: a stop-loss and a dynamic take-profit mechanism. The stop-loss (SL) mechanism is designed to liquidate positions immediately if the initial breakout fails. Rather than using an arbitrary fixed value, the stop-loss is dynamically scaled to the intraday volatility by setting the threshold as a fraction of the adjusted opening range width (Wu et al., 2021):

$$SL = TSL \times (B_u - B_l) \tag{3}$$

The coefficient TSL dictates the strictness of this protective leash, closing the trade if the unrealized loss exceeds this limit.

Conversely, the take-profit mechanism functions as a dynamic trailing exit designed to lock in realized gains. It activates once the unrealized profit reaches a predetermined threshold (TP). From that high-water mark, if the price retraces from its maximum peak by a specific ratio — defined by the parameter RDD (Ratio of maximum unrealized gain DrawDown) — the position is immediately closed to prevent profitable trades from turning into losses (Wu et al., 2021).

Ultimately, the strength of the evolutionary ORB model lies in its capacity to handle multidimensionality. By utilizing the Genetic Algorithm to continuously evolve the threshold adjustment coefficients (ϵ_1, ϵ_2) alongside the protective closing variables (TSL, RDD), the system can simultaneously balance aggressive momentum capture with stringent risk management without falling into the trap of over-optimization.

Figure 4.2: *Visual example in TradingView of the upper and lower bounds, as well as the stop-loss and take-profit mechanisms. Continuing the previous example shown in figure 3.1, we can spot how the retracement in price as big as a ratio of the stop-loss length would activate our take profit. In this case, RDD is equal to 1 for simplification purposes.*

5. Methodology

5.1 Parameter optimization process with Genetic Algorithms

As established in the conceptual framework, GAs are uniquely suited to handle complex financial optimizations. In our specific implementation, the search space for these parameters exceeds 65,000 combinations (in the models that incorporate both stop-loss and take-profits mechanisms), rendering traditional grid searches computationally prohibitive. The Genetic Algorithm proved highly efficient in navigating this non-convex landscape, reducing computational overhead by approximately 89% for stop-loss models and up to 97% for the most complex variants (Wu et al., 2021).

To ensure the reproducibility of the GAORB model, the framework was translated into an executable trading system using StrategyQuantX (SQX). The system is configured to evaluate conditions On Bar Open, ensuring a discrete decision process where information from the previous completed one-minute bar is used to trigger actions at the new bar's start. The operational environment is governed by a strict set of intraday constraints: a trading window from 08:00 to 16:00, a maximum of one trade per day to prevent overtrading, and the mandatory liquidation of any open positions at the session close.

Taken together, these blocks show that the model is a structured intraday state machine composed of sequential stages: daily reset of previous-day parameter's information, opening-range construction, parameter decoding (after the range between 8:00 and 8:30 is finished), volatility-adjusted threshold calculation (where the stop-loss and entry are calculated), conditional entry, and retracement-based liquidation (for the take-profit mechanism). This technical structure serves as a rigorous research instrument, allowing the academic ORB model to be stress-tested and optimized in a way that is transparent, auditable, and fully explicit at every decision point.

A detailed breakdown of the technical rules and variable initialization is provided in Appendix.

Once the trading logic has been translated into explicit rules, the next methodological problem is no longer how the strategy behaves, but how its parameters should be selected. In the original *GAORB* framework, the optimization problem emerges from the need to choose the threshold-adjustment coefficients (ϵ_1, ϵ_2) and the protective-closing parameters

(TSL, RDD) in a rational way. Wu et al. (2021) showed that, once these variables are encoded jointly, the search space exceeds 65,000 possible combinations, which makes brute-force optimization unattractive due to computational costs and overfitting risks.

To address this complexity, Wu et al. (2021) encode each candidate solution as a 16-bit chromosome. The first 12 bits represent the threshold coefficients (6 bits for ϵ_1 and 6 bits for ϵ_2), while the remaining 4 bits encode the protective-closing parameters. This design allows the algorithm to represent 64 possible values for each ϵ coefficient (mapped to the interval from -1 to 1) and 4 possible values for each exit parameter (TSL and RDD), mapped to the discrete set $\{1/3, 2/3, 1, \infty\}$. This encoding is relevant because it clarifies that the GA is not evolving generic trading rules from scratch but rather optimizing the internal parameters of a pre-specified ORB architecture.

Therefore, a critical component of our methodology is the discrete mapping of these optimization parameters within the StrategyQuant environment. Rather than allowing the Genetic Algorithm to search through infinite values, we have defined a specific search space that replicates the original research logic. These parameters are mapped as follows:

Table 5.1: *Range of Parameters*

Parameter	**Discrete mapping**
ϵ_1	$[-1, 1]$
ϵ_2	$[-1, 1]$
TSL	$\{0.33, 0.66, 1.0, \infty\}$
RDD	$\{0.33, 0.66, 1.0, \infty\}$

It is worth mentioning that the implementation includes a deliberate modification to the reference framework by Wu et al. (2021): the TP activation threshold, which is calculated with TSL as well, remains capped at 1.0 even when the hard stop-loss is inactive. This adjustment ensures the preservation of dynamic profit protection in all scenarios that must optimize a take-profit possibility, even those where the primary stop-loss has been relaxed. This is explained with more precision in the appendix.

While the original framework by Wu et al. (2021) utilized fixed manual probabilities for crossover (80%) and mutation (5%), this implementation leverages the proprietary evolutionary engine of StrategyQuantX. The SQX Optimizer is designed to prioritize search efficiency and robustness by utilizing an auto-adaptive genetic process. Instead of requiring manual tuning of low-level genetic operators — which can introduce additional optimization bias — the software manages the balance between exploration andexploitation internally. This approach ensures that the search remains focused on the convergence of the fitness function (Net Profit or Sharpe Ratio) across the discrete parameter space defined in our mapping.

The optimization process in SQX still adheres to the core principles of the *GAORB* model: it initializes a population of individuals (parameter sets), evaluates their fitness on the training data, and evolves them through successive generations. The termination logic remains consistent with the reference study, stopping after a set number of generations or upon reaching a stagnation plateau where no further improvement is found. By utilizing this professional-grade optimizer, the study transitions from a purely academic genetic simulation to a production-ready evolutionary search, ensuring that the resulting parameters are not only mathematically optimal within the model but also resilient to the market noise of EMD.

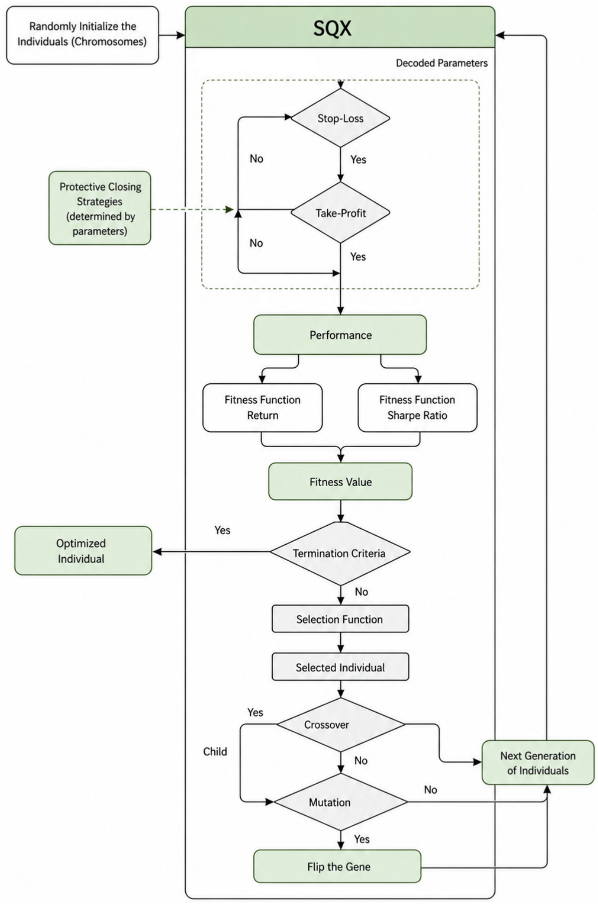

Figure 5.1: *Visual representation of the Genetic Algorithm framework, following the Wu et al. chart.*

Another detail from the paper that deserves mention is the selection stage. Rather than relying on a single selection operator, Wu et al. (2021) combine several standard GA procedures, including roulette-wheel selection, save-best selection, and stochastic universal sampling, with each method contributing one third of the surviving population. The methodological significance of this mixed design is that it avoids overcommitting to one selection logic: part of the next generation is driven by fitness-proportional survival, part by elitism, and part by diversity preservation. The genetic optimization was conducted using a consolidated evolutionary engine that streamlines the selection procedures described in the base literature. Specifically, the *Save-best* selection method from the paper is implemented within StrategyQuant, ensuring that the highest-performing genetic individuals of each generation are preserved for the next iteration. Although the software employs a standardized selection algorithm rather than a hybridized mix of Roulette and Stochastic Universal Sampling (SUS), the integrity of the search process is maintained.

Given that the total search space is defined by a finite 16-bit parameter structure (65,536 combinations), the standard evolutionary engine is mathematically sufficient to converge on the global optima without the need for specialized academic sampling techniques.

As theoretically defined in the conceptual framework, the model distinguishes between return-based (*GAORB_Ret*) and Sharpe-based (*GAORB_Sharpe*) optimization objectives. While the reference model by Wu et al., 2021 employs a probabilistic sigmoid function to transform these performance metrics into survival probabilities, the implementation in this thesis utilizes the Weighted Fitness Ranking architecture native to SQX. This approach achieves the exact same objective by assigning categorical weights strictly to the Sharpe Ratio or to Net Profit, respectively. If the search is ranked by return, the resulting exercise is conceptually aligned with *GAORB_Ret*, but if the search is ranked by a Sharpe-oriented criterion, it becomes the platform analogue of *GAORB_Sharpe*. Strictly speaking, the replication is not mathematically identical due to the absence of the sigmoid transformation, but at the methodological level the correspondence remains strong: one version of the optimization rewards raw profitability, while the other prioritizes profitability adjusted for volatility.

This distinction between return-based and Sharpe-based optimization is central to the empirical design of the paper and is equally important in this dissertation. A return-maximizing optimizer will tend to privilege aggressive parameter sets that exploit strong trends, whereas a sharpe-oriented optimizer will tend to prefer a more stable return path relative to volatility. In practical terms, the former may produce higher raw profits but greater variability, while the latter may sacrifice some upside in exchange for smoother risk-adjusted behavior. The point is not that one criterion is universally superior, but that the optimization target shapes the character of the final strategy.

What emerges from this protocol is a coherent two-layer optimization design. First, the strategy logic is constructed, ensuring that the model always remains an ORB strategy with adaptive thresholds and protective exits. Second, the GA engine searches within that constrained family of strategies for the parameter configuration that best satisfies the chosen objective. This division is important because it preserves interpretability: The optimizer is not inventing a completely opaque model; it is selecting the most suitable version of a well-defined trading hypothesis.

5.2 Walk-Forward Configuration

A genetic optimizer can identify promising parameter sets, but that alone is not sufficient to make the results credible. If the same data are used both to select the parameters and to evaluate them, the final performance will almost inevitably look better than what could realistically be achieved in unseen market conditions. For this reason, the evolutionary ORB model of Wu et al. (2021) does not rely on a single static optimization over the full sample. Instead, it adopts a rolling, or moving-window, procedure in which parameters are repeatedly re-estimated on recent data and then applied to the next out-of-sample period.

For each testing month, the model uses the previous two months as the training window. The GA is run only on that training segment, the best-performing parameter set is selected, and those parameters are then applied to the following month, which serves as the test segment. After that, the window moves forward by one month and the process is repeated. Thus, if March 2018 is the month to be traded, the optimization is carried out in January and February 2018, and the resulting parameters are used only in March 2018. The same logic is then rolled forward across the full sample.

To replicate this procedure, it is crucial to understand the functional distinction between both phases or windows: During the in-sample phase, the optimizer is allowed to search, compare, discard, and recombine parameter sets freely. This is the exploratory part of the process. By contrast, the out-of-sample phase is evaluative: the winning parameter set from the immediately preceding training window is frozen and applied to unseen data without further adjustment. If a strategy performs well only in the in-sample stage but fails in the out-of-sample segments, then the apparent optimization success is most likely the result of overfitting. Consequently, the walk-forward protocol acts as the first serious credibility filter for the evolutionary ORB model.

To operationalize this methodology, the historical data is divided into sequential In-Sample (IS) and Out-of-Sample (OOS) segments. As a practical equivalent of the rolling window used by Wu et al. (2021), this study implements a 2-to-1 split ratio, consisting of 60 days of training (IS) followed by 30 days of testing (OOS) in a fixed rollover sequence.

Recall that the historical data spans from 2009 to 2018, deliberately chosen to closely align with the testing period utilized by Wu et al. (2021), who evaluated their evolutionary model over a similar decade up to 2018. By mirroring their data window, we ensure a rigorous and "apples-to-apples" basis for comparing the behavior of the ORB framework in the U.S. MidCap market against the Taiwanese index, subjecting both to equivalent macroeconomic cycles and major market events.

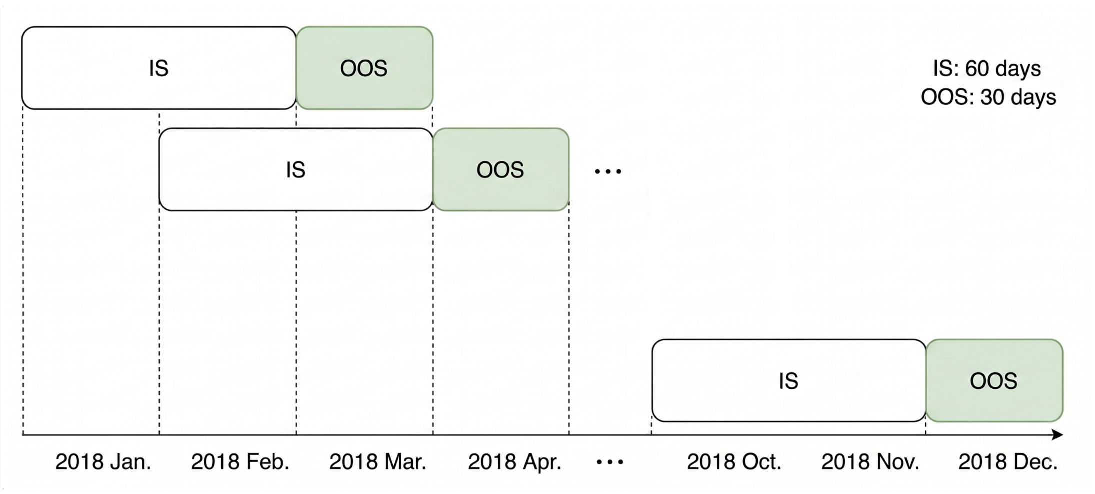

Figure 5.2: *Visual representation of the moving-window, Walk-Forward Analyisis.*

This walk-forward design serves several purposes at once: First, it limits look-ahead bias, because the model is never allowed to optimize based on information from the period it is about to trade. Second, it reduces selection bias, because no single parameter set is imposed on the entire historical sample. Third, it makes the strategy more compatible with the adaptive-markets perspective discussed earlier in the dissertation: if the market environment changes over time, then the parameters of an intraday momentum system should also be allowed to evolve. In other words, the walk-forward procedure is not just a technical safeguard against overfitting; it is also part of the economic logic of the model itself.

This point is particularly relevant for the ORB strategy implemented in this paper. The model depends on variables - such as breakout sensitivity, stop-loss aggressiveness, and retracement tolerance - whose effectiveness can vary substantially across volatility regimes. A parameter set that works well during a smooth, trending environment may become fragile in a choppier or more event-driven market. If a single optimization were performed once over the entire history of the EMD sample, the result could easily reflect an average of incompatible market states rather than a robust response to current conditions. By re-optimizing on a rolling basis, the strategy remains anchored to the most recent information instead of treating the market as stationary.

Finally, we could summarize the relationship between the GA and the walk-forward process as follows: The Genetic Algorithm solves the optimization problem inside each training window, while the Walk-Forward Analysis determines how often that optimization is allowed to restart and how honestly its result is evaluated on new data. In other words, the GA answers the question, "which parameter set is best for the recent past?", whereas walk-forward analysis answers the more important question, "does that selected parameter set still behave acceptably in the immediate future?" Only by combining both stages can the methodology claim to be adaptive without becoming purely in-sample.

6. Results

6.1 Analysis Roadmap

To provide a comprehensive evaluation of the evolutionary ORB framework, this chapter follows a structured analytical roadmap of the results obtained, comparing them with the ones obtained by Wu et al. (2021) and showcasing the major differences and similarities with their proposed model. It is important to note that, to maintain narrative clarity and a clean structural flow, all 2D and 3D optimization parameter charts referenced in this section have been placed in the Appendix. Meanwhile, the core comparative tables and equity curves remain in this main text.

This section will be structured in the following order: First, we will analyze the resulting parameters (ϵ_1, ϵ_2, TSL, RDD) in a unified evolutionary sequence, from the naive to the complete models. Second, we will evaluate the multidimensional 3D synergies between these parameters. Finally, we will compare the equity curves and key performance metrics across all models, contrasting them with the results obtained by Wu et al. and against the passive Buy & Hold benchmark, the EMD.

As defined in the conceptual framework, the six optimizations to be analyzed stem from the two different fitness functions applied across the three progressive protective categories:

1. **Naive ORB:** No stop-loss or take-profit mechanisms are active, focusing solely on the adaptive entry thresholds (ϵ_1, ϵ_2). In this context, leaving TSL and RDD in the value of 3 (so the parameter would have a value of ∞), they will act as no protective systems.

2. **ORB with Stop-Loss (ORB+SL):** Incorporates the TSL parameter to mitigate catastrophic intraday failures.

3. **ORB with SL and Take-Profit (ORB+SL+TP):** The complete model, adding the RDD retracement-based exit to "lock in" gains before market closure.

Before diving into the data, we can establish baseline expectations based on the nature of the evolutionary variables and the findings of the reference study:

- **Impact of Protective Exits:** We anticipate that models incorporating the stop-loss will demonstrate significantly higher Sharpe Ratios and lower Maximum Drawdowns (MDD) than the naive versions. While a tighter TSL may lower the overall win rate by exiting positions that might have eventually recovered, it should stabilize the equity curve.

- **The Take-Profit Trade-off:** Following the reference paper's conclusion, the inclusion of the take-profit via the RDD factor is expected to increase the win rate and provide further stability but at a potential cost to total net profit. In the original study, this mechanism often dismissed profitable trends, a behavior we will look to verify in the more efficient U.S. MidCap market.

- **Fitness Bias:** We expect the $GAORB\_Ret$ family to gravitate towards more aggressive parameters (wider thresholds or relaxed exits) to capture outlier moves. Conversely, the $GAORB\_Sharpe$ family should favor configurations that produce a smoother distribution of returns, even if it means sacrificing some of the upside.

6.2 Optimal Parameters and the 2D Results

To systematically evaluate how the evolutionary ORB model adapted to the EMD market, the resulting parameters of the six evolutionary iterations have been unified in Table 5.1. The analysis of these configurations reveals a clear behavioral shift as protective layers were added.

Table 6.1: *Unified Optimization Results across all Evolutionary Models*

Model	ϵ_1	ϵ_2	TSL	RDD
GA_ORB_Ret	-1	-1	3	3
GA_ORB_Sharpe	-1	-1	3	3
$GA\_ORB\_Ret\_SL$	-0.9375	-1	3	3
$GA\_ORB\_Sharpe\_SL$	0.90625	-0.65625	3	3
$GA\_ORB\_Ret\_SL\_TP$	0.59375	-0.46875	0	2
$GA\_ORB\_Sharpe\_SL\_TP$	0.125	-0.53125	3	0

The results of the first two optimization runs (the naive models) present a striking consensus between the return-oriented and the risk-adjusted models. Despite their different fitness functions, both iterations converged on the same set of optimal parameters: $\epsilon_1 = -1$ and $\epsilon_2 = -1$. Mathematically, a negative coefficient shifts the thresholds "inward" toward the center of the opening range. Rather than waiting for the price to breach the absolute high (h) or low (l) of the initial 30-minute window, the Genetic Algorithm determined that the most profitable and stable entry points occur within the range, specifically one standard deviation away from the extremes. The explanation of this scenario is far from objective, but it could highlight an early daily bias established well before the 30-minute extremes are touched.

When expanding the search to test whether adding a hard exit would enhance stability - the SL models -, a significant observation is that despite being given the option to select active stop-loss levels, both models converged on TSL mode 3. As specified in the methodology, mode 3 effectively deactivates the protective stop-loss, relying instead on the end-of-session liquidation. The rationale for this can be seen in the provided *tsl_mode* optimization charts in the Appendix, where mode 3 displays a vastly superior net profit compared to the tighter stop levels. This stands in contrast to the findings of Wu et al. (2021), who reported that the stop-loss mechanism was essential for stabilizing performance and reducing Maximum Drawdown by approximately 50%. In our specific test, the GA suggests that intraday noise is significant enough that a fixed stop-loss scaled by volatility often terminates potentially profitable trades prematurely.

Within this same SL tier, the Return model maintained the "inward" entry logic, with $\epsilon_1 = -0.9375$ and $\epsilon_2 = -1$. Like the naive return model, it prioritizes capturing the bulk of the intraday move by entering as soon as early momentum is confirmed within the opening range boundaries. In the case of the Sharpe model, we see a shift to positive epsilon values ($\epsilon_1 = 0.90625$). This represents an "outward" entry for long positions, where the price must break the 30-minute high plus nearly one full standard deviation of initial volatility. This shift is methodologically sound; by waiting for a more decisive breakout, the Sharpe-oriented model filters out "low-conviction" moves that contribute to volatility without adding proportional returns. This aligns with the "safety-first" theory in Sharpe-based optimizations discussed earlier.

Finally, the complete versions of the evolutionary framework (SL+TP models) explored the full 16-bit parameter space, including the retracement-based exit (RDD) which introduces a dynamic profit-protection mechanism. The introduction of take-profit parameters led to a notable shift in the entry threshold strategy compared to our naive benchmarks. In both models, the long entry coefficient (ϵ_1) moved into positive territory, shifting the breakout boundary outward. This suggests that when the strategy has the tools to manage a trade's lifecycle (via RDD), the GA prefers higher-conviction entries that have already cleared the initial opening-range noise. In the return-oriented model, the GA selected TSL mode 0 (the tightest stop-loss multiplier) and RDD mode 2 (the most lenient retracement factor). This configuration creates an aggressive profile: the strategy uses a very tight initial "leash" to cut losses quickly if the breakout fails immediately, but once the trade is in profit, it allows for significant market fluctuations (mode 2) before triggering the trailing exit. The Sharpe-oriented model presents a different philosophy: It rejected the hard stop-loss entirely (TSL mode 3) but converged on RDD mode 0, which is the most sensitive retracement setting. This indicates that for risk-adjusted stability in the EMD market, the GA finds more value in locking in gains at the first sign of a reversal than in using a hard intraday stop.

6.3 3D Results

The analysis of a trading system's sensitivity cannot rely solely on two-dimensional snapshots, as parameters in an evolutionary model rarely act in isolation. To truly understand the behavioral landscape of the $GAORB$ framework, we must examine the multidimensional synergies between entry thresholds and protective exits:

By visualizing these relationships in 3D (available in the Appendix), we can identify robust "parameter islands" where the strategy maintains high performance across a range of values, rather than relying on a single overfitted point.

Before analyzing the protective closing mechanisms, it is important to note the relationship between the entry coefficients. A 3D mapping of ϵ_1 (long entry) against ϵ_2 (short entry) reveals a fragmented cluster distribution rather than a linear correlation. This indicates that the optimal long breakout level does not necessarily dictate the optimal short level, or viceversa.

- **Net Profit Landscape:** In the 3D profit maps, we observe a high density of "profitable green bubbles" concentrated around TSL mode 3. However, a secondary competitive pillar exists at TSL Mode 0 when paired with RDD Mode 2. This suggests that for raw returns, the strategy either needs "total freedom" (no stop-loss) or a "tight-and-flexible" mix where a very close stop cuts initial failures, but a loose retracement factor allows profitable trends to breathe.

- **Sharpe Ratio Stability:** The 3D maps for the Sharpe Ratio mirror the profit findings almost exactly. The stability zones coincide with the profit ones, confirming that in the U.S. MidCap market, raw profitability and risk-adjusted quality are driven by the same parameter interactions. Notably, RDD Mode 0 (highly sensitive profit-taking) appears as a robust stabilizer across multiple ϵ configurations.

These findings allow us to form a joint conclusion that both supports and refines the findings of Wu et al. (2021). In the original study, the authors discovered that while the stop-loss mechanism was a vital stabilizer, the take-profit mechanism (RDD) was generally detrimental to performance as it limited sharp rises in the Taiwanese index.

Our 3D analysis on the EMD reveals a more nuanced reality:

- **Stop-Loss Sensitivity:** Unlike the Taiwanese market, where the SL was essential, the U.S. MidCap market exhibits higher levels of intraday noise. Our models frequently favor TSL mode 3, suggesting that hard stops are often "shaken out" by random price action before the true daily trend emerges.

- **The Value of RDD:** While Wu et al. (2021) did not recommend take-profit mechanisms, our GA discovered that RDD (specifically modes 0 and 2) can provide a superior exit architecture when TSL is relaxed.

In a more efficient market like the EMD, where trends may reverse sharply once institutional orders are filled, "locking in" gains via retracement proves to be a more effective volatility-control tool than a fixed stop-loss.

6.4 Comparisons against Wu et al.'s Results and Buy & Hold benckmark

Following the multidimensional parameter analysis, we now transition to the empirical results. This section evaluates the performance of the evolutionary models through their equity curves and key statistical metrics, providing a direct comparison between the return-optimized and risk-optimized families. The table below synthesizes the primary trading results for the six optimized strategies and the naive benchmark. As expected from our previous parameter analysis, the results for the basic return and Sharpe models are identical, reflecting their convergence to the same evolutionary solution.

Table 6.2: *Performance metrics of ORB-based strategies*

Strategy Name	Net Profit	Profit Factor	Sharpe Ratio	Win/Loss Ratio	Ret/DD
ORB_Benchmark	$22,490	1.02	0.04	1.06	0.33
GA_ORB_sharpe+SL+TP	$36,680	1.06	0.83	2.88	0.88
GA_ORB_sharpe+SL	$53,080	1.06	0.25	1.04	0.99
GA_ORB_sharpe	$67,070	1.07	0.47	1.13	1.16
GA_ORB_ret+SL+TP	$29,320	1.12	-0.23	0.14	1.12
GA_ORB_ret+SL	$75,020	1.08	0.47	1.12	1.34
GA_ORB_ret	$67,070	1.07	0.47	1.13	1.16

These metrics highlight a significant performance disparity when contrasted with the original findings of Wu et al. (2021). In the Taiwanese market (TXF), models incorporating a stop-loss mechanism (*GA_Ret_SL* and *GA_Sharpe_SL*) achieved an average annual return increase of 2.667% (peaking at 9.303%) and a Sharpe Ratio improvement of approximately 1.0 unit (peaking at 2.495). Furthermore, the original research reported that the stop-loss mechanism reduced the Maximum Drawdown by half, providing a substantially more stable growth curve. In our U.S. MidCap environment, while the evolutionary approach still generated profits, the absolute risk-adjusted stability remained lower, suggesting that the higher liquidity and noise of the EMD market imposes stricter limits on the “stabilizing” power of fixed stop-loss parameters.

Following the visualization structure used by Wu et al. (2021), we present the equity growth of the strategies alongside the underlying EMD index and the naive benchmark:

Figure 6.1: *Equity curves for EW/EMD (blue), ORB_Benchmark (green), GA_ORB_ret (red) ad GA_ORB_sharpe (orange)*

The first comparison features the basic evolutionary models. As observed in the graph, both strategies significantly outperform the *ORB_Benchmark* in total equity. However, when contrasted with the EMD index (Buy & Hold), the strategies show higher volatility and failed to capture the full structural bull run of the 2012–2016 period. This contrasts with Wu et al.'s (2021) research, whose basic GA models showed a much smoother outperformance over the benchmark.

Figure 6.2: *Equity curves for EW/EMD (red), ORB_Benchmark (blue), GA_ORB_ret+SL (orange) ad GA_ORB_sharpe+SL (green)*

The introduction of the stop-loss logic produced mixed results. While *GA_ORB_ret+SL* achieved the highest raw return (75.02 %), the equity curve remains highly correlated with the underlying index volatility. Interestingly, the *GA_ORB_sharpe + SL* model underperformed the naive versions, suggesting that in this market, a hard stop often cuts trades too early during high-volatility regimes.

Figure 6.3: *Equity curves for EW/EMD (blue), ORB_Benchmark (orange), GA_ORB_ret+SL+TP (green) ad GA_ORB_sharpe+SL+TP (red)*

The complete models incorporating the retracement-based exit (RDD) show the most significant behavioral shift. The $GA\_ORB\_sharpe + SL + TP$ model emerged as the most robust architecture, achieving a Sharpe Ratio of 0.83 and the highest win rate (52.29 %).

Comparing these visual results with the conclusions of the reference study reveals an interesting "Take-Profit Paradox". Notably, Wu et al. (2021) observed that while RDD mechanisms could increase win rates, they tended to misinterpret profitable trends and eliminate the possibility of sharp rises, leading the authors to recommend against their use. However, our data suggests a different reality for the EMD: the proactive profit conservation of the $SL + TP$ configuration is the best model that significantly improves the Sharpe Ratio relative to the benchmark. This suggests that in more efficient markets where trends reverse sharply once institutional orders are filled, "locking in" gains via retracement is a more effective volatility-control tool than the simple loss-cutting provided by a fixed stop-loss.

7. Conclusions

Reflecting upon the completion of this dissertation, we believe the research has successfully bridged the gap between theoretical evolutionary finance and practical algorithmic implementation. Although the empirical results did not fully meet our initial expectations, several lessons concerning market microstructure and strategy adaptation can be drawn from them.

The primary objective of designing and optimizing an intraday ORB strategy using Genetic Algorithms was fully realized through the robust environment of StrategyQuant over a demanding 2009–2018 testing period.

Relying on SQX as our research environment was a critical choice that enhanced the project's technical depth. It provided an industrial-grade testing ground, particularly through its logic construction with its AlgoWizard module and its optimization process through its Optimizer sections. While there is a clear trade-off — namely the 'black-box' nature of some internal genetic settings compared to a custom-coded script — the gain in operational consistency and the efficiency with which it handles complex parameter sets more than justifies this lack of granular visibility, and provides a pretty solid platform not only for small retail traders.

The first specific hurdle — modelling the complex logic of Wu et al. (2021) — required a significant investment of time, as it forced a transition from vague trading concepts to an unambiguous, executable state machine. By defining every decision point, from the initial 30-minute range accumulation to the specific standard deviation-based breakout levels, we achieved a high-fidelity logical and operational replication that is both auditable and repeatable. Furthermore, the evolutionary optimization objective was met by successfully navigating a search space of over 65,000 combinations, proving that GA-based models dramatically reduce the computational burden compared to traditional exhaustive search methods.

The comparative analysis with the reference study by Wu et al. (2021) reveals a fascinating narrative of market adaptation. One of the most striking points of convergence lies in the selection of entry thresholds. While the specific numerical outcomes differ, the logic of "inward" contraction was a possibility explicitly baked into the reference paper's design. Wu et al. (2021) defined the search range for their epsilon coefficients (ϵ_1, ϵ_2) from -1 to 1 in increments of 1/32. This mathematical range acknowledges that the most efficient entry point is not always at the absolute high or low of the opening range, but potentially inside it. In our EMD testing, the Genetic Algorithm frequently gravitated toward these negative epsilon values, effectively pulling the thresholds "inward" to capture early momentum before the price could reach the range extremes where "fakeouts" are common in highly efficient U.S. markets.

However, the divergence regarding the stop-loss mechanism was unexpected and provided a profound lesson. Wu et al. (2021) found that a stop-loss was essential for stabilizing performance and halving the Maximum Drawdown in the Taiwanese index. In contrast, our GA frequently rejected hard intraday stops in favor of the session-close exit. This likely stems from the fact that the EMD market is characterized by a higher level of noise and algorithmic competition than the Taiwan Index Futures. While the original study discouraged take-profit mechanisms as profit-limiters, my most stable results were achieved by incorporating the Retracement-based Exit (RDD), suggesting that in efficient markets, proactive profit conservation is more effective than simple loss-cutting.

The computational experience also highlighted the inherent complexity of evolutionary finance. While we have avoided direct hardware-based comparisons, the data clearly shows that as the dimensionality of the strategy grows — adding parameters for stop-losses and trailing exits — the time required for optimization increases significantly. This is not a linear growth; as the Genetic Algorithm evaluates the intricate interactions between entry thresholds and exit rules across sixty days of training data, the processing load scales with every added observation and variable. This process serves as a reminder that sophisticated risk management is a computationally expensive endeavor that requires a balance between search depth and operational efficiency.

The horizon for future research in this field is expansive, particularly regarding the integration of more advanced computational intelligence. While the Genetic Algorithm has proven its worth as a search engine, the next logical step would be to explore Machine Learning (ML) architectures or Neural Networks to enhance the strategy's adaptability. Rather than relying on static parameters evolved over a 60-day window, a Deep Learning model could be trained to identify non-linear patterns in pre-market volatility to set thresholds dynamically for each specific session. Such a system could potentially predict market “regimes” — switching between momentum and mean-reversion logic — before the opening bell even rings, offering a level of sophistication that goes beyond the capabilities of traditional technical indicators.

Equally important, a vital empirical extension of this research would involve conducting an out-of-sample backtest spanning from 2018 to the present day. This period, characterized by unprecedented market events such as the COVID-19 pandemic, supply chain crises, and a global shift toward a high-inflation regime, would provide an ultimate stress test for the model’s resilience. Evaluating how the evolutionary logic behaves in the face of such extreme structural shifts would confirm whether the system can maintain its competitive edge without degradation or if it requires more aggressive re-optimization cycles.

Beyond artificial intelligence, there is significant potential in optimizing the temporal structure of the strategy. This dissertation utilized a fixed 30-minute opening range as its foundation. However, future studies could treat the duration of the opening range itself as an optimizable variable. Determining whether a 15-minute or a 60-minute window better captures the institutional daily bias also in different markets — such as Gold, Oil, or even high-volatility cryptocurrencies — could significantly improve the strategy’s alpha-generating potential. By continuing to test the *GAORB* framework across diverse assets and integrating predictive ML models, we can continue to refine our understanding of how price discovery at the market open can be systematically exploited for modern risk management.

Finalizing this analysis has been an enlightening journey into the reality of algorithmic trading. It has shown that while no model is “perfect”, the use of evolutionary tools allows us to approach the markets with a rational, structured, and data-driven perspective that far exceeds discretionary intuition.

References

Allen, Franklin and Risto Karjalainen (1999). “Using genetic algorithms to find technical trading rules”. In: *Journal of Financial Economics* 51.2, pp. 245–271.

Crabel, Toby (1990). *Day Trading With Short Term Price Patterns and Opening Range Breakout*. Greenville, S.C.: Traders Press.

Fama, Eugene F. (1965). “The Behavior of Stock-Market Prices”. In: *The Journal of Business* 38.1, pp. 34–105.

Fama, Eugene F. (1970). “Efficient Capital Markets: A Review of Theory and Empirical Work”. In: *The Journal of Finance* 25.2, pp. 383–417.

Hill, John R., George Pruitt, and Lundy Hill (2000). *The Ultimate Trading Guide*. New York: John Wiley Sons.

Holmberg, Ulf, Carl Lönnbark, and Christian Lundström (2013). “Assessing the profitability of intraday opening range breakout strategies”. In: *Finance Research Letters* 10.1, pp. 27–33.

Lo, Andrew W. (2004). “The Adaptive Markets Hypothesis: Market Efficiency from an Perspective”. In: *The Journal of Portfolio Management* 30.5, pp. 15–29.

Lundström, Christian (2018). “Optimal Leverage in Day Trading”. In: *The Journal of Trading* 13.2, pp. 57–68.

Macedo, Luís Lobato, Pedro Cortesão Godinho, and Maria João Alves (2020). “A Comparative Study of Technical Trading Strategies Using a Genetic Algorithm”. In: *Computational Economics* 55.2, pp. 349–381.

Noever, David D. and Subbiah Baskaran (1994). “Genetic Algorithms Trading on the SP 500”. In: *AI in Finance* Fall, pp. 41–47.

Sönnert, Tobias (2015). “Day trading on the gold futures market using Opening Range Breakouts and GARCH”. Bachelor thesis. Umeå School of Business and Economics, Umeå University.

Tsai, Yi-Cheng et al. (2014). “Comparing Profitability of Day Trading Using ORB Strategies on Index Futures Markets in Taiwan, Hong-Kong and USA”. In: *Proceedings of the 10th Annual Conference of the Asia-Pacific Association of Derivatives (APAD)*.

Wu, Mu-En et al. (2021). “Evolutionary ORB-based model with protective closing strategies”. In: *Knowledge-Based Systems* 216, p. 106769.

Zarattini, Carlo, Andrea Barbon, and Andrew Aziz (2024). *A Profitable Day Trading Strategy For The U.S. Equity Market*. Tech. rep. 24-09. Swiss Finance Institute.

Appendix

The following appendix is structured to provide a comprehensive technical overview of the strategy's implementation and the results of the optimization process. It begins by detailing the core rule-based architecture and state variables, followed by a description of the operational settings used within StrategyQuantX, including the genetic algorithm and walk-forward engine configurations. Finally, a visual record of the optimization landscapes— both 2D and 3D — is presented to substantiate the performance metrics discussed in the main text.

7.1 Platform Configuration and Execution Environment

This fir'st part of the appendix provides a granular description of the rule-based architecture developed in the *AlgoWizard* module of StrategyQuant X. The objective is to provide a transparent and auditable record of the deterministic logic used to evaluate the evolutionary ORB framework.

First, we define the operational boundaries and the environmental settings where the trading logic resides.

The strategy's execution is governed by a discrete decision process where all conditions are evaluated *On Bar Open*. This ensures that the system updates its state once per one-minute bar, utilizing information from the previous completed bar and the new bar opening to avoid ambiguous intrabar interpretations. The following global settings define the execution environment:

- **Trading Window:** A limited signal window is active between 08:00 and 16:00.
- **Session Liquidation:** The platform is instructed to close all orders once the signal window ends; if the 16:00 liquidation fails, any remaining position is liquidated automatically at the end of the session.
- **Trade Frequency:** The maximum number of trades per day is restricted to one to capture a single primary breakout event and prevent high-frequency overtrading.
- **Risk Filters:** The strategy disables maximum-distance filters and does not impose fixed platform-level profit targets, as these are managed internally by the ORB logic.

- **MagicNumber:** A unique identifier is assigned to isolate the entries, exits, and state-dependent calculations of this specific strategy.

The following section details the internal memory and state-tracking mechanisms required for the strategy to navigate the session stages.

7.2 State Variables and Internal Memory

The implementation relies on persistent state variables to track the market's evolutionary stages throughout the session.

Range and Boundary Variables.

- *h* and *l*: Store the absolute highest and lowest prices observed during the opening range.
- *sigma*: Stores the standard deviation of prices during the initialization period.
- *Bu* and *Bl*: Define the adaptive upper and lower breakout boundaries.
- *barcount*: Tracks the number of one-minute bars accumulated since the range started.
- *Rangeready*: A Boolean flag indicating if the opening range is completed and frozen.

Protective-Closing Variables.

- *tsl_mode* and *RDD_mode*: Integer variables used to select the stop-loss and retracement multipliers.
- *TSL*, *RDD_factor*, and *TP_threshold_factor*: The decoded numerical working values for the exit engine.
- *SL*: The stop-loss distance in price units, scaled by the opening-range width.
- *TP_Active*: A logical switch indicating whether the trailing exit engine has been activated.
- *max_price* and *min_price*: References for the highest and lowest price reached since entry, essential for trailing logic.

7.3 Logical Rule Blocks

7.3.1 Daily Reset and Initialization

The *daily reset* rule acts as the session-level garbage collector. It triggers when the day of the week changes, resetting *barcount* to zero, *Rangeready* to *False*, and clearing values for *h*, *l*, *SL*, and *TP_Active* to ensure no data contamination between sessions. The initialization rule activates at 08:01 (when *barcount = 0*). It seeds the range by assigning *h* the value of *High[1]* and *l* the value of *Low[1]*, ensuring the range starts from a completed bar. During the accumulation phase (where *barcount* < 30), the strategy updates *h* and *l* as the rolling maximum and minimum of the initialization window.

7.3.2 Discrete Mapping of Parameters

When *barcount = 30*, the system executes mapping rules to transform integer optimization modes into numerical multipliers:

- **Stop-Loss (TSL):** Maps *tsl_mode* to values of 0.33, 0.66, 1.0, or 999999 (proxy for infinite/inactive stop).
- **Profit Protection Activation:** Maps to 0.33, 0.66, or 1.0. Notably, if the stop-loss is inactive (*tsl_mode = 3*), the activation threshold is capped at 1.0 to preserve dynamic protection.
- **Retracement Factor (RDD):** Maps *RDD_mode* to 0.33, 0.66, 1.0, or 999999.

7.3.3 Freezing the Range and Threshold Calculation

The *FreezeRange* rule triggers at 30 minutes to compute the operative thresholds. Volatility is calculated as:

$$\sigma = \text{StdDev}(\text{Close}, 30)[1]$$

The adaptive levels are defined as:

$$B_u = h + \epsilon_1 \sigma$$

$$B_l = l - \epsilon_2 \sigma$$

The stop-loss distance is then scaled:

$$SL = TSL \cdot (B_u - B_l)$$

Finally, *Rangeready* is set to *True* to allow entry signals.

7.3.4 Entry Conditions and Order Placement

A long signal requires *Rangeready* to be true, the previous bar high (*High[1]*) to exceed B_u, and the time to be at least 08:30. Short signals require *Low[1]* to fall below B_l. Orders are placed as market orders with the stop-loss set at $B_u - SL$ for longs or $B_l + SL$ for shorts. Duplicate trades are disabled. Upon entry, *max_price* (for longs) or *min_price* (for shorts) is initialized with the true executed order open price to ensure subsequent trailing calculations use real execution data.

7.3.5 Dynamic Exit Engine

The *Update_Max_price* rule continuously tracks the best favorable excursion achieved since entry. The *Activate_TP* rule then switches the strategy to retracement-based protection if unrealized profit satisfies:

$$\mathit{Close} - \mathit{OrderOpenPrice\_long} \geq \mathit{TP\_threshold\_factor} \cdot (B_u - B_l)$$

(Symmetric logic applies to short positions). The final *Exit_RDD* rule liquidates the position if the price retraces by the specified factor. For a long trade, the close condition is:

$$\mathit{Close} \leq \mathit{max\_price} - \mathit{RDD\_factor}\,(\mathit{max\_price} - \mathit{OrderOpenPrice\_long})$$

Once met, all positions are closed and *TP_Active* is reset to zero.

7.4 Visual Examples of Backtest Trade Executions

Figure 7.1: *Trade Execution: Basic ORB Strategy Parameters:* $\epsilon_1 = 0, \epsilon_2 = 0, TSL = 3, RDD = 3$.

This visual example (Figure 7.1) illustrates the most fundamental execution of the Opening Range Breakout (ORB) strategy. With threshold coefficients set to zero ($\epsilon_1 = 0, \epsilon_2 = 0$), the entry boundaries coincide perfectly with the high and low of the initial 30-minute range, without any volatility-based adjustment. Furthermore, the deactivation of protective closing mechanisms ($TSL = 3, RDD = 3$) results in a "naive" market exposure; once the breakout is triggered, the position is maintained throughout the session regardless of intraday fluctuations, concluding only with the mandatory liquidation at the market close.

Figure 7.2: *Trade Execution: Evolutionary ORB with SL/TP Parameters:* $\epsilon_1 = 1, \epsilon_2 = 1, TSL = 2, RDD = 2$.

In contrast, the second trade (Figure 7.2) demonstrates the advanced evolutionary logic of the *GAORB* model. By applying threshold coefficients of one ($\epsilon_1 = 1, \epsilon_2 = 1$), the entry levels are dynamically adjusted outward by adding the standard deviation of the opening range to the boundaries. Additionally, this execution showcases the dynamic take-profit mechanism in action ($RDD = 2$). Instead of waiting for the session end, the system monitors the price retracement from its intraday high-water mark; once the price pulls back by the specified ratio, the position is closed proactively to lock in realized gains before a potential trend reversal.

The next stage of the appendix documents how these logical rules are translated into an evolutionary search within the StrategyQuant X software.

7.5 How the optimized parameters are defined in the SQX optimizer

Once the GA is deployed as in Wu et al. (2021) work, the parameters selected for optimization include *RDD_mode*, *epsilon1*, *epsilon2*, and *tsl_mode*, which correspond directly to the protective-closing and threshold-adjustment variables of the reference framework. The numerical ranges used in SQX mirror the economic structure of the paper. The variables *epsilon1* and *epsilon2* are allowed to vary between -1 and 1 in steps of 0.03125, which is equivalent to increments of 1/32 and therefore directly aligned with the resolution used by Wu et al. (2021) The variables *tsl_mode* and *RDD_mode* take integer values from 0 to 3, which are then decoded in the strategy logic into the discrete multipliers already described in Section 5.2.2. Thus, even though SQX presents the problem through parameter tables rather than through binary chromosomes, the underlying economic search space remains very close to the one described in the paper.

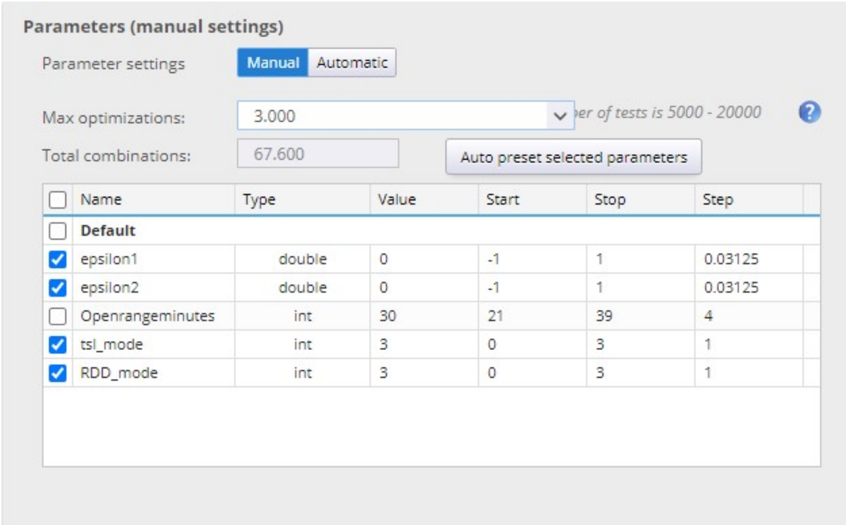

Figure 7.3: *Parameter selection in SQX*

7.6 Fitness ranking inside SQX.

The final bridge between the reference methodology and the platform implementation concerns the ranking criterion. StrategyQuant allows the user to choose the strategy-quality metric used during optimization, including alternatives such as net profit, return-to-drawdown measures, expectancy-based metrics, annual return relative to maximum drawdown, etc, but not a multi-objective weighted fitness. In the setup shown here, the selected ranking criterion is *Net Profit (Return)* and *Sharpe ratio.*

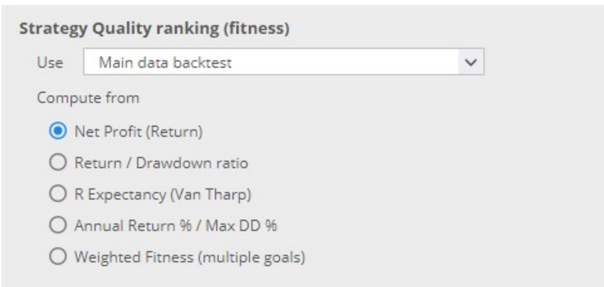

Figure 7.4: *Fitness selection by Net Profit in SQX*

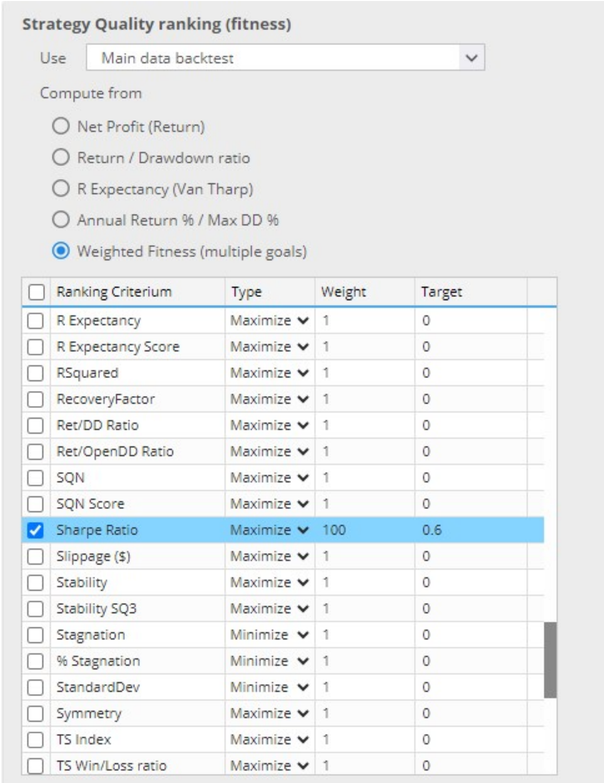

Figure 7.5: *Fitness selection by Sharpe Ratio in SQX*

7.7 Technical Configuration of the Walk-Forward Engine in StrategyQuantX

The walk-forward methodology described in the dissertation is executed using the *Walk-Forward optimization* mode within the StrategyQuant X *Optimizer*. The platform automates the division of historical data into repeated segments, ensuring that the held-out (OOS) portion always follows the training (IS) portion immediately. This configuration produces a concatenated equity curve that reflects the strategy's performance as parameters are updated progressively through time. In the specific settings used for this project, the following technical choices were made within the SQX interface:

- **Walk-Forward Type:** The software was configured to use a fixed rollover structure rather than a floating window.
- **In-Sample/Out-of-Sample Split:** A time-based split was defined to match the 60/30 day cycle (2:1 ratio) required to replicate the reference framework.
- **Concatenation:** SQX automatically joins the resulting OOS segments to report a single, realistic performance report of the adaptive system.

By leveraging these internal platform tools, the study ensures that the parameter selection process is conducted systematically, avoiding the manual errors associated with traditional backtesting methods.

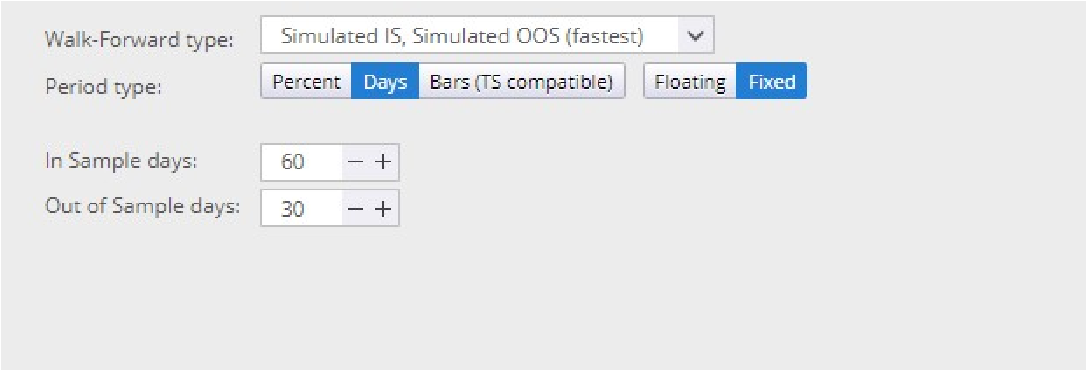

Figure 7.6: *Walk-Forward parametrization in SQX*

7.8 Visual representation of the 2D Results

The following visual section provides the documentation of the 2D optimization charts, showcasing the convergence of parameters for the Naive, SL, and Complete models under both Net Profit and Sharpe Ratio objectives:

Figure 7.7: *Optimization for ϵ_1 by Net profit fitness, -1 gave the highest net profit*

Figure 7.8: *Optimization for ϵ_2 by Net profit fitness, -1 gave the highest net profit*

Figure 7.9: *Optimization for ϵ_1 by Sharpe ratio fitness, -1 gave the highest net profit*

Figure 7.10: *Optimization for ϵ_2 by Sharpe ratio fitness, -1 gave the highest net profit*

Figure 7.11: *Optimization for ϵ_1 by Net profit fitness, -0.9375 value gave the highest net profit*

Figure 7.12: *Optimization for ϵ_2 by Net profit fitness, -1 value gave the highest net profit*

Figure 7.13: *Optimization for tsl_mode by Net profit fitness, 3 value gave the highest net profit*

Figure 7.14: *Optimization for* ϵ_1 *by Sharpe ratio fitness, 0.90625 value gave the highest sharpe ratio*

Figure 7.15: *Optimization for* ϵ_2 *by Sharpe ratio fitness, -0.65625 value gave the highest Sharpe ratio*

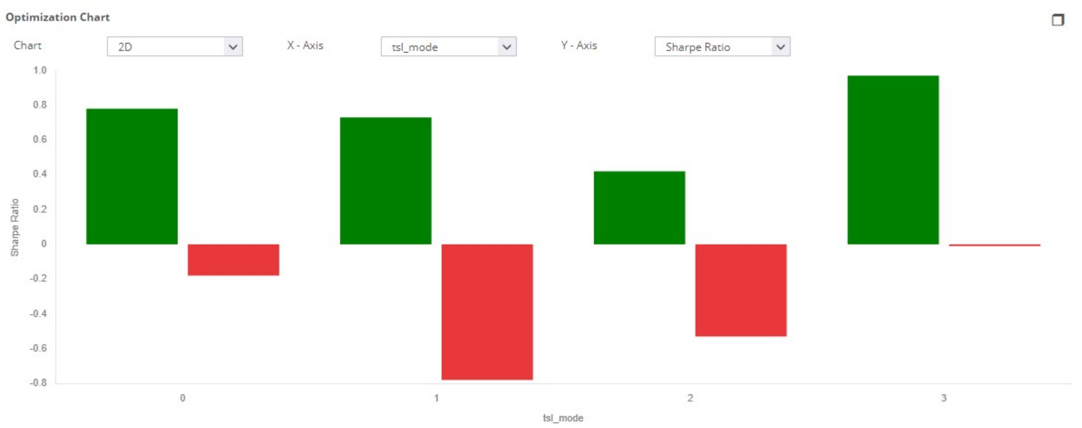

Figure 7.16: *Optimization for tsl_mode by Sharpe ratio fitness, 3 value gave the highest Sharpe ratio*

Figure 7.17: *Optimization for ϵ_1 by Net profit fitness, 0.59375 value gave the highest Net profit*

Figure 7.18: *Optimization for ϵ_2 by Net profit fitness, -0.46875 value gave the highest Net profit*

Figure 7.19: *Optimization for tsl_mode by Net profit fitness, 0 value gave the highest Net profit*

Figure 7.20: *Optimization for RDD_mode by Net profit fitness, 2 value gave the highest Net profit*

Figure 7.21: *Optimization for ϵ_1 by Sharpe ratio fitness, 0.125 value gave the highest Sharpe ratio*

Figure 7.22: *Optimization for ϵ_1 by Sharpe ratio fitness, -0.53125 value gave the highest Sharpe ratio*

Figure 7.23: *Optimization for tsl_mode by Sharpe ratio fitness, 3 value gave the highest Sharpe ratio*

Figure 7.24: *Optimization for RDD_mode by Sharpe ratio fitness, 0 value gave the highest Sharpe ratio*

7.9 Visual representation of the 3D Results

We conclude with the multidimensional 3D synergies, identifying the robust zones of convergence where entry and exit parameters intersect:

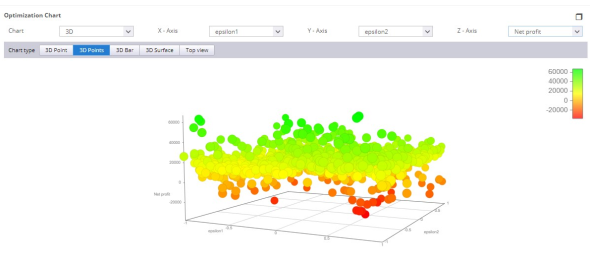

Figure 7.25: *Optimization for ϵ_1 and ϵ_2 by Net Profit.*

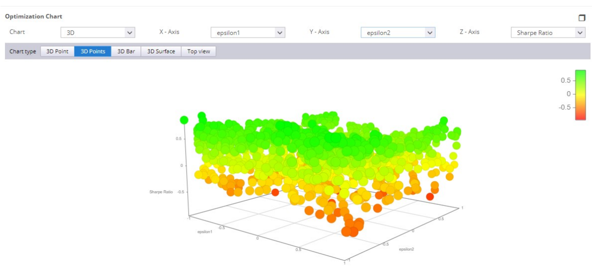

Figure 7.26: *Optimization for ϵ_1 and ϵ_2 by Sharpe Ratio.*

When we project TSL_mode and RDD_mode against our primary objectives—Net Profit and Sharpe Ratio—the 3D visualization provides a remarkably consistent picture of market resilience.

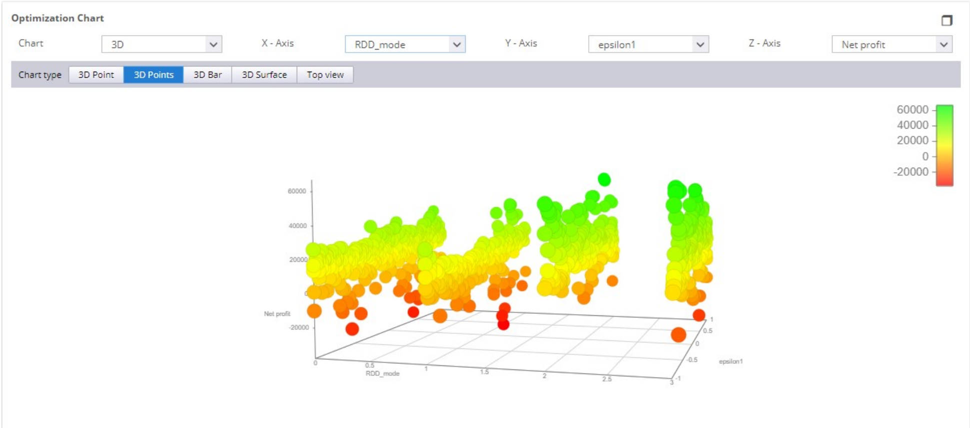

Figure 7.27: *Optimization for* ϵ_1 *and RDD by Net Profit.*

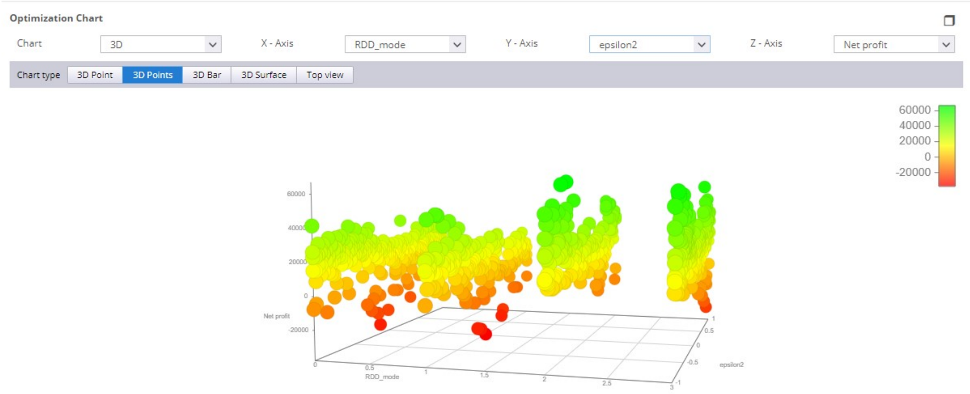

Figure 7.28: *Optimization for* ϵ_2 *and RDD by Net Profit.*

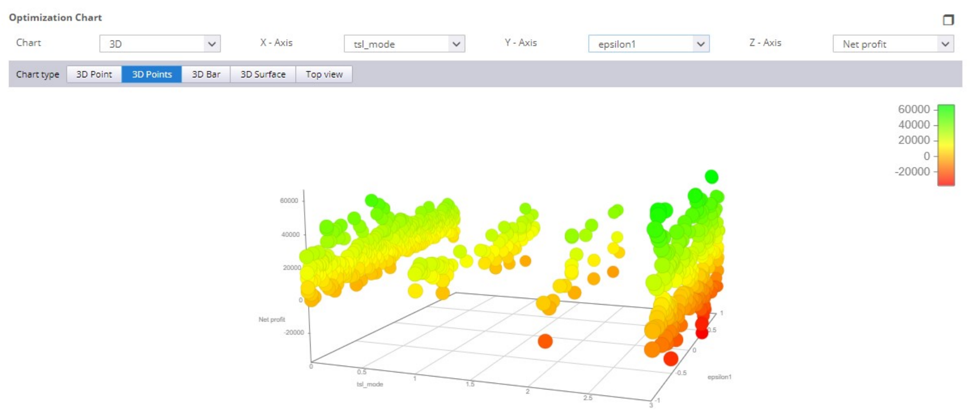

Figure 7.29: *Optimization for* ϵ_1 *and TSL by Net Profit.*

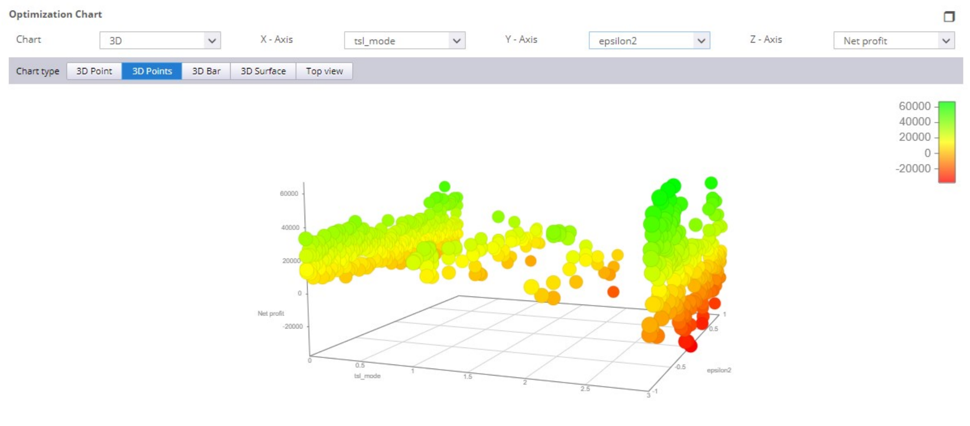

Figure 7.30: *Optimization for ϵ_2 and TSL by Net Profit.*

Figure 7.31: *Optimization for TSL and RDD by Net Profit.*

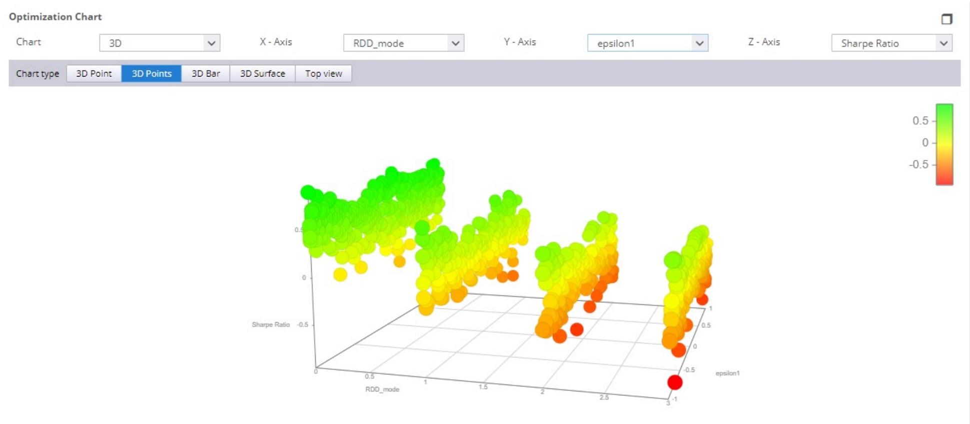

Figure 7.32: *Optimization for ϵ_1 and RDD by Sharpe Ratio.*

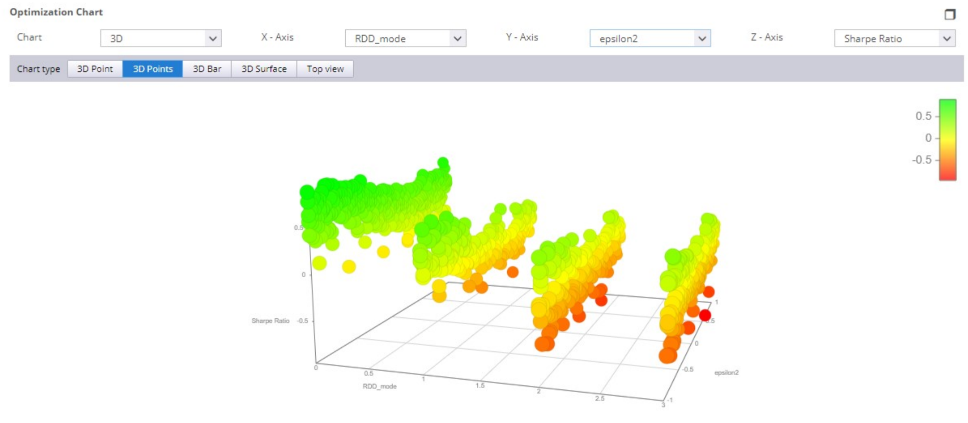

Figure 7.33: *Optimization for ϵ_2 and RDD by Sharpe Ratio.*

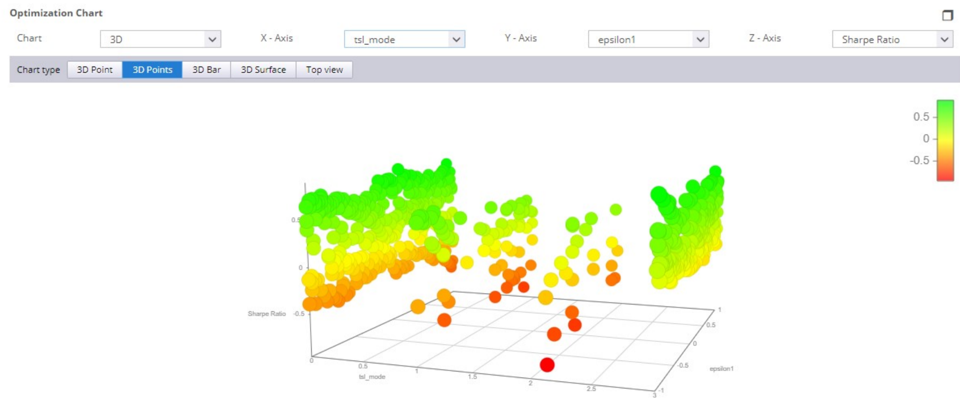

Figure 7.34: *Optimization for ϵ_1 and TSL by Sharpe Ratio.*

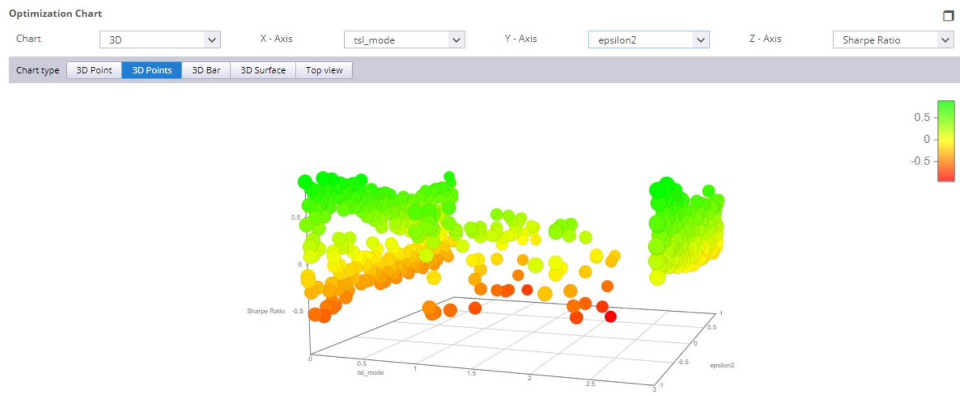

Figure 7.35: *Optimization for ϵ_2 and TSL by Sharpe Ratio.*

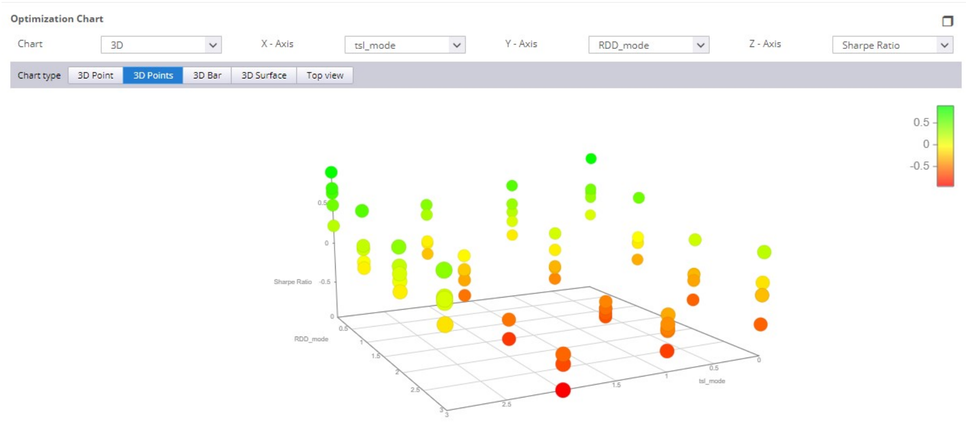

Figure 7.36: *Optimization for TSL and RDD by Sharpe Ratio.*

7.10 Repositories of the strategy logic

You can download the strategy in the following GitHub repository. You will be able to download it in the format .sqx, .txt, .mt4, .mt5 and .java.