2021 12th International Conference on Information and Knowledge Technology (IKT)

SPA Bot: Smart Price-Action Trading Bot for Cryptocurrency Market

Hamid Jazayeriy*
Department of Computer Engineering
Babol Noshivani University of Technoloy
Babol, Iran
JHamid@nit.ac.ir

Mohammad Daryani
Department of Computer Engineering
Babol Noshivani University of Technoloy
Babol, Iran
1mo.daryani@gmail.com

2021 12th International Conference on Information and Knowledge Technology (IKT) | 978-1-6654-0849-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/IKT54664.2021.9685662

***Abstract*— Cryptocurrencies can be traded 24 hours a day over the globe where crypto market never stop working. This market is known for being highly volatile and prices fluctuate rigorously in a minute. Even seasoned traders are unable to react quickly enough to this volatility. This is why automated trading bots get into the picture. On the other hand, crypto trading bots need more improvement to mimic expert traders' activities. In this paper, we propose a bot which is able to buy and sell using a dynamic price-action technique. Experimental results from back-testing show that the proposed bot effectively utilized the classical price-action technique for trading in cryptocurrency markets.**

Keywords*— cryptocurrency, automated trading, Bot, price-action, active true range

I. Introduction

Blockchain technology brings new business opportunities while the market capital of its related industries are growing dramatically in recent years. Cryptocurrencies are getting more popular and many individuals and companies invest in this 24×7 market. However, due to the high volatility in crypto market, trading is a very risky profession and many naïve traders lose their money. In addition, seasoned traders cannot work 24 hours a day, and they miss some trading opportunities [1].

Automated crypto trading bots are set of computer programs developed to automate pre-defined trading strategies by sending sell/buy orders to exchanges on behalf of their owners. They never miss the trading opportunities in 24 hours. Moreover, they never succumb to fear and greed, or any other human emotions. Both advanced traders and beginners can use automated trading bots based on their needs.

The quality of trading bots depend on the way they utilize artificial intelligence techniques to maximize the profit while minimizing the risk [2]. Smart bots use artificial intelligence to analyze and interpret market statistics before making decisions.

In general, artificial intelligence techniques can help trading bots to improve trading strategies, finding best setup for technical functions, and risk management [3]. In this study, we propose an automated trading bot based on the price-action technique. Price-action is a very popular trading technique among seasoned traders.

The rest of the paper is organized as follows: the next section reviews related works. Then, the proposed SPA bot is presented in section III, followed by its evaluation. Finally, section V presents our conclusions.

II. Related Work

Although the topic of automated trading in crypto market is an emergent field of study, there are a few valuable studies that will be reviewed in this section.

Traditional trading systems can be developed based on some well-known techniques which are popular among seasoned traders [4]. These techniques are usually based on some statistical models. The first generation of trading bots have been developed based on these models. The results from this bots showed that they are not trustworthy in all situations and may cause some loses. The second generation of trading bots tries to utilize trading algorithms by using artificial intelligence techniques.

Reinforcement learning (RL) is one of powerful AI tools where an agent (say bot) can learn from its actions [5]. Rewards and punishments help to find the best reaction to the environment. There are some tested RL-based agents that can be adapted in any environment. In [6], different RL-based agents (such as A1C, A2C, APPO, IMPALA and PPO) have been used for automated trading on BTC, BAT and NANO markets. They concluded that A2C was the best reinforcement learning algorithm for automated trading. Moreover, they found that cryptocurrencies with low capitalization (like NANO) are not suitable for algorithmic trading. Unfortunately, they found that the *buy and hold* strategy, despite its simplicity, outperforms all RL-based trading agents.

Prediction of cryptocurrencies in low time-frames can help bots for scalping or option trading. In [7], an artificial neural network (ANN) based machine proposed for algorithmic trading in binary option markets. In binary options, bots can win a reward for correct prediction of the future price movements or loss some money for wrong guess. The researchers, designed an algorithm for option-trading in 5-min time frame where a bot should predict the next 5-min price as up or down binary. They showed that the best overall model was a deep-MLP with 54% accuracy for predicting of the next up or down classification problem in BTC market. Unfortunately, the accuracy of 54% in not good enough for option trading in 5-min time frame.

In short, automated trading in crypto market is very young and needs more effort to reach high quality bots [8].

III. Proposed Method

To develop an automated trading bot, the following component should be considered: capital management, risk management and technical analysis [9], [10]. Each of these components can be improved by using artificial intelligence accordingly.

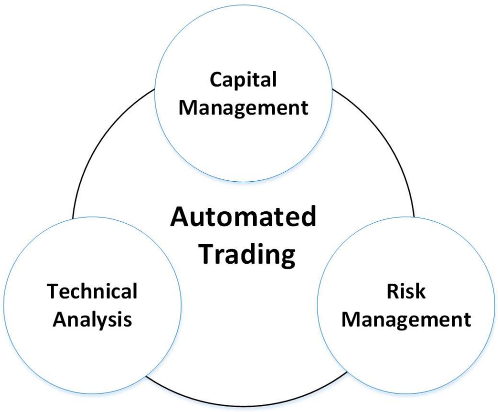

Fig 1. Elements of a generic trading bot.

Technical analysis usually helps to answer two crucial questions: when to enter the market by send buy/sell orders? And, when to close the active orders?
Traditional algorithms for trading usually apply fixed parameter in all time-frames. Finding the best performance of an algorithm needs artificial intelligence techniques to adapt its parameters in each time-frame.
In this study, a price-action based technique is proposed and utilized for automated trading by using artificial intelligence. Therefore, the main focus of this study is technical analysis. The proposed technique is named *smart price-action* (SPA). The rest of this section is illustrating the steps in the proposed price-action algorithm (SPA).

A. *Proposed Price-Action*

Algorithm 1 describes the steps of a generic price action. Finding a proper binderies for price movement is focal point of price-action technique.

***Algorithm 1**. Price-action*	
1	*Find upper and lower price boundaries*
2	*Select an active boundary*
3	*Wait for rejection or breakout*
4	*Check for confirmation*
5	*Enter a position*

The first three steps are related to the upper and lower boundaries.
Price in price-action is normally limited between to horizontal pivots. The upper pivot is called resistance and the lower one is call support.
Fig. 1 shows a sample boundaries. If price touches upper boundary, then two scenarios is considered:
(*i*) Rejection and return into the moving channel. In this case, a traders can get a sell position.
(*ii*) Breakout the channel. In this case, price trying to reach a new higher high and it gives an opportunity to get a buy position.
Similarly, a buy and sell opportunity may be occurred when price tested the lower boundary.

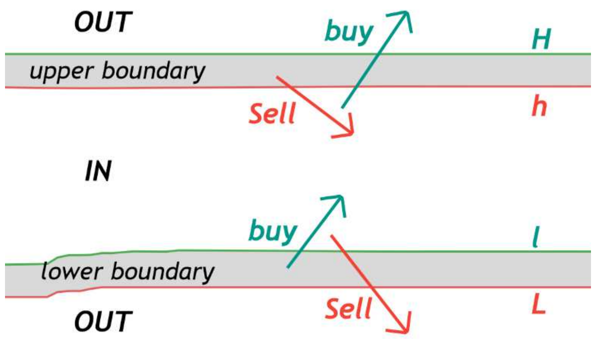

Fig 2. Dynamic upper and lower boundaries in smart price action.

Each boundary in Fig.2 is presented by an area surrounded by two lines. One of them presents the inner margin (lower case letters: *h, l*) and the other one presents the outer border of the area (upper case letters: H, L).
In this study we propose a dynamic boundaries based on the concept of active true range. The active swing (movement) can be calculated based on *high* and *low* values in Japanese candle stick.

A typical Japanese candle stick is presented by four points: *low*, *high*, *open* and *close*. Fig. 3 shows a typical Japanese candle stick.

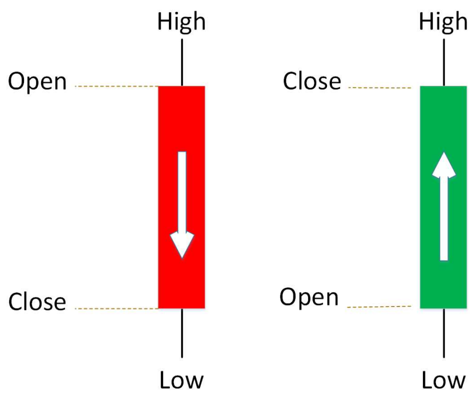

Fig 3. Japanese candle stick.

we have proposed a dynamic areas based on the concept of active true range (ATR) to present upper and lower boundaries needed for price-action. Active true range (ATR), is the average of swing values in the recent market depth (say n). In this study, μ and δ are used to represent the average and standard deviation of swing values, respectively.

$$\mu = ATR(n) = \frac{1}{n}\sum_{i=1}^{n}(high_i - low_i) \tag{1}$$

$$\delta = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|(high_i - low_i) - \mu|^2} \tag{2}$$

The upper boundary of the current price is presented by h and H values. Where h and H are the high inner and the high outer boundaries, respectively.

$$h = highest(close, n) - \alpha.\delta \tag{3}$$

$$H = max(highest(high, n), \quad h + margin) \tag{4}$$

The $margin$ value is defined as follows:

$$margin = \mu + \beta.\delta \tag{5}$$

Similarly, the lower boundary of the current price is presented by l and L values. Where l and L are the low inner and the low outer boundaries, respectively.

$$l = lowest(close, n) + \alpha.\delta \tag{6}$$

$$L = min(lowest(low, n), \quad l - margin) \tag{7}$$

In the above equations there are two parameters (α, β) which help to utilize the dynamic boundaries using soft-computing techniques.

Changing on the price value, may change the *low* and *high* values of the current candle, and ATR consequently. Therefore, the boundaries may change dynamically.

After finding the dynamic boundaries using active true range. The proposed algorithm waits for touching the boundaries by price movement. The following condition will be examined for sell and buy opportunities.

$$sell = \begin{cases} (close < h) \text{ and } (open > h) & rejection \\ or & \\ (close < L) \text{ and } (open > L) & breakout \end{cases} \tag{8}$$

$$buy = \begin{cases} (close > l) \text{ and } (open < l) & rejection \\ or & \\ (close > H) \text{ and } (open < H) & breakout \end{cases} \tag{9}$$

Fig. 4 demonstrates the performance of the proposed algorithm on VET/USDT spot market in 5 min time frame in Binance when rejection occurs.

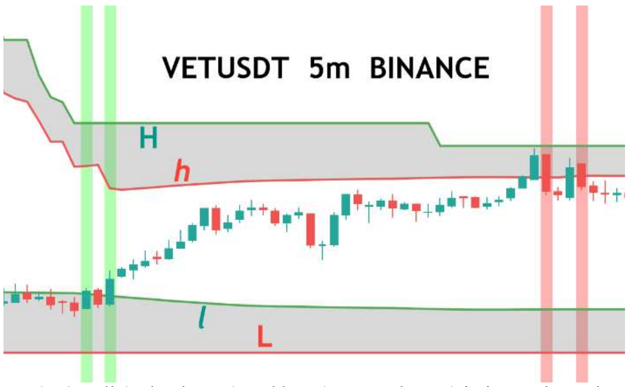

Fig 4. Sell (red columns) and buy (green columns) in inner channel when rejection occurs..

Fig. 5 demonstrates the performance of the proposed algorithm on VET/USDT spot market in 1 min time frame in Binance when breakout occurs from L border where a sell order can be considered. When breakout occurs a new l and L border will get into the picture.

Having sell/buy opportunity is not enough to enter a position. In other word, to increase the hit rate of the algorithm a confirmation is needed.

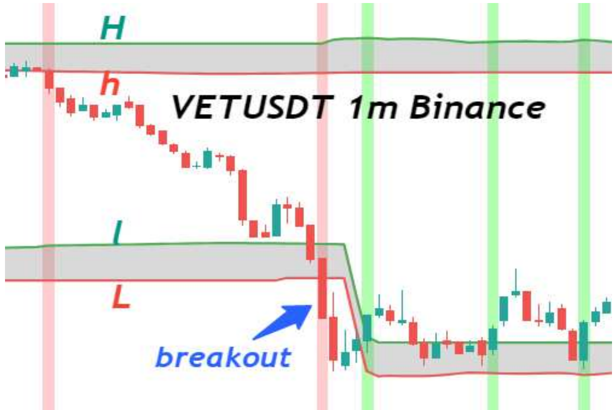

Fig 5. A breakout from outter border (L) shows a sell (red column) opportunity.

Next step in the proposed algorithm is the decision confirmation. A simple yet effective technique for confirmation is using moving average. Equation (10) show moving average of the given depth *d* based on the source (src) values.

$$ma(d) = \frac{1}{d}\sum_{i=1}^{d} src_i \tag{10}$$

The source value can be one of the followings: *close*, *hlc3*, or *hl2*, where $hlc3 \text{ is } \frac{1}{3}(high + low + close)$ and $hl2 \text{ is } \frac{1}{2}(high + low)$, respectively.

To utilize the algorithm we need to choose one of these sources as well as choosing the optimized depth *d* value.

Confirmation for buy and sell decisions can be made by using the following conditions.

$$confirmation = \begin{cases} close > ma(d) & buy \\ close < ma(d) & sell \end{cases} \tag{11}$$

Fig. 6 shows the effect of confirmation using moving average on VET/USDT spot market in 4H time frame in Binance. The candle which shows buy or sell decision should be crossed by a short moving average ($d = 5$).

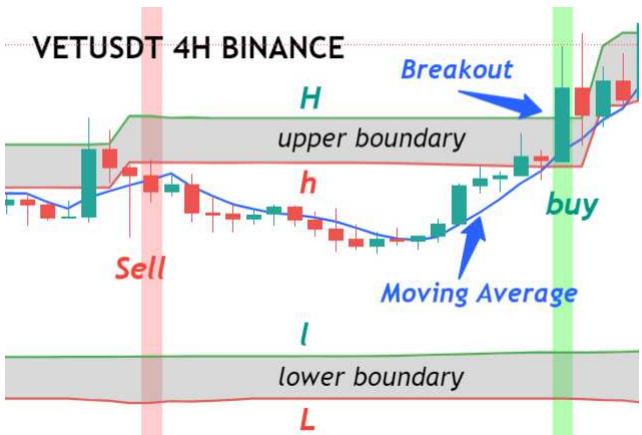

Fig 6. The effect of confirmation using moving average ($d = 5$).

Having long moving averages may reduce the chance of trade opportunities while short moving averages may lead the algorithm to a false detection (fake breakouts or rejections).

B. Utilizing the SPA using artificial intelligence

The quality of the proposed SPA algorithm depends on its parameters: n, α, β, d and src. In this study, a genetic algorithm (GA) is used to find the optimal values while back-testing the market. Each chromosome defined as set of 5 genes containing the following parameters:

n: depth of the market for calculating boundaries

α: coefficient in (3) and (6)

β: coefficient in (5)

d: length of confirmation moving average

src: data source for moving average

The search space of this optimization problem is very large. To reduce the search space, the domain of each parameter is limited by Fibonacci numbers and coefficients as follows:

$n \in \{13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597\}$

$\alpha \in \{1, 2, 3, 5, 8, 11\}$

$\beta \in \{0.38, 0.5, 0.61, 1, 1.44, 1.61, 2.61\}$

$d \in \{2, 3, 5, 8, 13\}$

$src \in \{close, hl2, hlc3\}$

The best value for this parameters should be finded by GA. In the next section, the experimental setup and results of the proposed method are explained.

IV. EVALUATION

To evaluate the quality of the proposed SPA algorithm, a trading setup needed. Accordingly, the BTCUSDT and VETUSDT are selected as two cryptocurrency markets for evaluation the SPA. Moreover, back-testing are taken in the last two years data provided in TradingView's website. At first, the parameters of the algorithm are calibrated, and then the win rate and the frequency of the trades in each time frame are evaluated.

The total profit of the investment depends on liquidity management as well as SPA algorithm and the risk management. The main goal of this study the evaluation of the proposed SPA algorithm. Therefore, the concept of the liquidity management is excluded from this study.

The quality of the proposed algorithm is related to its parameters' values. To find the best parameter setting we need to run algorithm and find the win rate of the submitted orders. By considering a risk to reward ratio of 1:2, the following formula is used to find the amount of stop-loss and take-profit in each time frame:

$$\Delta = \left[\frac{1}{200}\sum_{i=1}^{200}(high_i - low_i)/open_i\right] \times close \tag{12}$$

$$StopLoss = \begin{cases} close - \Delta & buy \\ close + \Delta & sell \end{cases} \tag{13}$$

$$TakeProfit = \begin{cases} close + 2\Delta & buy \\ close - 2\Delta & sell \end{cases} \tag{14}$$

Instead of using a fix value (or percentage) for stop-loss and take-profit, we have proposed a dynamic value, Δ, based on the average changes (price fluctuation) in the recent 200 candles. Equation (12) helps to find Δ, and set stop-loss and take-profit dynamically. Then, (13) and (14) are used to calculate the stop-loss and take-profit for buy and sell accordingly.

In this study, we have used a fixed value for risk and reward ratio $(\Delta, 2\Delta)$ to explore the win rate of the proposed SPA algorithm. This setting is taken for the simplicity of the ruin experiments and can be considered as a drawback of this study.

The win rate can be calculated by the following formula:

$$w = \frac{\#closed\ with\ profit}{total\ number\ of\ trades} \tag{15}$$

Together with win rate, the total number of trades should also be considered.

Table 1 shows the optimum parameter values for the proposed SPA algorithm. Each time frame needs to be setup using a set of different parameters' value to show its best performance.

Table 1. Parameter setting using evolutionary computing.

Time Frame	Parameters				
	n	α	β	d	src
1m	377	5	1.61	8	hlc3
5m	233	5	1.44	5	hlc3
30m	144	5	1	3	close
1H	89	3	1	2	close
4H	55	3	0.5	2	close
1D	34	2	0.5	2	close

The win rate of the trading algorithm is reported in Table 2. As it can be seen, in lower time frames the algorithm has lower win rate and the maximum win rate is reported in 1D time frame. On the other hand, the average number of trades per day in 1D is very low while the lower time frames have much more frequent trades. Table 3 reports the average number of trades per day for each time frame.

Table 2. Average win rate of the proposed SPA algorithm.

Time Frame	Win rate (%)	
	BTC	VET
1m	52	54
5m	63	61
30m	70	69
1H	73	74
4H	78	79
1D	81	82

Tables (2) and (3) show that high trading frequency would not guarantee the maximum win rate. To reach the best performance for SPA algorithm, we need to find the best time frame where the maximum profit is available.

Table 3. Number of trades per day.

Time Frame	frq	
	BTC	*VET*
1m	112	117
5m	21.42	22.05
30m	6.25	6.77
1H	1.62	1.73
4H	0.45	0.48
1D	0.085	0.09

As it mentioned, we have used a fixed risk to reward ration to evaluate the win rate of proposed algorithm. Preliminary observations of the running algorithm in higher time frames show that the reward ratio can be considered much bigger than what assumed in this study.

V. CONCLUSION

In this study, the smart price-action (SPA) algorithm is proposed for automated trading in the cryptocurrency market. This algorithm uses dynamic boundaries to keep track of the price movement and take a proper decisions (buy/sell) in the market. To optimize the trading parameters, a genetic algorithm (GA) is used. The experimental results show that the SPA has 81% win rate in daily (1D) time frame while the chance of having a daily trade is less than 0.1. On the other hand, the win rate for 1m time frame is about 50% and more than 100 trade situations may occur in this time frame.

We have used Fibonacci numbers to reduce the search space for finding the optimum values of parameters in the proposed algorithm. In the future, a more accurate and comprehensive study is needed to find the parameters' optimum value. Moreover, for the order confirmation, we have used the moving average while it may be improved by other technical indicators. Finally, finding a proper risk and liquidity management that suited for the proposed SPA algorithm can be an interesting field of study and practice.

Acknowledgement: This research work was partly supported by the *bitbotec.com*.

VI. REFERENCES

[1] Y. Andrianto and Y. Diputra, "The effect of cryptocurrency on investment portfolio effectiveness," *J. Financ. Account.*, vol. 5, no. 6, pp. 229–238, 2017.

[2] I. U. Haq, A. Maneengam, S. Chupradit, W. Suksatan, and C. Huo, "Economic Policy Uncertainty and Cryptocurrency Market as a Risk Management Avenue: A Systematic Review," *Risks*, vol. 9, no. 9, p. 163, 2021.

[3] S. Saksonova and I. Kuzmina-Merlino, "Cryptocurrency as an investment instrument in a modern financial market," *Вестник Санкт-Петербургского университета. Экономика*, vol. 35, no. 2, 2019.

[4] B. Huang, Y. Huan, L. Da Xu, L. Zheng, and Z. Zou, "Automated trading systems statistical and machine learning methods and hardware implementation: a survey," *Enterp. Inf. Syst.*, vol. 13, no. 1, pp. 132–144, 2019.

[5] M. A. H. Dempster and V. Leemans, "An automated FX trading system using adaptive reinforcement learning," *Expert Syst. Appl.*, vol. 30, no. 3, pp. 543–552, 2006.

[6] S. W. H. Akkerman, "Automatically trading small market capitalization cryptocurrencies using reinforcement learning," 2021.

[7] A. Ibrahim, R. Kashef, and L. Corrigan, "Predicting market movement direction for bitcoin: A comparison of time series modeling methods," *Comput. Electr. Eng.*, vol. 89, p. 106905, 2021.

[8] A. Bigiotti and A. Navarra, "Optimizing automated trading systems," in *The 2018 International Conference on Digital Science*, 2018, pp. 254–261.

[9] C. Y. Kim and K. Lee, "Risk management to cryptocurrency exchange and investors guidelines to prevent potential threats," in *2018 International Conference on Platform Technology and Service (PlatCon)*, 2018, pp. 1–6.

[10] A. Mikhaylov, N. Sokolinskaya, and E. Lopatin, "Asset allocation in equity, fixed-income and cryptocurrency on the base of individual risk sentiment," *Invest. Manag. \& Financ. Innov.*, vol. 16, no. 2, p. 171, 2019.