Multi-Objective Optimization and Genetic Programming of Technical Indicators for Trading

1st Joseph Adriel Solichin

Computer Science Department, School of Computer Science

Bina Nusantara University

Jakarta, Indonesia 11480

joseph.solichin@binus.ac.id

2nd Olivia Salma Syahid

Computer Science Department, School of Computer Science

Bina Nusantara University

Jakarta, Indonesia 11480

olivia.syahid@binus.ac.id

3rd Dr. Aditya Kurniawan, S.Kom., MMSI., CND, CEHmaster

Computer Science Department, School of Computer Science

Bina Nusantara University

Jakarta, Indonesia 11480

adkurniawan@binus.edu

4th Chrisando Ryan S.Kom, M.Kom

Computer Science Department, School of Computer Science

Bina Nusantara University

Jakarta, Indonesia 11480

chrisando.siahaan@binus.ac.id

Abstract— **This research studies the integration of Multi-Objective Optimization (MOO) and Genetic Programming (GP) in trading strategies using Relative Strength Index (RSI) and Slow Stochastic (SS) indicators. The MOO agent in our method is the niched-Pareto Genetic Algorithm (NSGA-II). Our research attempts to maximize risk-adjusted returns as well as cumulative returns by applying a multi-objective approach. Comparing the use of MOO and GP to conventional single-objective techniques, empirical data show a considerable improvement in strategy performance. Our results highlight how financial trading techniques can be improved by combining MOO and GP, possibly yielding larger returns at levels of controlled risk.**

Keywords— Multi-Objective Optimization, Genetic Programming, Technical Indicators, Trading Strategies, Cumulative Return, Risk-Adjusted Return

I. INTRODUCTION

Adopting a multi-objective optimization approach can improve the performance of GP strategies in automated trading. Various empirical works have been done in the literature on the application of Genetic Programming to financial markets gaining a 28% increase in portfolio value and thus leading to better prediction with technical Indicators like the MACD, EMA, and RSI [1]. Another study explored the use of multi-objectives optimization approach to increase the performance of the Genetic Programming-based strategies and shows that cumulative return is also improved from 14.30% to 62.05% of which much better than single objective strategy [3]. While there is no wanting of research studies on technical and directional change indicators, other indicators - like RSI and Slow Stochastic - have not received the same level of attention.

Developed by J. Welles Wilder in 1978, Relative Strength Index (RSI) is a momentum oscillator that measures the speed and change of price movements of a security. How does it work: Individual RSI is defined as the ratio of average upward moves and average downward moves. If RSI is above 70 it shows overbought market and below 30 shows oversold market. It is bound between 0 and 100, where any value above 70 indicates an overbought condition, resulting in a possible sell signal, while any value below 30 indicates an oversold condition and a possible buy signal. RSI assists traders and analysts in determining future price movements of securities by informing as to how strong current market conditions are, whether an asset is under or over-performing the market, thus helping take part in decision makings.

On the other hand, Slow Stochastic Oscillator (SS) is a technical indicator that functions much like RSI in that it is used to spot potential turning points by evaluating if a security's closing price is close to its price range over a predefined time span. Originating from the Stochastic Oscillator developed by George Lane in the 1950s, the Slow Stochastic is derived by smoothing the %K line with a 3-day simple moving average, thus forming the %D line and reducing the volatility and sensitivity of the original indicator. This indicator oscillates between 0 and 100, with levels above 80 typically signaling that the asset is overbought and levels below 20 indicating that it is oversold. By providing insights into the momentum and the potential endurance of an asset's current direction, the Slow Stochastic helps traders identify opportunities where the price momentum might be slowing down or picking up, suggesting possible entry or exit points in the market.

Building on the work of [1][3], we investigate whether multi-objective programming and genetic programming can further improve the cumulative return (high return and minimal risk) by using RSI and Slow Stochastic, and comparing them.

II. Literature Review and background

In this section, we will define our research scope. Then, some relevant previous works will be presented

A. Research Scope

Our work is about trading strategy with Multi-Objective Optimization and Genetic Programming-based strategies. In the financial field [2], Multi-Objective Optimization is used to identify significant patterns of technical analysis in the financial time series. Two objectives are examined, which are the quality of matches and area. The niched-Pareto Genetic Algorithm (NSGA-II) is used to determine the appropriate sharp interval for the downtrends, uptrends, and head-and-shoulders.

Genetic-Programming algorithm is inspired by nature [1]. Based on Britannica articles, Genetic Algorithm is a type of evolutionary computer algorithm in which symbols representing possible solutions are "bred". The GP algorithms are sometimes used in research with artificial life, cellular automatons, and neural network.

In the 1990s, the field of GP was rejuvenated by John Koza. Since then, a particular feature of the GP literature has been a strong interest in real-world problem domains. One domain which has delighted significant attention is financial and economics. A foundation of GP is its ability to interact with both the solution form and relevant solution parameters. As a result, GP can simultaneously create new program (solution) and also optimise their parameter s. This offers specific benefit in finance and economics as we often lack strong theoretical models which well-explain phenomena of interest.[8]

So, by combining the Multi-Objective Optimisation and Genetic Programming we can get the higher return of cumulative return.

B. Related Work

The multi-objective GP proposed in [1] optimises the condition of DC and technical indicators based on the market with fixed indicators parameters. Contrary to them, our GP optimises the parameters of indicators with a fixed market condition. While both use Pareto Front (NSGA-II) [5] as the multi-objective optimization agent.

III. Methodology

This section presents the proposed algorithm and the trading strategy that we used. For the proposed algorithm, we describe the terminal set and the function set we use, we discuss how the model is represented, we define the fitness function we use and also discuss details of GP.

A. Multi-objective Genetic Programming Model

1) Terminal set:

The terminal set comprises the terminals of the GP tree and relies on the input data that will be used during

GP training. We used 2 technical indicators; they were RSI and SS.

2) Function set:

The function set is quite simple, two logical operators, namely AND and OR, four relational operators, namely less than (<), greater than (>), less than or equal (≤), and greater than or equal (≥).

3) GA model representation:

Starting with an initial population, each individual undergoes selection, crossover, and mutation processes. In selection, a hybrid approach is used combining tournament selection and NSGA-II to maintain genetic diversity and select the most fit individuals. Crossover (with a probability *cxpb*) and mutation (with a probability *mutpb*) are applied to create offspring, introducing variation and exploring new genetic combinations.

Individuals with outdated or invalid fitness are re-evaluated to ensure updated performance metrics. The parameters including *cxpb* and *mutpb* will be discussed in Section V-C. Algorithm 1 presents the GA model used.

Algorithm 1 Genetic Algorithm

Require: Initialise variables (*ngen* represents the amount of generation)
for each generation from 1 to *ngen*
  //Selection process to create offspring
  offspring <- hybrid_selection(population, length of population)
  //Clone the selected individuals to prepare for crossover and mutation
  offspring <- clone(offspring)
  //Crossover operation on offspring pairs
  for each pair (child1, child2) in offspring taken two at a time
    if random chance < *cxpb* **then**
      mate(child1, child2)
    Invalidate fitness of child1 and child2
    end if
  //Mutation operation on offspring
  for each mutant in offspring
    if random chance < *mutpb* **then**
    mutate(mutant)
    Invalidate fitness of mutant
    end if
  //Re-evaluate fitness for individuals with invalidated fitness
  invalid_ind <- list of offspring with invalidated fitness
  for each ind in invalid_ind
    ind.fitness <- evaluate(toolbox, ind)
  //Update population with offspring
  population <- selectNSGA2(all individuals in population + offspring, length of population)
return population

---

4) Fitness function:

In this GP trading strategy, we use two fitness function: cumulative return and risk-adjusted return that is defined as:

$$RAR = \frac{\sum_{i=1}^{n}(T[i] - \frac{r_f}{1512})}{n \cdot \sigma_d}$$

$$CR = \sum_{i=1}^{n} T[i]$$

The formula above will only be processed if length of $T > 0$, and $\sigma_d > 0$. Where T represents the array of raw returns from trades; r_f represents the annual risk-free, set by default at 0.02 (2%) divided by 1512, since there are 252 days and 6 hours per day of active trading sessions; n is the number of trades in T; RAR is calculated by dividing mean excess returns by downside returns. The model essentially computes a version of Sortino Ratio that measures of return per unit of downside risk. The cumulative return is calculated by summing all of the trade returns.

5) Multi-objective optimisation

We rely on NSGA-II (Non-dominated Sorting Genetic Algorithm) to evaluate more than one fitness function. In NSGA-II, the population is first sorted based on the level of dominance, grouping solutions into different fronts. The first front is entirely non-dominated and is considered closest to the ideal Pareto front. Subsequent fronts have increasing levels of dominance. To maintain diversity, NSGA-II also calculates the crowding distance, which measures how close an individual is to its neighbours, favouring those with a larger distance to prevent clustering of solutions.

The final result of an NSGA-II process is a population of solutions that approximates the true Pareto front as closely as possible.

TABLE I

CONFIGURATION OF THE GP ALGORITHM

Configuration	Value
Terminal Set	Ephemeral Random Constant (ERC), Slow Stochastic(SS), Relative Strength Index(RSI)
Function Set	AND, OR, $<$, $>$, $\leq$,$\geq$
Genetic Operators	Two-point crossover, mutation
Selection	NSGA-II(Pareto Front), Tournament

B. Trading Strategy

The trading strategy integrates the use of the Relative Strength Index (RSI) with predefined profit and loss thresholds to govern entry and exit decisions, similar to the approach used with the Slow Stochastic Oscillator. For RSI, buy signals are triggered when RSI value drops to below 30, indicating the asset might be oversold. Sell signals are issued under two conditions: if the RSI climbs above 70, suggesting overbought conditions, or if the returns either surpass the take-profit (erc_tp) threshold or drop below the stop-loss (erc_sl) limit.

For SS, buy signals are generated when %K crosses above %D while under 20, indicating an oversold condition. Sell signals are triggered when %K crosses below %D above 80, showing an overbought condition, or when trades meet predefined profit or stop-loss thresholds.

The number of concurrent open positions is limited to three to manage risk effectively. By the design, we remark that selling an asset that we don't own is not possible (short-selling is not permitted, since gold is fundamentally a bullish asset.).

This setup aims to capture profits by exploiting short-term market trends and reversals while simultaneously protecting against significant losses. Profit or loss on each trade is calculated as the percentage change in price from the point of entry to the exit, allowing the strategy to dynamically adjust to evolving market conditions and optimise the risk-return profile of the trading portfolio.

IV. EXPERIMENTAL SETUP

In this section, we will present the experimental setup including the data, benchmarks, and parameters.

A. Data

The GP trading strategy is applied to the hourly historical data of gold spot for the last one year (XAUUSD).

B. GP Parameters

Table II shows the parameters used in the GA Algorithm.

TABLE II
PARAMETERS OF THE GP ALGORITHM

Parameters	Value
Generation	50
Population size	500
Crossover probability	0.95
Mutation probability	0.5
Tournament size	3
ERC	0 to 1

V. RESULT AND ANALYSIS

TABLE III
ONE OF THE PROGRAM'S RESULT

Measurement	Cumulative Return		Risk-adjusted Return	
Indicator	RSI	SS	RSI	SS
Average	20.8%	14.56%	1.07	10.21
Minimal	11.68%	14.55%	1.05	10.21
Maximal	20.92%	14.56%	3.07	10.22

Measurement	RSI	SS
Parameter	RSI Period: 12	%K Period: 22, %D Period: 9
TP Percentage	0.65%	0.58%
SL Percentage	0.65%	0.71%

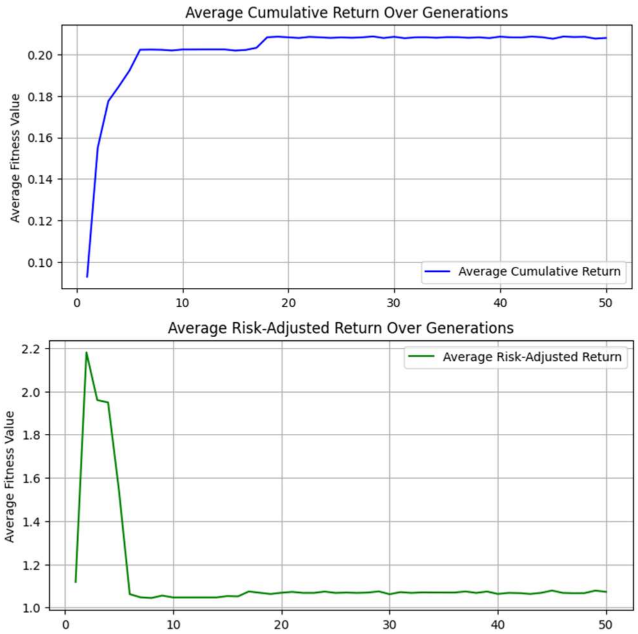

Fig. 1. A result of RSI-based program

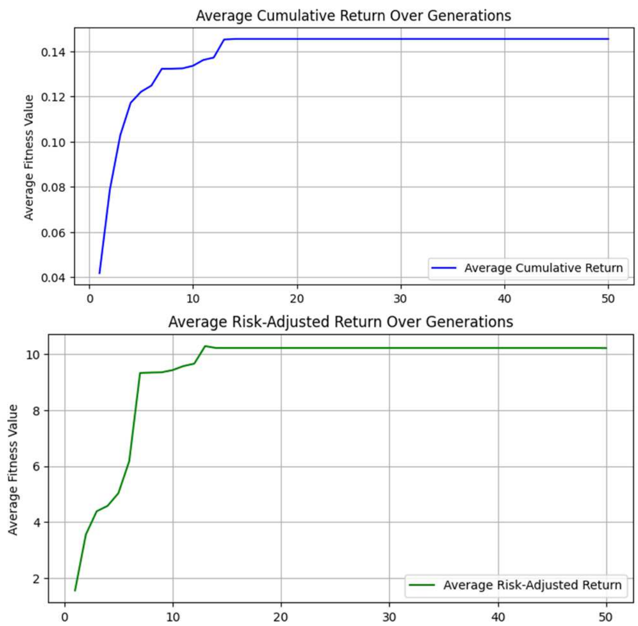

Fig. 2. A result of SS-based program

Analysing from the Table III results, we can see that the indicators perform opposite results. RSI-based results a higher average cumulative return than SS-based, 20.8% to 14.56%. On the contrary, RSI-based average RAR result is much lower than SS-based, 1.07 to 10.22. The difference almost reached 1000%. Here, we find that SS has a greatly more downside return. We suspect this is caused by the negative result of risk-reward (tp = 0.58%, sl = 0.71%), producing more chance and space for big drawdown.

Looking at the results in Fig. 1. And Fig. 2., we observe distinct patterns. Average cumulative return for both RSI and SS shows a similar parabolic pattern indicating a better solution in most iterations. On the contrary, average risk-adjusted return shows a unique graph for each indicator. RSI-based GP presents an early spike after initialization followed by stabilisation, suggesting that the GP iterates towards more balanced solutions.

While 20% does not seem much, we need to remember that XAU/USD has a smaller change percentage per tick compared to other assets. 1 pip in XAU/USD is $0.01 so a 1% increase at $2000 per ounce means 2000 pips increase. As a benchmark, 1 pip in S&P500 in $0.1 so a 1% increase at $5000 means 500 pips increase. Here we can see that a 1% move in gold roughly equals to four times in pip movement of S&P500. We also need to note that, the trading model we implemented does not use leverage or margin. Thus, using one will greatly increase these numbers.

In summary, the result shows a massive improvement on optimising RSI and SS indicators.

VI. CONCLUSION

As a summary, our study offers a thorough investigation into the effectiveness of MOO and GP for trading strategies with RSI and SS technical indicators. The results show that by optimizing for both cumulative returns and risk-adjusted returns, MOO and GP has greatly improved the trading strategy. While the RSI-based GP strategies showed a more dynamic evolution, indicating varying degrees of market adaptation and risk profiles, the SS-based GP strategies were able to rapidly produce steady return profiles. The study confirms the usefulness of momentum indicators such as RSI and SS in identifying market trends and reversals, but it also emphasizes the subtle variations in how these indicators exhibit predictive power when genetically programmed.

REFERENCES

[1] Santur, Yunus; Ulas, Mustafa; Karabatak Murat: Genetic Algorithm Based Approach for Algorithmic Trading in Financial Markets (2023)

[2] Rink, Kevin: The Predictive Ability of Technical Trading Rules: An Empirical Analysis of Developed and Emerging Equity Markets (August, 2023)

[3] Long, Xinpeng; Kampouridis, Michael; Kanellopoulos,Panagiotis: Multi-Objective Optimization and Genetic Programming for Trading by Combining Directional Changes and Technical Indicators (2023)

[4] Christodoulaki, Eva; Kampouridis, Michael; Kanellopoulos, Panagiotis : Technical and Sentiment Analysis in Financial Forecasting with Genetic Programming (2022)

[5] Chen, Yi; Zhou, Aimin: Multiobjective Portfolio Optimization via Pareto Front Evolution(2021)

[6] Ayala, J., García-Torres, M., Noguera, J. L. V., Gómez-Vela, F., & Divina, F.: Technical Analysis Strategy Optimization Using a Machine Learning Approach in Stock Market Indicies (2021)

[7] Brabazon, Anthony; Kampourdis, Michael; O'Neill, Michael: Applications of Genetic Programming to Finance and Economics: Past, Present, Future (2019)

[8] Faijareon,Chawwalit; Sornil, O.: Evolving and Combining Technical Indicators to Generate Trading Strategies (2019)

[9] Abuselidze, G., Slobodianyk, A., & Reznik, N.: Ensuring Trading Strategies Profitability with Technical Analysis Tools and Computer Technology (2019)

[10] Gunantara, Nyoman: A Review of Multi-Objective Optimization: Methods and Its Applications (2018)

[11] Long, X., Kampouridis, M., Kanellopoulos, P.: Genetic programming for combining directional changes indicators in international stock markets. In: International Conference on Parallel Problem Solving from Nature. pp. 33–47. Springer (2022)

[12] Lee, J., Sabbaghi, N.: Multi-objective optimization case study for algorithmic trading strategies in foreign exchange markets (2019)

[13] Martins, T., Neves, R.: Applying genetic algorithms with speciation for optimization of grid template pattern detection in financial markets (2020)

[14] Tran, M., Pham-Hi, D., Bui, M.: Parameter Optimization for Trading Algorithms of Technical Agent