Bitcoin Optimized Signal Allocation Strategies using Decomposition

Sherin M.Omran[1*], Wessam H. El-Behaidy[2], Aliaa A. A. Youssif[3]
Department of Computer Science, College of Computers and Artificial Intelligence, Helwan University, Cairo 11795, Egypt[1, 2]
College of Computing and Information Technology, Arab Academy for Science, Technology & Maritime Transport (AASTMT), Smart Village 12577, Egypt[3]

***Abstract*—Bitcoin is the first and most famous cryptocurrency. It is a virtual currency that is operated in a decentralized form using cryptographic strategies called blockchains. Although it has experienced significant market acceptance by traders and investors in recent years, it also suffers from volatility and riskiness. Technical analysis is one of the most powerful tools used for trading signals' allocation using some algorithmic strategies called technical indicators. In this research, a newly proposed multi-objectives decomposition-based particle swarm optimization algorithm is used to find the best parameter values for some technical indicators, which in turn generates the best trading signals for Bitcoin trading. In this context, three conflicting objectives have been used, i.e., the return on investment, the Sortino-ratio, and the number of trades. The proposed algorithm is compared to the original MOEA/D algorithm as well as the indicators using their original parameters. Results showed the superiority of the proposed algorithm during the training and testing periods over the other benchmarks.**

Keywords*—*Bitcoin; technical analysis; decomposition; particle swarm optimization; MOEA/D

I. Introduction

Unlike the well-known physical currencies such as dollars or euros, bitcoin is a sort of digital currency backed by cryptographic protocols called blockchains [1]. These cryptographic protocols facilitate secure online payment with no need for intermediaries. There exist significant fluctuations in the bitcoin prices every day which increases its volatility and raises the level of risk [2]. This volatility enormously affects the traders' outcomes. So, powerful algorithms are always needed to help the traders and investors get the best returns with the minimum level of risk.

The algorithmic trading strategies for cryptocurrencies can be classified into three different models, i.e., models based on machine learning, Portfolio Optimization (PO) models, and trading-strategies optimization models.

Machine learning and deep learning are used in the literature as predictive models to forecast the future price patterns based on the analysis of historical market data [3], [4], [5], and [6].

PO in the context of cryptocurrencies entails selecting a combination of cryptocurrencies and determining the appropriate weights for each asset in order to achieve the desired portfolio characteristics. Depending on the preferences of the investor, the objective may be to maximize returns, minimize risk, or achieve a specific risk-return trade-off [7]. Portfolio optimization researches can be categorized into statistical models such as Modern Portfolio Theory (MPT) as found in [8], [7], and [9] and Multi-Objective Evolutionary Algorithms (MOEA) models such as [10], and [11].

The trading strategies optimization models mainly aim to allocate the optimal trading signals that enhance the trading outcomes. In study [12] a signal herding model is proposed in order to enhance the decision-making process. In study [13] the trading signals are generated based on the market tweets sentiments whereas in study [14], both time series analysis and social signals are used to produce tractable trading strategies.

Trading signals generation or allocation can be achieved using Technical Analysis (TA) based indicators. Although the optimization of the TA based trading-strategies was found in the literature for other types of markets such as physical currencies and stock markets, it was not found for cryptocurrencies. So, this research tries to cover the lack of this research point by proposing new optimized algorithmic trading-strategies for some Technical Indicators (TI).

As can be seen previously, cryptocurrency trading can be considered as a Multi-objective Optimization (MO) problem. As it involves handling a set of conflicting objectives simultaneously such as maximizing the returns and the percentage of the profitable trades to the non-profitable ones, minimizing risks, the transaction costs, and the number of trades, etc. [15].

The MO problem is described as in Eq. (1) [16]:

$$\text{Μαξιμιζε } F(x) = (f_1\,(x)\,, \dots \dots f_m(x))^T \qquad (1)$$

subject to $x \in \Omega$

Such that: Ω is the variable or decision space, F: $\Omega \rightarrow R^m$ is the objective space, where *m* is the total number of objectives. As there are multiple contradictions between the different objectives, there can never be a unique solution that satisfies all the objectives simultaneously but rather a set containing the whole non-dominated solutions referred to as Pareto Set (PS). Let $x_1, x_2 \in R^m$, it can be said that x_1 dominates x_2 only if $x_1^i \geq x_2^i$ for each $i \in \{1, 2, \dots, m\}$ and $x_1^j > x_2^j$ at least once for $j \in \{1,2, \dots, m\}$. The set of the objective vectors corresponding to the points in the PS is called the Pareto Front (PF).

MOEA based on Decomposition (MOEA/D) is one of the most simple and efficient techniques to solve MO problems. It can efficiently find the best set of non-dominated solutions regardless of the increasing number of objectives [17], [18]. On the contrary, MOEA based on Pareto dominance can efficiently handle problems with lower number of objectives, however by increasing the number of objectives it can hardly cover the entire (PF) [18].

MOEA/D transforms the MO problem into a simple set of scalar sub-problems then, it solves each of them simultaneously and independently. This transformation is achieved with the help of two basic factors, i.e., aggregation or also called Scalarization Functions (SF) and a set of well-selected weight vectors. These are the basic factors that control the performance of MOEA/D algorithms.

New decomposition strategies were found in the literature either by proposing new SFs as in [19], and [20] or by using a combination of different SFs [21], and [22].

Some changes to the original weight generation mechanism that was originally proposed by Zhang [17] were also found in the literature such as MOEA/D-URAW [23], AWD-MOEA/D [24] and MOEA/D-AWG [25].

The MOEA/D algorithms were successfully applied to different application areas such as PO [11], [26], image segmentation [27], and network routing [28].

In this paper a new MO Particle Swarm Optimization Algorithm using Decomposition (MOPSO/D) is proposed for BTC trading signals' allocation. The algorithm optimizes the original parameters of three of the most used TIs. A new weight generation strategy was also presented and used in order to further improve the performance of the algorithm. The proposed algorithm is compared to the original MOEA/D and the TIs with their original parameters based on three objectives, i.e., the return, the risk, and the number of trades.

The rest of this paper will be organized as follows: In Section II, the trading signals allocation mechanisms are described. Section III shows the proposed algorithm, whereas Section IV presents the results and the conclusion in Section V.

II. Trading Signals Allocation Mechanisms

Cryptocurrencies are well-known for their extreme volatility, which refers to the huge and sudden price changes they encounter over short periods of time. This nature raises the riskiness level of such markets, making them hardly predictable. So, for the investors to get the benefits from crypto-market trading, there should be powerful algorithms that ensure the profitability and safety of their trading. As seen in Fig. 1, the Bitcoin value started in 2017 at about 1,000 USD and has risen to 20,000 USD. Then, it dropped back again, up to 4,000 USD by 2019.

TA tools (specifically the Technical Indicators TA) are frequently used by investors for their simplicity. They are used either as the basic trading strategies or as confirmation tools to generate and allocate the trading signals, i.e., buy and sell. For example, most PO tools basically depend on them in order to get their final trading decisions [29] and [30].

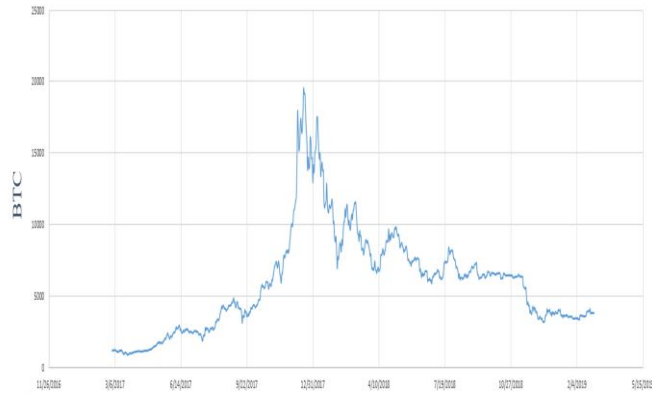

Fig. 1. The value of Bitcoin (BTC) in USD from 2017 to 2019.

A. Technical Indicators (TIs)

The TIs are mathematical formulations (equations) that are operated in an algorithmic manner. Each of these indicators has its own parameter combinations that controls its final performance [31]. The original values of these parameters are generated by their creators, however with more challenging markets such as the crypto market it could not always provide the required performance . The main objective of this study is to find the optimal set of parameters that helps allocating the best trading signals (buy and sell).

Three TIs are considered in this study, i.e., Double Weighted Moving-Average (DWMA), Exponentially Smoothed Rate of Change (ES-RoC) and Stochastic Relative Strength Index (S-RSI).

The DWMA indicator in [32] is one of the versions of Moving Average indicators (MAs). This type of indicators is used to generate an updated price average based on the selected time frame. The term double here means that the signals are generated based on the crossovers between two WMAs with different time frames.

A buy signal is provided when the shorter WMA crosses from below to above the longer WMA and vice versa for the sell signal. The WMA is calculated as seen in Eq. (2), such that n is the number of days used for calculations and P refers to the daily prices. For example, P_1 is the closing price for the first day of calculations and P_n.is the most recent price value at day n. The DWMA has two parameters n_1 and n_2 for the short and long WMAs consequently, such that the original parameters values are {20,50}.

$$WMA = \frac{(P_1 * n) + (P_2 * n-1) + \cdots + (P_{n-1} * 2) + \square(P_n)}{\frac{[n*(n+1)]}{2}} \quad (2)$$

The ES-RoC [31] indicator calculates the rate of change for the values of the Exponential MA (EMA) values over the past n days. The EMA is a type of MAs where the weighting for the days of calculations grows exponentially rather than the linear weighting as in WMA. The range of the outputs of this indicator is always in the range ±100.

The signals are generated by crossing above or below the center line (which is zero in this case) for buy and sell signals in sequence without considering a definite overbought or oversold level. This indicator is mainly used to evaluate the strength of the upcoming trends. The ES-RoC is calculated as shown in Eq. (3), where $EMA_{current}$ is the value of the EMA on the last day of calculation and EMA_n is the value of the EMA *n*-days ago.

In this study, two extra parameters, i.e., overbought and oversold levels are produced for trading signals generation. Rather than crossing above or below zero line, the buy and sell signals are generated by crossing through the oversold and overbought levels for buy and sell signals consequently. The original parameter values for ES-RoC are 14 and 20 days for the RoC and the EMA sequentially.

$$ES_{RoC} = \frac{EMA_{current} - EMA_n}{EMA_n} \times 100 \tag{3}$$

S-RSI [31] is a mixed indicator that applies the Stochastic-Oscillator (SO) indicator to the RSI value. RSI is another indicator that analyses current price levels against those of the recent past. The output values lie in the range from 0 and 100 that is used to identify the oversold and overbought levels.

The signals are generated by the crossovers with these levels the same way as the ES-RoC. The value of S-RSI [31] is calculated as shown in Eq. (4). Such that *n* and *t* are the timespans for RSI and S-RSI in sequence. The original parameters of the S-RSI are 14 days for both *n* and *t* with an overbought level of 70 and oversold level of 30.

$$S\_RSI = \frac{RSI_{current} - min(RSI_t)}{max(RSI_t) - min(RSI_t)} \tag{4}$$

$$RSI_{current} = 100 - \frac{100}{1 + \frac{avg\_gain(n)}{avg\_loss(n)}} \tag{5}$$

B. Objective functions

For the optimization process at hand, three conflicting objectives are selected.

- The percentage Return on Investments (RoI): it evaluates the profitability of the investment strategy by comparing the investment net gain to the total investment costs as shown in Eq. (6). This is a maximization objective as the investors always aim to maximize their profits.

$$RoI = \frac{Net\_gains}{Investment\_costs} * 100 \tag{6}$$

- Sortino Ratio (SR): is a measure of the risk of investment. It is calculated as in Eq. (7), where σ_{down} is the Standard Deviation (SD) of the returns that are below the average (the downward SD). The target here is to maximize the SR in order to minimize the risk.

$$SR = \frac{aggregated\_return}{\sigma_d} \tag{7}$$

- The number of trades: The target here is to get the minimum number of trades.

III. The Proposed Algorithm

As mentioned before, the idea behind the decomposition-based algorithms is to simplify the MO problem and convert it into a group of single objective SPs. This simplification is accomplished using the SF or aggregation function. Different scalarization approaches were found in the literature. Among the most recommended SFs due to its simplicity and adaptability to different types of problems is the Chebyshev approach [33]. As a result, a number of variants were proposed in the literature to enhance its performances.

The Augmented Chebyshev (ACh) is one of the variants with an additional augmentation term that was proposed in order to improve the quality of the PF solutions and to discard the weak optimal solutions [34]. This is the one which is adopted in this research.

The approximated PF of the MO problem represented in Eq. (1) can be obtained by simplifying it into simpler scalar SP. The objective function of the j^{th} SP is given as in Eq. (8).

minimize

$$g^{ACh}(x \mid w^j, z^*) = max_{1 \le i \le N}\{ w_i^j \left| \frac{f_i(x) - z_i^{nad}}{z^* - z_i^{nad}} \right| \} + \rho \sum_{i=1}^{N} |f_i(x) - z_i^*| \tag{8}$$

Such that: $z^* = (z_1^*, z_2^*, \dots \dots z_m^*)^T$ is the reference point, i.e., $z_i^* = \max\{f_i(x) | x \in \Omega\}$ for $i = 1 \to m$ objectives and z_i^{nad} is the nadir point, where $z_i^{nad} = \min\{f_i(x) | x \in \Omega\}$.

The second part of the previous equation is the augmentation term, where ρ is the augmentation parameter which is a very small value such that $\rho \in [0.001, 0.1]$ [35].

Each SP is assigned a separate weight vector, such that each vector has m elements one for each objective $w = \{w_1, w_2, \dots . w_m\}$, where each element value ≥ 0 and the summation of all elements equals 1.

A neighborhood technique is utilized to optimize each SP based on its neighboring SPs. The j^{th} SP neighborhood comprises the set of SPs that are with T distance from j, such that T represents the neighborhood size. The steps of the original MOEA/D can be found in [18].

The proposed algorithm (algorithm 1) aims to optimize the parameters of three algorithmic trading approaches over three challenging objective functions. In this research, a new MOPSO/D algorithm is implemented, where the algorithm starts by randomly generating the initial particles' positions, velocities, such that, $P^j(t)$ and $v^j(t)$ are the current position and velocity of particle j at time t in sequence. In our application, the particles' positions represent the indicators' parameters.

The PSO considers the interactions and movements of particles as a swarm to locate the optimal points in the search space. Over different iterations, the particles cluster into an ideal position in the search space by employing both exploration and exploitation. The particles attempt to improve their position in the known beneficial regions by exploitation, while also exploring undiscovered regions of the feasible space.

Both exploitation and exploration processes can be managed during the velocity update step. Such that the accelerators c_1 and c_2 are used for the exploitation purposes, whereas. The inertia component ω is needed for exploration.

Algorithm 1: The proposed MOPSO/D algorithm for BTC trading

Inputs:

'C historical prices.

: The number of SPs.

The number of objectives.

: An initial set of weight vectors.

The size of the neighborhood.

npty External Archive (EA).

Steps:

1. Initialization:

- Generate a swarm of particles $x^1 \rightarrow x^N$ at random.
- For each particle j, initiate the current position P^j. The personal and global best positions, i.e., P^j_{Pbest}, and P^j_{Gbest} are initially set as P^j.
- For each particle j, initialize the velocity v^j=0.
- Initialize the reference point z^* and nadir point z^{nad}
- Calculate the Euclidean distances for each couple of weight vectors.
- Define the neighborhood of each particle j as, $B(j) = \{j_1, \cdots, j_T\}$, where $w^{j_1}, \cdots, w^{j_T}$ are the T closest weight vector to w^j.

2. Update

For (each iteration $i =$ 1→ $iter$), do

For (each particle j = 1→ N), do

Calculate: g^{ACh}.

According to the g^{ACh} *value,* update P^i_{Pbest} and P^i_{Gbest}.

Update the reference and nadir points.

Update the velocity as:

$$v^j(t+1) = \omega\, v^j(t) + c_1 r_1 \left(P^j_{Pbest} - P^j(t)\right) + c_2 r_2 \left(P^j_{Gbest} - P^j(t)\right)$$

where, ω is the inertia component, c_1 and c_2 are two predefined constant accelerators, r_1, and r_2 are two random variables ∈ [0,1].

Update the current position as:

$$P^j(t+1) = P^j(t) + v^j(t+1)$$

Apply mutation operator.

Update the EA.

End

Update weight vectors W (algorithm 2).

Update neighborhood $B(j)$.

End

3. Return EA.

For further exploration of the search space, a uniform mutation is added to the algorithm to enhance the diversity of the current algorithm; however the resultant particles after mutation are subjected to a repair process that ensures that the new particles positions are never worse than the previous ones. Finally, the nondominant solutions are returned in a final External Archive (EA).

A. *Weight Assignment*

As mentioned before, the weight assignment method plays a crucial role in influencing the search process of MOEA/D algorithms. The diversity or the distribution of the weight vectors affects the quality of the final solutions to a high degree. Similar or identical vectors generate poor solutions that could not efficiently cover the whole PF.

The methods for generating weight vectors can be categorized as either uniform or random weight creation. Uniform weight creation involves constructing weights in repetitive patterns to guarantee the equitable distribution of vectors over the PF.

In [17], Zhang introduced a structured or uniform weight distribution architecture, such that each weight vector ∈ $\left\{0, \frac{1}{H}, \frac{2}{H}, \ldots, \frac{H}{H}\right\}$, with H being a positive integer parameter used for regulation. The number of SPs or weight vectors $N = C^{m-1}_{H+m-1}$ (C refers to the mathematical combinations).

The uniform distribution is effective in problems with continuous PFs. Nevertheless, it is not suitable for problems with complicated or scattered PFs [36]. A further problem with uniform distribution is that it occasionally generates vectors that are similar or extremely near to each other, resulting in duplicate solutions.

On the contrary, the random distribution of weights allows for a more comprehensive investigation of the search space, as it produces vectors that are not necessarily evenly distributed throughout the PF [18]. This, in turn, yields a range of distinctive and varied solutions since it generates vectors that are dissimilar from each other. The drawback with randomly distributed weights is that there is no assurance that the resulting vectors can accurately represent the whole PF [18].

Since the shape of the (PF) for real world problems is scattered and cannot be easily covered, finding the Pareto optimal points cannot be obtained by simple search strategies. To overcome this problem, an alternative scenario suggesting a Dual Weight assignment mechanism has been implemented denoted as (MOPSO/D-DW).

As seen, both uniform and random assignments have their benefits and drawbacks. In this study, a new weight assignment method is developed to combine the benefits of both strategies into a single algorithm (Algorithm 2).

The proposed strategy is a simple yet efficient. The main idea is to switch between the uniform and random generations over the different iterations.

This process ensures keeping the whole PF covered through the uniform distribution iterations while hitting other random areas during the random iterations. In this case, the particles neighbors are recalculated during each iteration which improves the particles experience through the search process improving both the diversity and convergence.

Algorithm 2: Dual weight assignment algorithm

Inputs:

- *H:* A regulating integer parameter greater than zero.
- *m:* The number of objectives.
- *iter:* The number of iterations.

Steps:

1. Calculate the number of weight vectors $= C_{H+m-1}^{m-1}$.
2. Let U be a set of values in the range [0,1] with an increment of $1/\mathrm{H}$. U = {$0, 1/\mathrm{H}, 2/\mathrm{H}, \ldots \ldots 1$}.
3. Evaluate

For (*j*=1→*N*), do

  For (*i*=1→*m*), do

  - Generate a new non repeated weight vector w_i^j, such that $w_i \in U$ and $\sum_{i=1}^{m} w_i = 1$.
  - Append w_i^j to $\mathrm{W_U}$, such that $\mathrm{W_U}$ is the set of all uniformly generated vectors.

  End

End

For (*i*=1→ *iter*), do

If (*i* % 2== 0), then

For (*j*=1→*N*), do

  - Generate a weight vector w_i^j, with w_i randomly selected from [0,1] and $\sum_{i=1}^{\mathcal{N}} w_i = 1$.
  - Append w_i^j to W.

  End

Else W← W_U

End

End

4. Return W

To summarize the optimization process in a more understandable way, the algorithmic trader first generates a set of random indicators' parameters (particles positions) along with a collection of a uniformly generated weight vectors one for each particle. The particles positions are then assessed by applying the indicators parameters to the training data set and calculating the resultant values for each of the objectives.

Moreover, a fitness aggregation technique in Eq. (8) is then employed to aggregate the three objectives. Based on the aggregated fitness value, the personal and global best positions are updated which in turn are used for updating both the velocities and positions of the particles. The algorithm continues till the maximum iterations and the final results (the best parameters) are obtained from the EA. These parameters are then tested on the testing data set.

IV. Experimental Results

The proposed algorithm is used to find the optimal trading signals for (BTC versus USD) trading during the interval from 3/1/2017 till 3/1/2019 which is considered as the training interval. The optimal set of parameters found during the search are tested upon the closing prices during the testing interval 3/1/2019 till 3/1/2021.

A. Evaluation Metrics

An evaluation metric serves as an indicator or assessment of the quality of the solutions that have been developed. Multiple measures exist for evaluating MO algorithms, each designed to assess distinct characteristics [37]. The convergence, diversity, and statistical measures are crucial assessment factors. Various metrics have been discovered that assess either a single criterion or multiple criteria concurrently. The study included three distinct indicators: the Generational-Distance (GD), the Hypervolume (HV), and the Average Fitness (AF).

- The GD serves as a metric to quantify the distance between the produced non-dominated solutions (the estimated PF) and the actual or true PF [38]. In real-world problems, it is possible to utilize a reference set derived from the collection of estimated PFs produced from all the search techniques under consideration. The lowest GD value indicates the proximity of the derived solution set to the true PF or reference set, and conversely, the higher the GD value, the more away the solution set is from the true Pareto Front [38].
- The HV indicator quantifies both the diversity and the convergence. It measures the *m*-dimensional volume of the objective space region, which is defined by the estimated PF and a reference point that is dominated by all solutions in the front [37].
- The AF is the average of the fitness values of the PF solutions found along a set of independent runs.

B. Parameter Settings

The parameter settings are as follows: the swarm size N = 351, the uniform weight regulating parameter H = 25. The total number of iterations is 150. The augmentation variable ρ = 0.05. The mutation rate=0.15. A neighborhood size (T) = 20. For PSO parameters, the inertia component ω =0.8, both constant accelerators c_1 and c_2 are set as 0.5. For the original MOEA/D, the crossover rate =0.8 and the other parameters are kept as before.

The indicators parameters are as follows: the time spans for all the indicators ∈ [3, 120] days. The ES-RoC over bought and sold levels range within ±10% from the center line. The S-RSI overbought level ∈ [60, 90], while the oversold level ∈ [10, 40].

C. Results

To evaluate convergence and diversity of the solutions generated by the proposed algorithm (MOPSO/D-DW) against the original MOEA/D, each algorithm is evaluated over 10 independent runs. The reported results in the following tables are the best and average values for each algorithm over these runs.

Table I shows a comparison between the values of the best and average GDs obtained by both algorithms. The best results among the two alternatives are displayed in bold. Again, the best GDs are the lowest values.

Table II shows a comparison between the best and average HVs obtained by the algorithms, where the best values (the higher values) are also shown in bold as before.

As seen from the tables, the proposed MOPSO/D-DW algorithm showed the best values in terms of both metrics, i.e., GD and HV. The proposed strategy always showed lower GDs and higher HVs which ensures closer solutions to the true PF with the best distribution of solutions.

To evaluate the efficiency of trading signals generated by the proposed algorithm, the AF of the obtained solutions over 10 independent runs are compared against both the MOEA/D and the TIs using their original parameters.

Due to the conflicts among the different objectives, the evaluation of the counterpart strategies is performed through a ranking methodology, such that each objective is ranked as compared to the same objective over the three counterpart strategies. The best rank is given a value of one, while the worst is given a rank of three. Finally, the total ranks obtained by each strategy are summed in order to evaluate its overall performance.

Table III shows a comparison of the three trading strategies, i.e., the TIs using their original parameters, the MOEA/D and the proposed MOPSO/D-DW during training.

TABLE I. A COMPARISON OF THE BEST AND AVERAGE GD OBTAINED BY EACH ALGORITHM

		MOEA/D	MOPSO/D-DW
DWMA	Best	1.40E-03	**0.00E+00**
	Average	2.53E-02	**0.00E+00**
ES-RoC	Best	5.91E-02	**3.20E-03**
	Average	7.83E-02	**4.34E-03**
S-RSI	Best	4.32E-01	**1.73E-01**
	Average	8.74E-01	**2.95E-01**

TABLE II. A COMPARISON OF THE BEST AND AVERAGE HV OBTAINED BY EACH ALGORITHM

		MOEA/D	MOPSO/D-DW
DWMA	Best	9.09E-02	**9.72E-02**
	Average	5.98E-02	**9.72E-02**
ES-RoC	Best	2.00E-03	**2.19E-02**
	Average	1.54E-03	**1.14E-02**
S-RSI	Best	5.06E-02	**2.91E-01**
	Average	2.97E-02	**2.64E-01**

TABLE III. THE AF OBTAINED BY THE THREE TRADING STRATRGIES FOR EACH INDICATOR OVER 10 INDEPENDENT RUNS DURING TRAINING

		Original parameters			MOEA/D			MOPSO/D-DW		
		RoI%	SR	Trades	RoI%	SR	Trades	RoI%	SR	Trades
DWMA	AF	518.49	0.87	10	559.91	0.88	5.92	704.18	1.08	7.41
	Rank	3	3	3	2	2	1	1	1	2
ES-RoC	AF	-29.85	-0.07	7	102.19	5.03	1.46	264.27	7.3	2.44
	Rank	3	3	3	2	2	1	1	1	2
S-RSI	AF	-75.6	-0.34	59	216.3	0.37	17.97	437.97	0.57	16.98
	Rank	3	3	3	2	2	2	1	1	1
Total ranking			27			16			11	

TABLE IV. THE AF OBTAINED BY THE THREE TRADING STRATRGIES FOR EACH INDICATOR OVER 10 INDEPENDENT RUNS DURING TESTING

		Original parameters			MOEA/D			MOPSO/D-DW		
		RoI%	SR	Trades	RoI%	SR	Trades	RoI%	SR	Trades
DWMA	AF	**983.77**	**1.03**	7	627.34	1.1	**6.26**	804.8	1.01	7.94
	Rank	1	1	2	3	3	1	2	2	3
ES-RoC	AF	-4.29	0.02	50	72.76	0.16	**20.06**	**76.63**	**0.28**	22.65
	Rank	3	3	3	2	2	1	1	1	2
S-RSI	AF	25.17	0.42	50	87.58	1.68	0.63	**203.11**	**2.05**	**1.01**
	Rank	3	3	3	2	2	2	1	1	1
Total ranking			22			18			14	

As seen from the table, the proposed MOPSO/D-DW algorithm could provide the best ranking followed by the MOEA/D with the TI's original parameters are given the worst ranking.

The same process is repeated in order to evaluate the benchmark methodologies during testing. As seen from Table IV, the proposed algorithm again provided the best ranking followed by MOEA/D.

Fig. 2 shows the summation of the ranks obtained by the last tables during both training and testing for each of the algorithms. Again, the algorithm assigned the lowest sum of ranks is the best. As can be seen, the proposed algorithm could achieve the best ranking during both the periods under study.

For further analysis, the best and median values obtained by the proposed algorithm are examined during both training and testing periods, such that the comparison is performed based on the average performance of each objective independently. In this case, the best obtainable RoI values for each indicator (i.e., DWMA, S-RoC, S-RSI) are averaged during both training and testing, and so on for the rest of the objectives. This process is repeated for the median value for each objective during both training and testing. The best and median values are compared to the original TIs in order to check the overall performance of the generated trading strategies in different ways, especially during the testing period, which is the main challenge.

Fig. 3 shows the best and median RoI values obtained by the proposed algorithm MOPSO/D-DW over 10 runs during both training and testing compared to the original TI's. As seen, during training, the best and median RoI values extremely exceed the indicators' original parameters RoI. During testing, it can be noted that both the median and the original indicators' parameters provide returns (RoI) that are close to each other, however, the median RoI of the proposed algorithm is still higher than the original parameters.

Fig. 4 shows the best and median SR obtained by the proposed algorithm versus the original indicators during both training and testing. Again, SR is a measure of the risk with higher values indicate lower risk. It can be seen that, the proposed algorithm provides better SR in all cases.

To figure out the effect of the selected parameters on the final trading signals in more detail, an example showing the difference between the effect of trading using the DWMA original parameters, i.e., 20-50 days and one of the generated solutions by the proposed algorithm, i.e., 18-38 days during the testing period for BTC, is clarified through Fig. 6 and Fig. 7 consequently.

Fig. 5 shows the best and median number of trades obtained during the two periods. In this case each buy-sell pair is considered as a single trade. As previously mentioned, lower values are always required as this in turn reduces the trading commissions. The proposed algorithm could also maintain the best number of trades in all the cases.

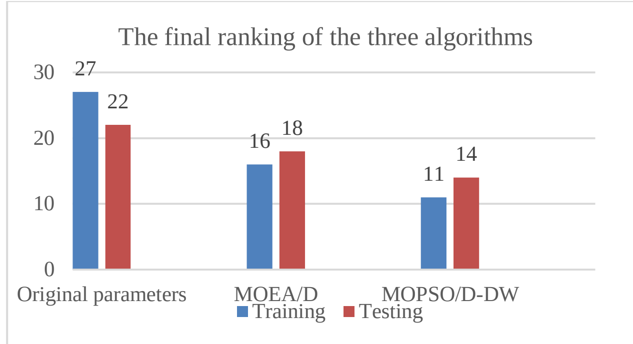

Fig. 2. The final summation of the ranks generated by the three trading strategies.

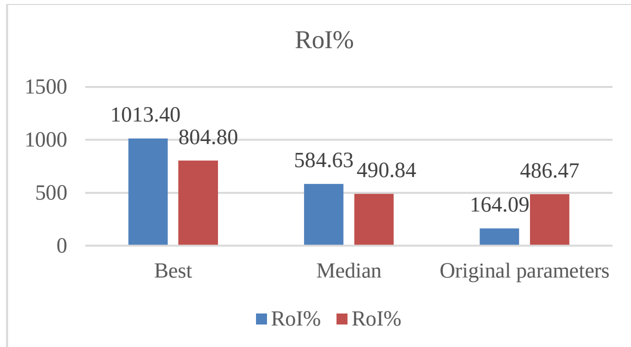

Fig. 3. The best and median RoI values obtained by the MOPSO/D-DW compared against the original TIs during both training and testing.

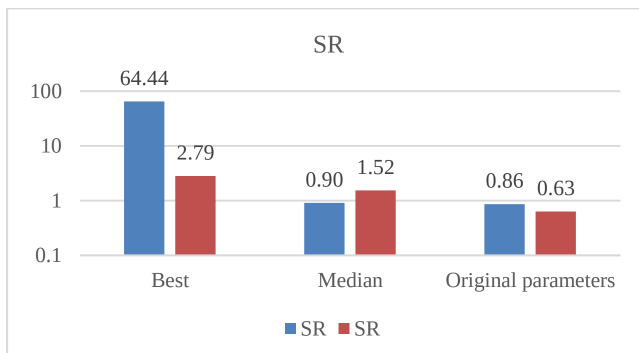

Fig. 4. The best and median SR values obtained by the MOPSO/D-DW compared against the original TIs during both training and testing.

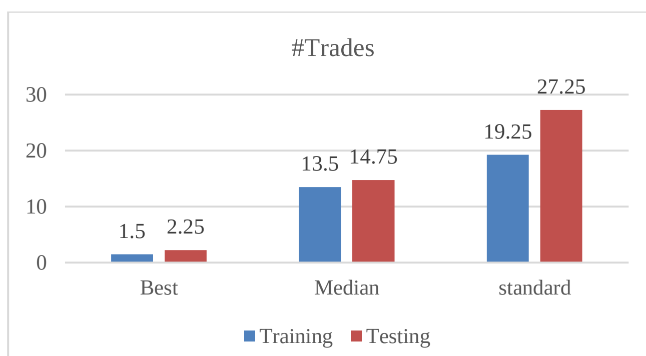

Fig. 5. The best and median number of trades obtained by the MOPSO/D-DW compared against the original TIs during both training and testing.

Fig. 6. The Crossovers between an 20-50-days DWMA indicator (the red line for the short WMA, whereas the blue line is for the long WMA) for BTC / USD trading (The buying and selling prices are highlighted in blue in each case).

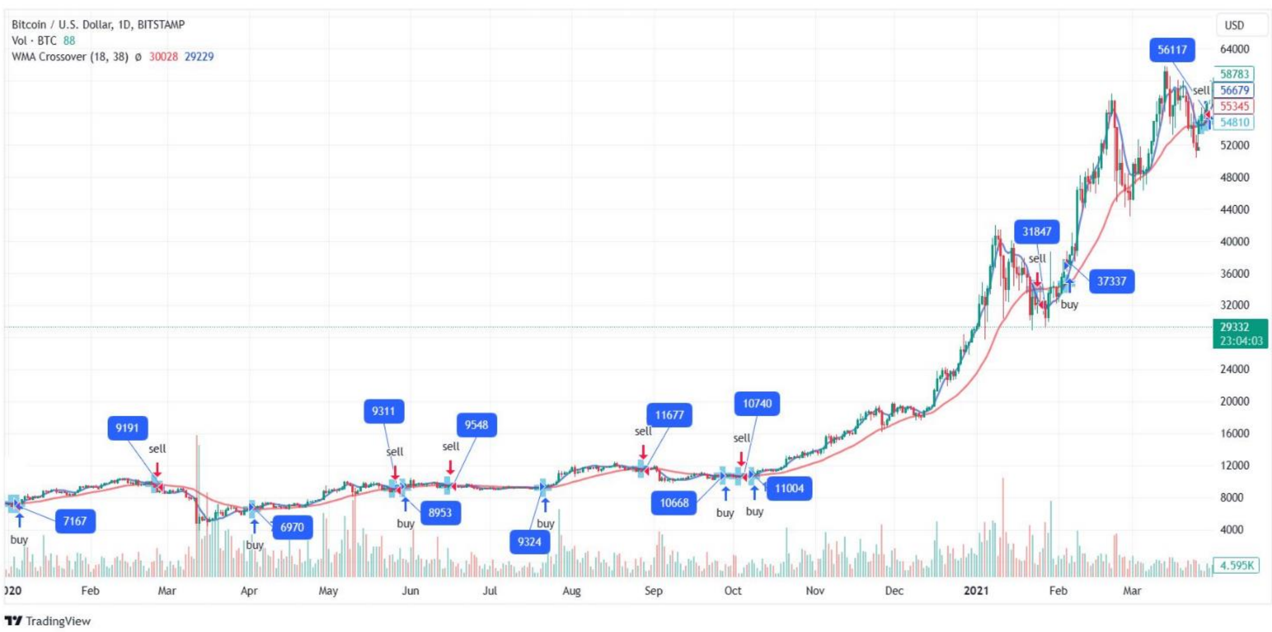

Fig. 7. The Crossovers between an 18-38-days DWMA indicator (the red line for the short WMA, whereas the blue line is for the long WMA) for BTC / USD trading (The buying and selling prices are highlighted in blue in each case).

The figures show the set of generated signals, i.e., buy and sell, by each parameter combination, showing the buying and selling price in each case. As seen, both of the parameter sets provide a set of profitable trades during the period under study; however, the generated parameter combination in this case is more sensitive to market changes, providing more profitable trades.

V. Conclusion

Cryptocurrency, specifically Bitcoin (BTC), is a decentralized digital payment system that takes its name from the encryption mechanism used for verifying their digital transactions. As there is no need for traditional banking strategies, it attracted millions of traders all over the world. Crypto markets always suffer from instability or volatility which either yields to high returns or extremely harmful losses.

So, the study and analysis of the market is a major demand for investors to get the best returns with the least possible risks.

In this research, a new algorithm is proposed to optimize the trading signals' allocation strategies found in the literature named TIs. The algorithm proposes a new MOPSO/D that is based on a new dual weight assignment methodology MOPSO/D-DW. The algorithm is used to optimize three of the most famous and widely used TIs named DWMA, ES-RoC, and S-RSI.

The algorithm is compared to the MOEA/D and the TIs with their original parameters based on three objectives. The RoI is used for the evaluation of the investment returns, the SR is used for risk evaluation and the third objective is a calculation of the total number of trades. Results showed that the proposed algorithm showed promising results in terms of all the evaluation metrics during the training and testing intervals.

The problem with this optimization process is that the parameters that provide the best returns during the training period are not always the best during testing, but they still maintain good performance with high returns or at least no general losses. This is due to the extreme volatility found during both training and testing, such that the market shows a sideway performance during the testing period, with an extreme jump at the end of the period. For future research, a more complex normalization process could be tested instead of the linear normalization used in this research. The optimization process can be extended to more indicators and a large number of cryptocurrencies. The proposed dual weight-generation strategy could also be tested under different conditions.

REFERENCES

[1] M. V. Alstyne, "Why Bitcoin Has Value," *Communications of the ACM,* vol. 57, no. 5, pp. 30-32, 2014.

[2] N. Daskalakis and P. Georgitseas, An Introduction to Cryptocurrencies: The Crypto Market Ecosystem, London, 2020.

[3] J. Chen, "Analysis of Bitcoin Price Prediction Using Machine Learning," *J. Risk Financial Manag.,* vol. 16, no. 51, 2023.

[4] M. J. Hamayel and A. Y. Owda, "A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms," *AI,* vol. 2, no. 4, pp. 477-496, 2021.

[5] P. Jaquart, D. Dann and C. Weinhardt, "Short-term bitcoin market prediction via machine learning," *The Journal of Finance and Data Science,* vol. 7, pp. 45-66, 2021.

[6] G. Borrageiro, N. Firoozye and P. Barucca, "The Recurrent Reinforcement Learning Crypto Agent," *IEEE Access,* vol. 10, pp. 38590-38599, 2022.

[7] L. Lorenzo and J. Arroyo , "Online risk-based portfolio allocation on subsets of crypto assets applying a prototype-based clustering algorithm," *Financial Innovation,* vol. 9, no. 25, 2023.

[8] B. Joemon, M. A. Ghazanfar, M. A. Azam, N. Z. Jhanjhi and A. A. Khan, "Novel heuristics for Stock portfolio optimization using machine learning and Modern Portfolio Theory," in *International Conference on Business Analytics for Technology and Security (ICBATS)*, Dubai, 2023.

[9] P. Hrytsiuk, T. Babych and L. Bachyshyna, "Cryptocurrency Portfolio Optimization Using Value-At-Risk Measure," in *6th International Conference on Strategies, Models and Technologies of Economic Systems Management (SMTESM 2019)*, 2019.

[10] Y. He and . C. Aranha, "Solving Portfolio Optimization Problems Using MOEA/D and L´evy Flight," *ArXiv,* vol. abs/2003.06737, 2020.

[11] Q. Zhang, H. Li, D. Maringer and a. E. Tsang, "MOEA/D with NBI-style Tchebycheff approach for Portfolio Management," in *IEEE Congress on Evolutionary Computation,* Barcelona, Spain , 2010.

[12] D. Philippas, N. Philippas, P. Tziogkidis and H. Rjiba, "Signal-herding in cryptocurrencies," *Journal of International Financial Markets, Institutions and Money,* vol. 65, 2020.

[13] M.-F. Leung, L. Chan, W.-C. Hung, S.-F. Tsoi, C.-H. Lam and Y.-H. Cheng, "An Intelligent System for Trading Signal of Cryptocurrency Based on Market Tweets Sentiments," *FinTech.,* vol. 2, no. 1, pp. 153-169, 2023.

[14] D. Garcia and F. Schweitzer, "Social signals and algorithmic trading of Bitcoin," *R. Soc. open sci.,* vol. 2, no. 9, 2015.

[15] T. E. Koker and D. Koutmos, "Cryptocurrency Trading Using Machine Learning," *Risk and Financial Management,* vol. 13, p. 178, 2020.

[16] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms: An Introduction", In: Wang, L., Ng, A., Deb, K. (eds) Multi-objective Evolutionary Optimisation for Product Design and Manufacturing. Springer, 2011.

[17] Q. Zhang and a. H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Transactions on Evolutionary Computation,* vol. 11, no. 6, pp. 712 - 731, 2007.

[18] S. M. Omran, W. H. El-Behaidy and A. A. A. Youssif, "Decomposition Based Multi-objectives Evolutionary Algorithms Challenges and Circumvention," *Intelligent Computing,* vol. 1229, pp. 82-93, Advances in Intelligent Systems and Computing. Springer SAI 2020.

[19] S. Jiang, S. Yang, Y. Wang and a. X. Liu, "Scalarizing Functions in Decomposition-Based Multiobjective Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation,* vol. 22, no. 2, pp. 296 - 313, 2018.

[20] M. Grabisch, J.-L. Marichal, R. Mesiar and E. Pap, "Aggregation Functions. Part I: Means," *Information Sciences, Elsevier,* vol. 181, no. 1, pp. 1-22, 2011.

[21] M. Pescador-Rojas and a. C. A. C. Coello, "Collaborative and Adaptive Strategies of Different Scalarizing Functions in MOEA/D," in *IEEE Congress on Evolutionary Computation (CEC)*, 2018.

[22] X. Ma, Q. Zhang, G. Tian, J. Yang and a. Z. Zhu, "On Tchebycheff Decomposition Approaches for Multi-objective Evolutionary Optimization," *IEEE Transactions on Evolutionary Computation,* vol. 22, no. 2, pp. 226 - 244, 2018.

[23] L. R. C. d. Farias, P. H. M. Braga, H. F. Bassani and a. A. F. R. Araújo, "MOEA/D with Uniformly Randomly Adaptive Weights," in *GECCO '18*, Kyoto, Japan, 2018.

[24] X. Guo, X. Wang and a. Z. Wei, "MOEA/D with Adaptive Weight Vector Design," in *11th International Conference on Computational Intelligence and Security*, Shenzhen, China, 2015.

[25] M. Wu, S. Kwong, Y. Jia, K. Li and a. Q. Zhang, "Adaptive weights generation for decomposition-based multi-objective optimization using Gaussian process regression," in *GECCO '17 Proceedings of the Genetic and Evolutionary Computation Conference*, Berlin, Germany, 2017.

[26] K. Erwin and A. Engelbrecht, "Meta-heuristics for portfolio optimization," *Soft Computing,* 2023.

[27] S. Sarkar, S. Das and a. S. S. Chaudhuri, "Multi-Level Thresholding with a Decomposition-based Multi-Objective Evolutionary Algorithm for Segmenting Natural and Medical Images," *Applied Soft Computing,* vol. 50, pp. 142-157, 2017.

[28] H. Xing, Z. Wang, T. Li and a. H. Li, "An Improved MOEA/D Algorithm for Multi-objective Multicast Routing with Network Coding," *Applied soft computing* , vol. 59, pp. 88-103, 2017.

[29] K. Metaxiotis and K. Liagkouras, "Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review," *Expert Systems with Applications,* vol. 39, no. 14, pp. 11685-11698, 2012.

[30] K. Frajtova-Michalikova, E. Spuchľakova and M. Misankova, "Portfolio Optimization," in *4th World Conference on Business, Economics and Management, WCBEM*, 2015.

[31] M. A. Lim, The Handbook of Technical Analysis, Singapore: John Wiley & Sons , 2016.

[32] Y. Su, C. Cui and H. Qu, "Self-Attentive Moving Average for Time Series Prediction," *MDPI in Applied sciences,* vol. 12, 2022.

[33] X. Ma, Q. Zhang, G. Tian, J. Yang and a. Z. Zhu, "On Tchebycheff Decomposition Approaches for Multi-objective Evolutionary Optimization," *IEEE Transactions on Evolutionary Computation,* vol. 22, no. 2, pp. 226 - 244, 2018.

[34] T. Chugh, "Scalarizing Functions in Bayesian Multiobjective Optimization," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, Glasgow, UK, 2020.

[35] K. Dachert, J. Gorski and K. Klamroth, "An augmented weighted Tchebycheff method with adaptively chosen parameters for discrete bicriteria optimization problems," *Computers & Operations Research,* vol. 39, pp. 2929-2943, 2012.

[36] X. Chen, J. Yin, D. Yu and X. Fan , "A decomposition-based many-objective evolutionary algorithm with adaptive weight vector strategy," *Applied Soft Computing,* vol. 128, 2022.

[37] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel and L. Salomon, "Performance indicators in multiobjective optimization," *European Journal of Operational Research,* vol. 292, no. 2, pp. 397-422, 2020.

[38] Miqing Li and T. Chen, "How to Evaluate Solutions in Pareto-based Search-Based Software Engineering? A Critical Review and Methodological Guidance," *IEEE Transactions on Software Engineering,* vol. 48, no. 5, pp. 1771-1799, 2022.