Particle Swarm Optimization of Custom Bitcoin Trading Algorithm

Mateusz Zdzisław Jaworski
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
m.z.jaworski@student.utwente.nl

ABSTRACT

Relative Strength Index (RSI) is a well-known technical indicator, which outputs are frequently used by traders and algorithms alike as a part of their decision process. Unfortunately, it is also suspect to generating numerous false signals, which reduces its performance. In this study we attempt to mitigate this issue by assembling a custom RSI based trading algorithm, referred to as 3-IRSI. Consequently, Particle Swarm Optimization (PSO) is applied in order to find optimal Bitcoin trading setups. Its performance is then compared against PSO optimized RSI and one other commonly used algorithm, as well as buy-and-hold strategy. Experiment results are presented in the form of descriptive statistics and demonstrate that the resulting algorithm is capable of outperforming its peers.

Keywords

Numerical Optimization, Particle Swarm Optimization, Oscillators, Technical Indicators, RSI, MACD, 3-IRSI

1. INTRODUCTION

Bitcoin emerged in 2009 [3] as a peer-to-peer electronic cash system [18] and since then it has received a lot of attention due to its decentralized nature. Its unique characteristics have prompted research into understanding how it can be classified [14] and properly regulated [16, 20]. As of today bitcoin is experiencing a widespread adoption. Unfortunately, its price can experience a lot of volatility [2]. Which makes it hard for investors to achieve profit and as a result they seek algorithms that would improve their portfolio's performance [6].

Such algorithms can be grouped into many categories, with technical being one of the oldest and most widely applied [5, 15]. Aforementioned algorithms can be further classified into multiple subcategories depending on the range of utilized techniques. One of the most widely used are trend-following indicators such as Moving Average Convergence Divergence (MACD) [1] and momentum oscillators such as Relative Strength Index (RSI) [22]. Unfortunately, during prolonged bull markets and bear markets oscillators such as RSI are known for their ability to remain in one of the two value extremes. This result in their ability to misidentify market bottoms and ups and as a consequence in suboptimal trading decisions [23].

In this paper we propose a new RSI based algorithm, denoted as 3-Period Integrated Relative Strength Index (3-IRSI), which hopes to alleviate some of the RSI's shortcomings. The need to evaluate its performance resulted in the formulation of the following research questions:

- **RQI**: How does 3-IRSI fare against buy-and-hold strategy?
- **RQII**: What is the performance of the 3-IRSI in comparison to RSI and MACD?

In order to answer those research questions, we have used one hour resolution bitcoin trading data ranging from April 2019 up to and including November 2021. Subsequently, Particle Swarm Optimization (PSO) [12] was applied in order to find optimal 3-IRSI, RSI and MACD setups. The results are presented in the form of descriptive statistics and demonstrate that, under chosen methodology, 3-IRSI is capable of substantially outperforming its peers.

2. RELATED WORK

Even though technical analysis was originally developed for a different type of financial assets, Corbet et al. demonstrated that it is capable of generating useful Bitcoin trading signals [7]. Similarly, Detzel et al. demonstrated that moving averages can predict Bitcoin returns [8]. Furthermore, Huang et al. evince that their model utilizing 124 different technical analysis algorithms possess a strong predictive power and is capable of outperforming buy-and-hold strategy [11].

Briza and Naval Jr. demonstrated that their setup consisting of numerous technical indicators paired with Particle Swarm Optimizer and a custom multi-objective optimization function radically surpassed performance of various considered alternatives [4]. Likewise, Cohen demonstrated that PSO equipped with a multi-objective optimization function consisting of: percentage of dollar value invested (OMDD), net profitability factor (NPF) and profitability percentage (PP) is capable of yielding profitable MACD and RSI setups [6]. Whereas, Nakano et al. [19] used a neural network paired with a mixture of different technical indicators, attaining higher performance than the distinct rules themselves.

As previously demonstrated, current research can be divided into two main categories. Firstly, a lot of emphasis is put on evaluating performance of pre-existing technical indicators. Secondly, such indicators are often used as a part of some greater system. In this context our main contribution is investigation into whenever it may be worthwhile to build upon them.

36th *Twente Student Conference on IT* February. 4th, 2022, Enschede, The Netherlands.

3. EXPERIMENT DESIGN

For the purpose of this experiment, we will assume a frictionless market, 1,000 euros of starting capital and use OHLCV Bitcoin trading data of one hour resolution. The data itself range from April 2019 up to and including November 2021. The choice of the OHLCV format has been made due to its perceived popularity and ability to capture the most important aspects of the historical price activity. In essence, OHLCV is a five-tuple containing selected information about trading activity within some predetermined time frame such as one hour or a day. Its exact structure can be judged from (1).

$$OHLCV = (OPEN, HIGH, LOW, CLOSE, VOLUME) \tag{1}$$

OPEN and CLOSE refer to the price of the underlying asset at the beginning and at the end of the specified time frame. HIGH and LOW to the highest and lowest value achieved within that period, whereas VOLUME refers to the amount of the asset traded. The example of so-called candlestick pattern resulting from plotting OHLCV data can be seen in Figure 1. A red candlestick implies its CLOSE value is lower than its OPEN value, conversely for green.

Figure 1. Candlestick pattern resulting from plotting Bitcoin's one hour OHLCV data

In order to simulate a real-world scenario, our experimental setup contains an internal clock denoted as t_n. This clock is then incremented using "ticks" of one hour. Next, clock's time is used to retrieve a vector of preceding 100 hours of the Bitcoin trading activity. Where the number of data entries per tick has been chosen arbitrarily as a balancing factor between signal accuracy and the amount of data in needed of processing. As a next step this OHLCV vector is processed using (2). This operation yields a vector of the last 100 hourly bitcoin trends.

$$GAIN\_LOSS(OHLCV) = \frac{CLOSE - OPEN}{OPEN} * 100 \tag{2}$$

Subsequently, such input vectors are processed by a decision module consisting of either RSI, 3-IRSI or MACD and some internal logic ensuring that assets cannot be sold or bought twice in a row. Finally, portfolio's value at the time of the last SELL action is returned. Resulting chain of actions is subsequently denoted as M_GAIN in Figure 4.

We will now briefly give more background to the RSI and MACD, which will be followed by the exact algorithm for 3-IRSI and explanation how each trading strategy was incorporated into Particle Swarm Optimizer.

3.1 Relative Strength Index

Relative Strength Index is a bounded momentum indicator [22] whose output ranges from zero up to 100. Its exact formula can be seen bellow as (3) and (4). Where $AVG\_GAIN(Input)_\alpha$ and $AVG\_LOSS(Input)_\alpha$ refers to the average gain and loss incurred during last α periods. Example RSI output can be seen in Figure 2.

$$RSI(Input)_\alpha = 100 - \frac{100}{1 + RS(Input)_\alpha} \tag{3}$$

$$RS(Input)_\alpha = \frac{AVG\_GAIN(Input)_\alpha}{AVG\_LOSS(Input)_\alpha} \tag{4}$$

It's important to note that many trading strategies utilizing this indicator exist. Alas due to the time constraints, we will consider only 'touch' trading strategy [23]. This strategy generates BUY and SELL signals whenever RSI output 'touches' values of 30 and 70.

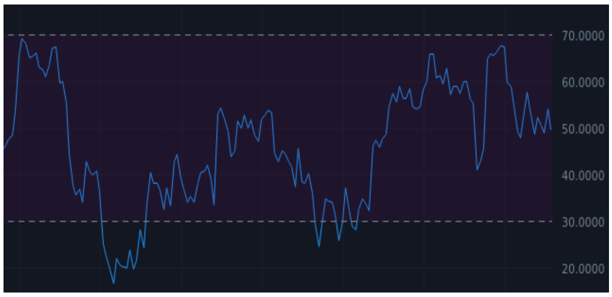

Figure 2. RSI signal resulting from Figure's 1 data

3.2 Moving Average Convergence Divergence

Moving Average Convergence Divergence is an unbounded trend indicator [1], which is constructed from a difference between two Exponential Moving Averages (EMAs) [17] as can be seen in (6). This, in turn, is processed by another EMA generating so-called signal line (5). Resulting signals are then overlaid and their crossovers are used to generate BUY and SELL signals.

$$SIGNAL(Input)_{\alpha,\beta,\gamma} = (EMA_\gamma \circ MACD_{\alpha,\beta})(Input) \tag{5}$$

$$MACD(Input)_{\alpha,\beta} = EMA(Input)_\alpha - EMA(Input)_\beta \tag{6}$$

Figure 3 contains a sample output of previously mentioned operations, where MACD and signal line are yellow and red respectively.

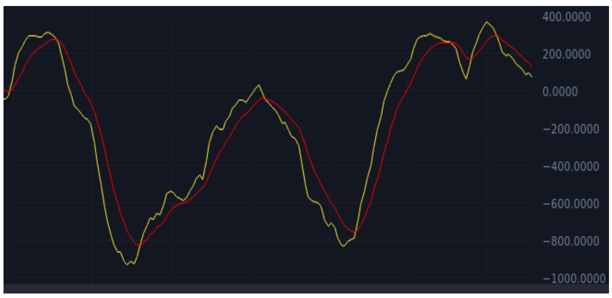

Figure 3. MACD and signal line resulting from Figure's 1 data

3.3 3-Period Integrated Relative Strength Index

3-Period Integrated Relative Strength Index is assembled by first computing RSI signal, which results in a data vector denoted as D in (7). Consequently, (8) denotes that D is divided into 3 semi-overlapping series of a semi-varied length which are subsequently normalized using (9) and integrated using Riemann sums.

$$D = \begin{bmatrix} d_0 & d_1 & \cdots & d_{n-1} \end{bmatrix} \tag{7}$$

$$VEC(D)_{\alpha,\beta,\gamma} = \begin{bmatrix} (RIEMANN \circ NORM_{0,\alpha})(D) \\ (RIEMANN \circ NORM_{0,\beta})(D) \\ (RIEMANN \circ NORM_{0,\gamma})(D) \end{bmatrix} = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix} = V \tag{8}$$

$$NORM(D)_{x,y} = \begin{bmatrix} \begin{bmatrix} \frac{x+0}{y-x} \\ \frac{d_{x+0}}{100} \end{bmatrix} \begin{bmatrix} \frac{x+1}{y-x} \\ \frac{d_{x+1}}{100} \end{bmatrix} \begin{bmatrix} \frac{x+2}{y-x} \\ \frac{d_{x+2}}{100} \end{bmatrix} \cdots \begin{bmatrix} \frac{y}{y-x} \\ \frac{d_y}{100} \end{bmatrix} \end{bmatrix} \tag{9}$$

Equation (10) uses the resulting pressure vector V in order to compute the final decision vector B, which in itself is achieved using some simple Boolean rules. This in theory should detect direction of a trend, and be more resistant to sudden irregularities present within the RSI output. Finally, (11) generates final trading decisions.

$$D\_VEC(V) = \begin{bmatrix} v_2 > v_1 \wedge v_1 > v_0 \\ v_2 < v_1 \wedge v_1 > v_0 \\ v_2 < v_1 \wedge v_1 < v_0 \\ v_2 \geq v_1 \wedge v_1 \leq v_0 \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = B \tag{10}$$

$$DECISION(B) = \begin{cases} BUY & \text{if } b_0 \vee b_1 \\ SELL & \text{if } b_2 \\ WAIT & \text{if } b_3 \end{cases} \tag{11}$$

It should be pointed out that decision rules presented in (10) are not final, as there are many equally valid alternatives. As an example D_VEC could utilize some scaling parameter Ω ensuring the appropriate numerical differences between two or more pressure vectors.

3.4 Particle Swarm Optimization

Particle Swarm Optimization is a computationally inexpensive and effective [13] way of optimizing nonlinear functions [12]. Which is proven to work with dynamic systems [9, 10] at the same being independent of its gradient. This differentiates PSO from other methods such as gradient descent [21], subsequently making it deal for our purposes. As due to the number of potential variables, we cannot reasonably identify market's function.

Native PSO works by generating a set of P particles. Each particle p is capable of retaining its current coordinates d_p, communicating its local best positions c_p and reacting to and electing $global\_best$. Where best is defined as a set of coordinates minimizing a certain user provided $UTILITY$ function. This, in turn, allows them to effectively explore given search space as each particle's velocity v_p represents among the others a product of c_p and $global\_best$.

Our experimental setup which is based on Cohen's work [6] can be seen in Figure 4. In the following paragraphs we are going to explain the main differences between it and that of native PSO.

Firstly, each particle's velocity and location has to be extended up to d dimensions, where the exact number is determined by a number of subscript variables present within

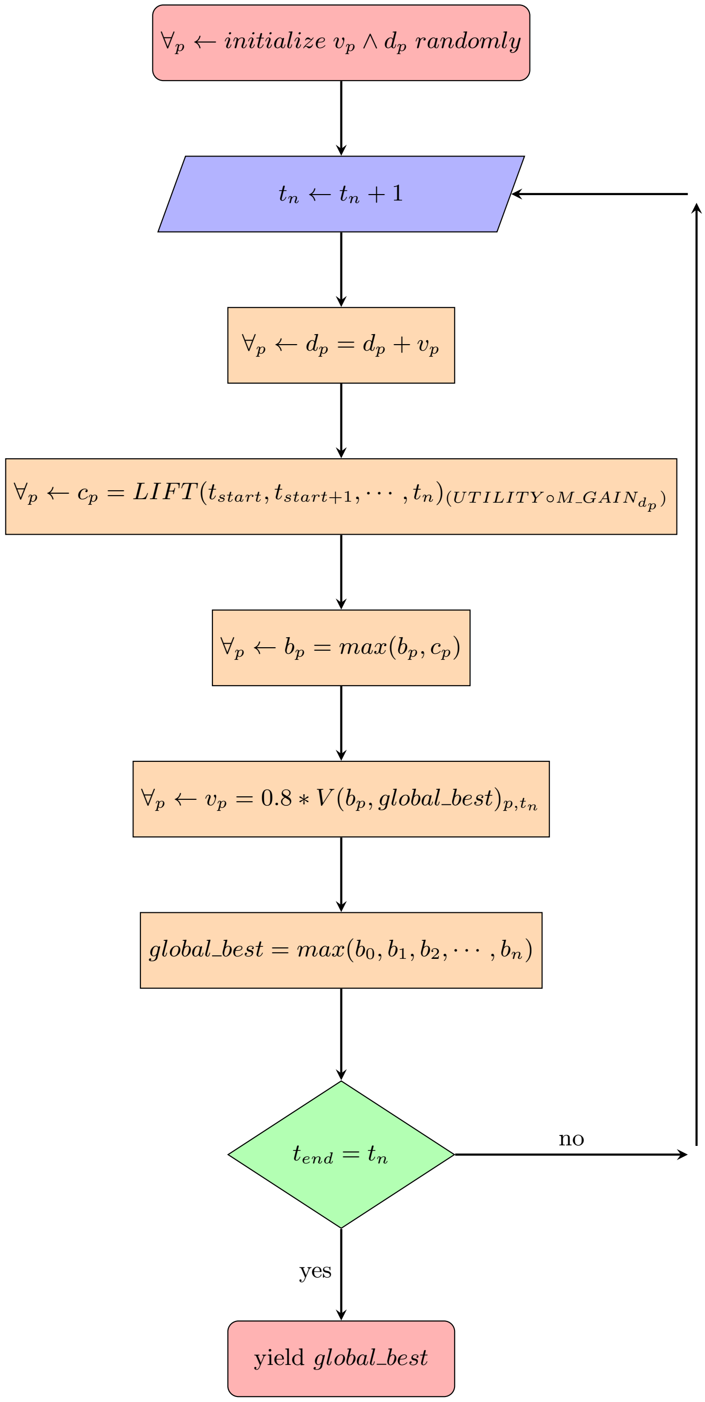

Figure 4. Modified PSO's flow chart diagram

the optimized trading strategy. This result in one dimension for RSI, three for MACD and four for 3-IRSI (one additional dimension is for the RSI itself). Subsequently, we use this information in order to update particle's velocity V formula, which can be seen in (12).

$$V(b_n, g)_{n,t} = v_{n,t-1} + \begin{bmatrix} \beta \\ \vdots \\ d \text{ times} \\ \vdots \\ \beta \end{bmatrix} \begin{bmatrix} RAND(0,\alpha) \\ \vdots \\ d \text{ times} \\ \vdots \\ RAND(0,\alpha) \end{bmatrix} b_n + \begin{bmatrix} \gamma \\ \vdots \\ d \text{ times} \\ \vdots \\ \gamma \end{bmatrix} \begin{bmatrix} RAND(0,\alpha) \\ \vdots \\ d \text{ times} \\ \vdots \\ RAND(0,\alpha) \end{bmatrix} g \quad (12)$$

Secondly, as our setup utilizes multi-objective $UTILITY$ function. The resulting value will be maximized instead of minimized [6]. The $UTILITY$ function's formula is almost the same as presented by Cohen (figure 5A) [6] and can be seen in (13). Chosen values were selected in order to ensure $UTILITY$'s primary focus on minimizing maximum loss experienced by the portfolio. At this point we have to point out that Cohen's original equation contains an error. As certain border cases such as systems generating as many gains as losses would result in computing natural logarithm of zero, which is undefined. This issue is subsequently resolved in (14).

$$UTILITY(Value) = sum \left\{ \begin{bmatrix} 0.6 \\ 0.2 \\ 0.2 \end{bmatrix} \begin{bmatrix} (FIX \circ OMDD)(Value) \\ (FIX \circ PP)(Value) \\ (FIX \circ NPF)(Value) \end{bmatrix} \right\} \quad (13)$$

$$FIX(Value) = \begin{cases} ln(Value) & \text{if } Value \neq 0 \\ 0 & \text{if } Value = 0 \end{cases} \quad (14)$$

Similarly formulas for NPF and $OMDD$ are the same as presented by Cohen (equations 3 and 4 respectively) [6], and are provided here for the sake of completeness as (15) and (16). Cohen defines PP as simply "percentage of profitable trades in relation to all trades" [6], which is something we adhere to.

$$OMDD(Value) = 100 - MDD\_PERCENTAGE(Value) \quad (15)$$

$$NPF(Value) = (PF(Value) - 1) * 100 \quad (16)$$

Whereas, $LIFT$ adapts function composition of M_GAIN with $UTILITY$ such that it can accept a list of timestamps and process them in order. At the same time ensuring that previously processed timestamps are disregarded. This modification allows our system to simulate a real world scenario in which new data is processed as soon as it is available.

Because of the size of the total solution space the experiment was re-run multiple times using different coordinate cut-off values which were chosen on an arbitrary basis. Consequently, the best setups are reported.

4. RESULTS

In this section we present outcomes of applying Particle Swarm Optimization to RSI, MACD and 3-IRSI. Subsequently we discuss their performance and use this information in order to answer research questions.

4.1 Data

Resulting data can be found in Tables 1, 2 and 3. Each table is divided into four vertical categories. The first category specifies setup's number; second is used to indicate strategy's parameters; third reports value of the $UTILITY$ function, Opposite Maximum Drawdown, Net Profit Factor and Profitable Percentage denoted as U, OMDD, NPF and PP correspondingly; the fourth category is used exclusively for the purpose of reporting Net Profits (NP), which itself is in euros. Furthermore, each table is divided into two horizontal categories. First one is used for reporting setup's specific data, whereas second is used for summarizing it by the means of descriptive statistics. The numerical data are rounded to two decimal places and the best setups in terms of NP are further emphasised using bold characters.

As **buy-and-hold** strategy was not a subject to the Particle Swarm Optimization process, it will not be found in any of the tables. That is why its net profit of **10384.05** is reported here.

4.2 Research Question I

Table 3 demonstrates that 3-IRSI is capable of delivering substantial gains. The most profitable setups, namely 3 and 4 yield about 20584 euros of net profit. Interestingly enough even though we attempted to maximize OMDD, those two setups are not the ones maximizing it, which is achieved by setup 8 instead; which at the same time achieves slightly lower profits. Moreover setup 2 which achieved the second highest OMDD value performs substantially worse than setups 3, 4 and 8. At the same time setup's 8 NPF is much higher than that of setup 3 or 4. This implies a tendency of those two systems to gain and at the same time lose greater amount of assets due to suboptimal trading choices. Finally, PP of each system demonstrates that even though they executed profitable trading decisions less than half a time, it was nonetheless sufficient to achieve profits far outstretch buy-and-hold strategy.

To summarize, six out of eight 3-IRSI setups managed to outperform buy-and-hold strategy, with best systems outperforming it by as much as 98%. At the same time the worst setup in terms of NP, namely setup 7 achieved only about 10% of its contender profits. This illustrates that under chosen methodology certain 3-IRSI setups are capable of substantially outperforming the buy-and-hold strategy. This answers **RQI**.

4.3 Research Question II

Table 1 illustrates that RSI setup number 1 achieved the highest net profit. At the same its $UTILITY$ function value is the same as that of the second setup, which achieved almost three times lower profit. The comparable $UTILITY$ function value is to be attributed to much higher PP. Interestingly enough in the case of RSI, higher values of PP do not appear to directly translate into higher net profits; which is clearly demonstrated by setups number 1, 7 and 8. Similarly to 3-IRSI setups with the highest NPF such as setup 6 are not the ones achieving the most profit. Moreover, unlike 3-IRSI; setup attaining the lowest value of OMDD namely setup 3, does not translate to a lowest net profit as that belongs to setup number 4.

Whereas MACD statistics depicted in Table 2, demonstrate that setup 6 achieved the highest profit and at the same time has the lowest PP and the highest OMDD. Unfortunately the profit achieved by this setup is substantially lower than that of RSI or 3-IRSI. Moreover, overall inferior performance of the considered setups di-

Setup	**X**	**U**	**OMDD**	**NPF**	**PP**	**NP**
1	**2**	**4.05**	**83.34**	**18.85**	**58.03**	**11661.93**
2	5	4.05	77.75	20.01	66.54	3866.96
3	10	3.06	37.41	1.30	65.40	77.03
4	11	3.01	38.04	-7.87	62.72	-355.67
5	21	3.12	45.14	-1.09	65.96	-35.55
6	25	3.81	50.41	23.15	63.38	652.36
7	28	3.15	49.00	-2.06	59.18	-52.4
8	35	3.65	43.92	14.63	68.97	351.39
Min	2	3.01	37.41	-7.87	58.03	-355.67
Max	35	4.05	83.34	23.15	68.97	11661.93
Mean	17.13	3.49	53.13	8.37	63.77	2020.76
Variance	139.84	0.20	309.53	144.99	13.92	16995976.75
SD	11.83	0.45	17.59	12.04	3.73	4122.62

Table 1. RSI

Setup	**X**	**Y**	**Z**	**U**	**OMDD**	**NPF**	**PP**	**NP**
1	8	15	13	2.80	31.87	-2.29	37.46	-314.72
2	14	15	7	2.75	29.09	-2.65	37.71	-351.71
3	15	24	3	2.75	28.74	-1.16	38.96	-196.48
4	17	26	5	2.91	38.31	-0.27	36.91	-45.64
5	18	19	5	2.80	31.90	-1.36	37.52	-205.29
6	**21**	**31**	**11**	**3.48**	**67.58**	**3.25**	**35.20**	**610.34**
7	25	15	10	3.02	46.68	-0.90	35.40	-131.35
8	25	28	10	3.10	45.33	1.67	35.25	248.64
Min	8	15	3	2.75	28.74	-2.65	35.20	-351.71
Max	25	31	13	3.48	67.58	3.25	38.96	610.34
Mean	17.88	21.63	8	2.95	39.94	-0.46	36.80	-48.28
Variance	33.27	41.70	12.29	0.06	173.43	4.00	1.91	105503.54
SD	5.77	6.46	3.51	0.25	13.17	2.00	1.38	324.81

Table 2. MACD

Setup	**X**	**Y_α**	**Z_β**	**K_γ**	**U**	**OMDD**	**NPF**	**PP**	**NP**
1	10	34	19	12	4.05	75.07	32.67	45.76	13389.60
2	10	39	17	12	4.06	75.12	32.80	46.11	12461.70
3	**12**	**30**	**17**	**12**	**4.07**	**74.70**	**34.43**	**47.25**	**20584.00**
4	**14**	**30**	**17**	**12**	**4.07**	**74.70**	**34.43**	**47.25**	**20584.00**
5	14	31	19	11	3.97	67.16	29.82	47.38	19272.10
6	14	34	10	14	3.98	74.75	23.83	44.32	8829.14
7	15	34	10	14	3.64	54.56	11.05	43.98	1005.13
8	15	34	15	12	4.20	86.00	44.29	47.08	17918.20
Min	10	30	10	11	3.64	54.56	11.05	43.98	1005.13
Max	15	39	19	14	4.20	86.00	44.29	47.38	20584.00
Mean	13	33.25	15.5	12.38	4.01	72.76	35.84	46.14	14255.48
Variance	4.29	8.79	13.14	1.13	0.03	80.04	355.66	1.86	46809522.94
SD	2.07	2.96	3.63	1.06	0.16	8.95	18.86	1.36	6841.75

Table 3. 3-IRSI

rectly maps to their considerably lower $UTILITY$ function value, with its mean value oscillating around three. This contrasts with a mean value of 4.01 for 3-IRSI and 3.49 for RSI. This substantial difference is a direct result of much lower PP, NPF and OMDD values. Most notably NPF mean value is negative, which directly translates to a negative NP, where as opposite is true for RSI and 3-IRSI. At the same time OMDD mean value of 39.94 implies that the systems utilizing MACD can result in portfolios experiencing a draw down of as much as 60.06%.

Previously conducted analysis reveals that 3-IRSI appears to outperform RSI in terms of NP and NPF, with best setups experiencing as much as 76.5% greater profits. Whereas RSI managed to achieve much higher PP. Moreover even though RSI's OMDD values show more spread, they are capable of approaching the values experienced by the 3-IRSI. When it comes to comparing performance of 3-IRSI to MACD, the former appears to outperform the latter in all considered metrics. This answers **RQII**.

5. DISCUSSION

Our findings suggest that under the assumption of frictionless market; 3-IRSI is capable of outperforming RSI, MACD as well as buy-and-hold strategy. Unfortunately due to the short time span given to conduct this study, relatively small amount of data could be gathered; as a result the exact magnitude of this phenomenon remains largely unknown. Nonetheless, current results indicate it is possible to build upon already existing technical indicators and achieve even better performance in terms of net profit.

At the same time, data generated by this experiment results in some secondary findings. First of all, we have to note the apparent suboptimality of the $UTILITY$ function in regards to maximizing 3-IRSI's NP. As higher U values do not necessarily translate to higher NPs. This indicates that current setup can be further refined by either incorporating more metrics or by trying different input parameters.

Whereas, the high performance of the RSI setup number 1, supports the previously held notion of a highly volatile Bitcoin price action [2]. As setups with a sorter look-back period are known to perform better in more volatile markets [23]. Moreover, $GAIN\_LOSS$ used as a data truncation method paired with the RSI indicator appears to support Cohen's previous observation about unsuitability of the original RSI setup for trading Bitcoin and at the same time extends it to one hour resolution [6]. This is due to presence of this setup within multiple concurrent runs, yet resulting particles were continuously drifting away from $X = 14$ to either higher or lower input values. Which can be observed by inspecting U and X values of setups number 3, 4 and 5.

The overall weak performance of the MACD indicator could be a direct result of the chosen data truncation method. Which future studies should try to verify by using either $CLOSE$ or $\frac{OPEN+CLOSE}{2}$ instead of $GAIN\_LOSS$. Another possible explanation is its unsuitability for trading at this temporal resolution, this could be verified by repeating current experiment using OHLCVs of different time resolutions.

Finally, as this experiment was limited only to friction-less markets, more research is needed in order to verify wherever 3-IRSI is capable of delivering similar performance without this assumption.

6. CONCLUSION

In this study we have assembled a custom RSI based indicator called 3-IRSI which was then optimized using particle swarms and subsequently evaluated against buy-and-hold strategy as well as PSO optimized RSI and MACD. The resulting algorithm managed to outperform its peers. Unfortunately, due to the time allocated for the study, more data are needed in order to know the exact magnitude of this phenomenon. Nonetheless, current research indicates that it is possible to build upon already existing technical indicators in order to achieve even better performance. As a result more future research should take this route.

APPENDIX

A. Glossary

bear market market trending down. 1

bull market market trending up. 1

friction-less market market with no commission fees. 2, 5, 6

B. REFERENCES

[1] G. Appel. *Technical analysis: power tools for active investors.* FT Press, 2005.

[2] C. Baek and M. Elbeck. Bitcoins as an investment or speculative vehicle? a first look. *Applied Economics Letters*, 22(1):30–34, 2015.

[3] R. Böhme, N. Christin, B. Edelman, and T. Moore. Bitcoin: Economics, technology, and governance. *Journal of economic Perspectives*, 29(2):213–38, 2015.

[4] A. C. Briza and P. C. Naval Jr. Stock trading system based on the multi-objective particle swarm optimization of technical indicators on end-of-day market data. *Applied Soft Computing*, 11(1):1191–1201, 2011.

[5] D. P. Brown and R. H. Jennings. On technical analysis. *The Review of Financial Studies*, 2(4):527–551, 1989.

[6] G. Cohen. Optimizing algorithmic strategies for trading bitcoin. *Computational Economics*, 57(2), 2021.

[7] S. Corbet, V. Eraslan, B. Lucey, and A. Sensoy. The effectiveness of technical trading rules in cryptocurrency markets. *Finance Research Letters*, 31:32–37, 2019.

[8] A. Detzel, H. Liu, J. Strauss, G. Zhou, and Y. Zhu. Bitcoin: Predictability and profitability via technical analysis. *SSRN Electronic Journal*, pages 1–35, 2018.

[9] R. Eberhart, P. Simpson, and R. Dobbins. *Computational intelligence PC tools.* Academic Press Professional, Inc., 1996.

[10] R. C. Eberhart and Y. Shi. Tracking and optimizing dynamic systems with particle swarms. In *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, volume 1, pages 94–100. IEEE, 2001.

[11] J.-Z. Huang, W. Huang, and J. Ni. Predicting bitcoin returns using high-dimensional technical indicators. *The Journal of Finance and Data Science*, 5(3):140–155, 2019.

[12] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE, 1995.

[13] A. Kumar, S. Pant, M. Ram, and S. Singh. On solving complex reliability optimization problem using multi-objective particle swarm optimization. In *Mathematics Applied to Engineering*, pages 115–131. Elsevier, 2017.

[14] J. H. Kwon. Tail behavior of bitcoin, the dollar, gold and the stock market index. *Journal of International Financial Markets, Institutions and Money*, 67:101202, 2020.

[15] A. W. Lo and J. Hasanhodzic. *The evolution of technical analysis: financial prediction from Babylonian tablets to Bloomberg terminals*, volume 96. John Wiley & Sons, 2010.

[16] W. J. Luther. Regulating bitcoin: On what grounds? *Luther, William J*, 2016.

[17] J. J. Murphy. *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications.* Penguin, 1999.

[18] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260, 2008.

[19] M. Nakano, A. Takahashi, and S. Takahashi. Bitcoin technical trading with artificial neural network. *Physica A: Statistical Mechanics and its Applications*, 510:587–609, 2018.

[20] M. Prentis. Digital metal: Regulating bitcoin as a commodity. *Case W. Res. L. Rev.*, 66:609, 2015.

[21] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[22] J. W. Wilder. *New concepts in technical trading systems*. Trend Research, 1978.

[23] W.-K. Wong, M. Manzur, and B.-K. Chew. How rewarding is technical analysis? evidence from singapore stock market. *Applied Financial Economics*, 13(7):543–551, 2003.