A Reinforcement Learning Approach to Bitcoin Trading: Proximal Policy Optimization with Trend-Following and Risk-Aware Reward Design

Victor Vladareanu[1,*]

[1]Robotics and Mechatronics Department, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

ABSTRACT

This study proposes a reinforcement learning based trading strategy for Bitcoin using Proximal Policy Optimization with a trend following and risk aware reward design. The model is developed within a custom trading environment that incorporates multiple technical indicators, including trend, momentum, and volatility features, to capture market dynamics. A continuous action space is employed to enable flexible portfolio allocation between cash and Bitcoin, allowing the agent to learn dynamic position sizing rather than discrete buy or sell decisions. The reward function is designed to encourage profit generation while penalizing excessive risk, trading activity, and drawdowns. The proposed model is evaluated on historical Bitcoin data and compared with a Buy and Hold baseline using metrics such as total return, Sharpe ratio, maximum drawdown, trading frequency, and transaction costs. The results show that while the PPO strategy does not outperform Buy and Hold in terms of total return, it achieves superior risk adjusted performance with a higher Sharpe ratio and more stable portfolio growth. However, the model exhibits high trading frequency, leading to increased transaction costs that reduce overall profitability. These findings demonstrate that reinforcement learning offers a promising approach for developing adaptive and risk sensitive trading strategies, although further improvements are required to enhance trading efficiency and cost management.

Keywords Reinforcement Learning, Proximal Policy Optimization, Bitcoin Trading, Trend Following, Risk-Aware Reward

INTRODUCTION

The rapid growth of cryptocurrency markets, particularly Bitcoin, has attracted significant attention from both researchers and practitioners due to its high volatility and potential for substantial returns [1]. Traditional trading strategies such as Buy and Hold are widely used because of their simplicity and effectiveness during prolonged bullish periods. However, these strategies lack adaptability and are unable to respond dynamically to changing market conditions, which can lead to significant losses during periods of high volatility or market downturns. As a result, there is a growing need for intelligent trading approaches that can continuously adapt to market dynamics while balancing profitability and risk.

In recent years, reinforcement learning has emerged as a promising approach for developing automated trading strategies due to its ability to model sequential decision making and learn optimal policies through

Submitted: 28 September 2025
Accepted: 10 November 2025
Published: 23 May 2026

Corresponding author
Victor Vladareanu,
victor.vladareanu@vipro.edu.ro

Additional Information and Declarations can be found on page 142

DOI: 10.47738/jdmdc.v3i2.64

How to cite this article: V. Vladareanu, "A Reinforcement Learning Approach to Bitcoin Trading: Proximal Policy Optimization with Trend-Following and Risk-Aware Reward Design," *J. Digit. Mark. Digit. Curr.*, vol. 3, no. 2, pp. 131-143, 2026.

interaction with the environment [2]. Various studies have applied deep reinforcement learning algorithms such as Deep Q Network, Advantage Actor Critic, and Proximal Policy Optimization to financial markets, including cryptocurrency trading [3]. These approaches have demonstrated the capability to generate competitive returns and adapt to complex market patterns. Furthermore, the integration of technical indicators and feature engineering techniques has been shown to improve the performance of reinforcement learning agents by providing richer representations of market states.

Despite these advancements, several challenges remain in the application of reinforcement learning to cryptocurrency trading. Many existing studies primarily focus on maximizing cumulative returns without adequately considering risk management, which often leads to unstable performance and large drawdowns [4]. In addition, reinforcement learning agents tend to exhibit overtrading behavior, resulting in excessive transaction costs that significantly reduce net profitability. Another limitation is that many models do not explicitly incorporate trend following behavior, even though trend-based strategies are widely recognized as effective in financial markets. These gaps highlight the need for a more balanced approach that integrates profitability, risk control, and trading efficiency.

To address these limitations, this study proposes a reinforcement learning based trading strategy using Proximal Policy Optimization with a trend following and risk aware reward design. The proposed model utilizes a continuous action space to enable flexible portfolio allocation between cash and Bitcoin, allowing for dynamic position sizing. A comprehensive set of technical indicators is incorporated to capture market trends, momentum, and volatility. In addition, the reward function is carefully designed to encourage profitable trading decisions while penalizing excessive risk, frequent trading, and drawdowns [5]. The performance of the proposed approach is evaluated using historical Bitcoin data and compared with a Buy and Hold baseline across multiple metrics, including return, Sharpe ratio, drawdown, and transaction costs.

The main contributions of this study can be summarized as follows. First, it introduces a reinforcement learning trading framework that integrates trend following features with a risk aware reward mechanism. Second, it provides a comprehensive evaluation of the PPO based strategy against a traditional passive strategy under realistic trading conditions. Third, it offers insights into the trade-off between return, risk, and transaction costs in reinforcement learning based trading systems.

Literature Review and Related Works

The application of reinforcement learning in financial markets has gained significant attention due to its ability to model sequential decision making and adapt to dynamic environments. Reinforcement learning techniques

have been widely explored in quantitative trading, where agents learn optimal trading policies through interaction with market data. Existing studies show that reinforcement learning is capable of outperforming traditional rule-based strategies in complex and highly volatile environments such as cryptocurrency markets [6], [7]. In particular, deep reinforcement learning methods have demonstrated strong potential in capturing non-linear patterns and temporal dependencies in financial time series data [8].

Several reinforcement learning algorithms have been applied to cryptocurrency trading, including Deep Q Network, Advantage Actor Critic, and Proximal Policy Optimization. Among these, policy-based methods such as PPO are preferred due to their stability and suitability for continuous action spaces [9], [10]. Prior works have shown that PPO can effectively learn trading strategies that generate competitive returns compared to traditional benchmarks, including Buy and Hold [11]. In addition, reinforcement learning has been successfully used for portfolio management and asset allocation, where agents dynamically adjust portfolio weights to optimize performance [12], [13].

Feature engineering plays a crucial role in improving the performance of reinforcement learning models. Many studies incorporate technical indicators such as moving averages, momentum, volatility, and oscillators to enhance the representation of market states [14], [15]. These indicators enable the agent to better capture market trends and price dynamics, which are essential for developing effective trading strategies. Furthermore, some approaches combine reinforcement learning with traditional machine learning techniques to improve predictive accuracy and decision making [16].

Despite these advancements, several limitations remain. One major challenge is that many reinforcements learning-based trading models primarily focus on maximizing returns without adequately considering risk, which can lead to unstable performance and large drawdowns [17]. In addition, excessive trading activity is a common issue, as agents may frequently rebalance their portfolios, resulting in high transaction costs that reduce net profitability [18]. Another limitation is the lack of explicit incorporation of trend following behavior, even though trend-based strategies are widely recognized as effective in financial markets [19].

Recent studies have attempted to address these issues by introducing risk aware reward functions, multi objective optimization, and improved environment design. For example, incorporating penalties for drawdown and transaction costs has been shown to improve the stability and robustness of trading strategies [20]. Additionally, some approaches utilize ensemble methods or multi agent systems to enhance performance across different market conditions. However, achieving a balance between profitability, risk management, and trading efficiency

remains an open challenge in reinforcement learning based cryptocurrency trading.

Methodology

Overview

This study proposes a reinforcement learning based trading framework for Bitcoin using Proximal Policy Optimization. The methodology consists of several main components, including data preprocessing, feature engineering, environment design, model training, and performance evaluation. The overall workflow of the proposed approach is illustrated in figure 1, which presents the sequential steps from data collection and preprocessing to model training, backtesting, and performance analysis. This framework is designed to simulate a realistic trading environment in which the agent learns to optimize portfolio allocation through continuous interaction with historical market data.

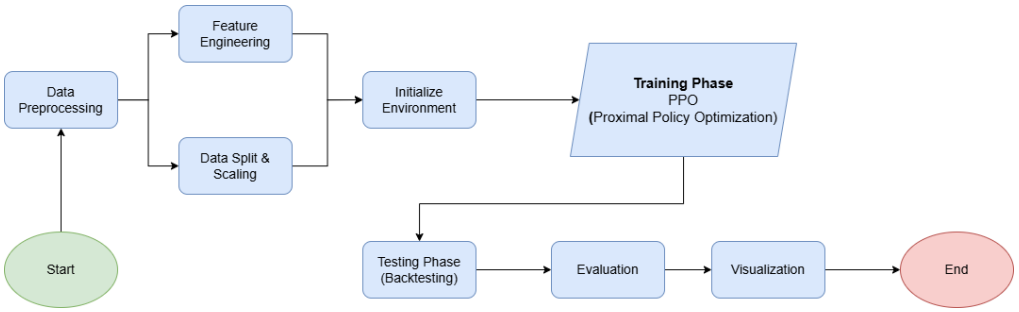

Figure 1 Research Framework

Data Collection and Preprocessing

The dataset used in this study consists of historical Bitcoin market data, including Open, High, Low, Close prices, trading volume, and market capitalization. The data is cleaned, sorted chronologically, and transformed into numerical format. Missing values and anomalies are handled using forward and backward filling. The return at each time step is computed as:

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}} \quad (1)$$

$P_t$represents the closing price at time t.

In addition, logarithmic return is calculated as:

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right) \quad (2)$$

The dataset is then split into training and testing sets using an 80:20 ratio, and feature scaling is applied using standardization.

Feature Engineering

A set of technical indicators is constructed to represent market conditions, including trend indicators, momentum, volatility, and oscillators. These features include moving averages, exponential moving averages, MACD, RSI, Bollinger Band width, and volume-based indicators. The engineered features provide a comprehensive representation of price dynamics and market behavior.

Reinforcement Learning Environment

A custom trading environment is implemented using Gymnasium to simulate trading interactions. The agent operates in a long only setting, where capital is allocated between cash and Bitcoin. The state consists of a rolling window of historical features with window size W. The observation vector is defined as:

$$s_t = [x_{t-W+1}, x_{t-W+2}, \dots, x_t, p_t, c_t, d_t] \tag{3}$$

x_trepresents feature vectors, p_tis the current position, c_tis the cash ratio, and d_tis the current drawdown. The action is defined as a continuous value:

$$a_t \in [0,1] \tag{4}$$

$a_t = 0$represents full cash and $a_t = 1$represents full allocation to Bitcoin.

To reduce excessive fluctuations, the action is smoothed as:

$$\tilde{a}_t = \alpha a_{t-1} + (1-\alpha)a_t \tag{5}$$

$\alpha$is a smoothing factor.

The reward function is designed to balance profitability and risk. The primary component is the portfolio return:

$$R_t^{portfolio} = \frac{V_{t+1} - V_t}{V_t} \tag{6}$$

The overall reward is defined as:

$$\text{Reward}_t = 100 \cdot R_t^{portfolio} + \beta_1 \cdot R_t^{market} - \beta_2 \cdot \Delta a_t - \beta_3 \cdot DD_t \tag{7}$$

$R_t^{market}$is the market return

$\Delta a_t$is the change in position

$DD_t$is the drawdown

$\beta_1, \beta_2, \beta_3$are weighting parameters

This formulation encourages the agent to follow market trends, minimize excessive trading, and reduce exposure to large losses.

Model Training

The trading agent is trained using the PPO algorithm with a multilayer perceptron policy network. The objective of PPO is to maximize the expected cumulative reward:

$$J(\theta) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R_t\right] \quad (8)$$

$\gamma$is the discount factor.

Training is performed over multiple episodes with randomized starting points to improve generalization.

Backtesting and Evaluation

After training, the model is evaluated on unseen test data using a full period backtesting approach to simulate real trading conditions without further learning or parameter updates. During this phase, the trained agent generates trading decisions sequentially at each time step based on the observed state, and the corresponding portfolio value is continuously updated and recorded over time. The performance of the strategy is assessed using several key evaluation metrics to provide a comprehensive analysis. Total return is used to measure overall profitability, while the Sharpe ratio evaluates risk adjusted performance by considering return relative to volatility. Maximum drawdown is included to quantify the largest peak to trough loss and assess downside risk exposure. In addition, transaction costs are calculated to account for realistic trading frictions resulting from frequent rebalancing, and trading frequency is measured to capture the level of market activity and strategy aggressiveness. Together, these metrics provide a balanced evaluation of both return generation and risk management capabilities of the proposed trading model.

Visualization

The performance of the model is further analyzed using equity curves and trading action plots to provide a visual interpretation of the trading behavior and portfolio evolution over time. The equity curve illustrates the growth of the portfolio value throughout the evaluation period, allowing for the identification of trends, volatility patterns, and drawdown periods. Meanwhile, the trading action plot presents the timing and magnitude of position adjustments made by the agent, highlighting how the model responds to different market conditions. Together, these visualizations offer valuable insights into the dynamic decision-making process of the agent and help to better understand the relationship between market movements and the resulting trading actions.

Algorithm 1: PPO-Based Bitcoin Trading Strategy

Input: Dataset D, initial balance B_0, window size W, transaction fee τ, episode length T
Output: Trained policy π_θ, portfolio value V_t, actions a_t
Process:

Start

Initialize $cash = B_0, units = 0, position = 0$

For each episode $e = 1 \dots E$:

Select starting index t_0

Initialize state $s_t$using windowed features

For each step $t = t_0 \dots t_0 + T$:

Sample action

$$a_t \sim \pi_\theta(a_t \mid s_t)$$

Smooth action

$$\tilde{a}_t = \alpha \cdot position + (1 - \alpha) \cdot a_t$$

Compute portfolio value

$$V_t = cash + units \cdot C_t$$

Compute target allocation

$$A_t = \tilde{a}_t \cdot V_t$$

Compute adjustment

$$\Delta A_t = A_t - units \cdot C_t$$

Compute transaction fee

$$fee = \tau \cdot \mid \Delta A_t \mid$$

Update portfolio

$$cash = cash - \Delta A_t - fee$$

$$units = \frac{A_t}{C_t}$$

Move to next step and compute new value

$$V_{t+1} = cash + units \cdot C_{t+1}$$

Compute returns

$$R^{portfolio} = \frac{V_{t+1} - V_t}{V_t}$$

$$R^{market} = \frac{C_{t+1} - C_t}{C_t}$$

Compute drawdown

$$DD = \frac{Peak - V_{t+1}}{Peak}$$

Compute reward

$$r_t = 100R^{portfolio} + \beta_1 R^{market} - \beta_2 \mid \Delta a \mid - \beta_3 DD$$

Update policy parameters

$$\theta \leftarrow \theta + \nabla_\theta J^{PPO}(\theta)$$

Update state $s_t \rightarrow s_{t+1}$

Return π_θ, V_t, a_t

End

Results

The performance of the proposed Proximal Policy Optimization (PPO) based trading strategy is evaluated using an out of sample test dataset and compared against a Buy and Hold baseline to assess its effectiveness in real market conditions. The evaluation framework incorporates multiple performance metrics to provide a comprehensive analysis, including total return to measure overall profitability, Sharpe ratio to assess risk adjusted performance, maximum drawdown to

capture downside risk exposure, trading frequency to reflect the level of market activity and decision-making intensity, and transaction costs to account for the impact of realistic trading frictions. This combination of metrics enables a balanced assessment of both return generation and risk management capabilities, allowing for a detailed comparison between an active reinforcement learning driven strategy and a passive investment approach.

Quantitative Performance

The numerical results of both strategies are summarized in table 1. Starting from an initial capital of 10,000, the PPO based strategy increases the portfolio value to 42,973, achieving a total return of 329.74%. This result demonstrates the ability of the reinforcement learning model to generate substantial profits through dynamic allocation decisions over the evaluation period. In comparison, the Buy and Hold strategy achieves a higher final portfolio value of 49,334, corresponding to a total return of 393.84%. The superior performance of Buy and Hold in terms of absolute return indicates that maintaining full exposure to Bitcoin is advantageous during prolonged upward market conditions.

In terms of risk adjusted performance, the PPO strategy achieves a Sharpe ratio of 1.49, which is higher than the Buy and Hold strategy with a Sharpe ratio of 1.42. This suggests that the PPO model is more efficient in generating returns relative to the level of risk taken. Despite this difference, both strategies exhibit similar maximum drawdowns of approximately 51.8%, indicating that they are exposed to comparable levels of downside risk during adverse market movements. These results highlight that while the PPO strategy does not outperform Buy and Hold in total return, it provides improved performance when considering the balance between return and risk.

Table 1 Performance Comparison of PPO and Buy-and-Hold Strategies

Strategy	Final Value	Return (%)	Sharpe Ratio	Max Drawdown (%)	Rebalances	Fees Paid
PPO Long-Only	42,973	329.74%	1.49	-51.85%	567	232.10
Buy & hold	49,334	393.84%	1.42	-51.86%	2	59.29

Equity Curve Characteristics

The equity curve comparison between the PPO based strategy and the Buy and Hold strategy is shown in figure 2. Both strategies exhibit significant portfolio growth throughout the evaluation period, indicating that they are able to capture the overall upward movement of the Bitcoin market. However, clear differences can be observed in the pattern of growth. The Buy and Hold strategy demonstrate a steeper and more continuous increase in portfolio value, particularly during extended bullish phases where the market experiences strong upward momentum.

In contrast, the PPO strategy shows a more gradual and stepwise growth trajectory. Periods of slower growth and temporary plateaus are visible, reflecting the agent's dynamic adjustment of market exposure over time. The PPO model does not consistently maintain full investment in the asset, which results in missed opportunities during sharp price increases but also indicates more controlled participation in the market. This behavior suggests that the model adapts its allocation based on changing market conditions, leading to a smoother progression of portfolio value compared to the more aggressive growth observed in the Buy and Hold strategy.

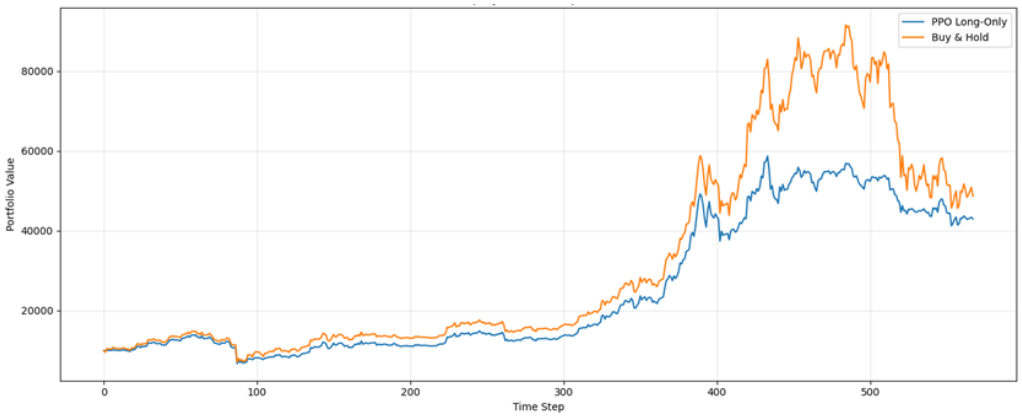

Figure 2 Equity Curve Comparison between PPO and Buy-and-Hold Strategies

Risk Metrics

Risk is evaluated using two key metrics, namely the Sharpe ratio and maximum drawdown, to capture both return efficiency and downside exposure. The PPO strategy achieves a Sharpe ratio of 1.49, which is higher than the Buy and Hold strategy at 1.42, indicating that the PPO model generates higher returns relative to the level of risk taken. This suggests that the reinforcement learning agent is more effective in balancing profit generation with volatility control over the evaluation period. In terms of downside risk, both strategies exhibit nearly identical maximum drawdowns of approximately 51.8%, indicating that they are exposed to similar levels of peak to trough losses during adverse market conditions. Despite achieving comparable drawdown levels, the higher Sharpe ratio of the PPO strategy highlights its ability to deliver more consistent returns under similar risk exposure.

Trading Activity and Transaction Costs

A notable difference between the two strategies lies in trading frequency, which reflects how actively each approach interacts with the market. The PPO agent performs a total of 567 rebalancing actions during the test period, indicating a highly active strategy that continuously adjusts its portfolio allocation in response to changing market conditions. In contrast, the Buy and Hold strategy executes only two transactions, consisting of an initial purchase and a final liquidation, thereby maintaining a constant

exposure to the asset throughout the entire period. This substantial difference in trading activity leads to significantly higher transaction costs for the PPO strategy, amounting to 232.10 compared to only 59.29 for Buy and Hold. The increased costs associated with frequent trading reduce the net profitability of the PPO model, highlighting the impact of transaction fees as an important factor in evaluating the effectiveness of reinforcement learning based trading strategies.

Portfolio Allocation Behavior

The trading actions executed by the PPO agent are illustrated in figure 3, providing a detailed view of how the model adjusts its position over time in response to market dynamics. The PPO agent continuously modifies its allocation between cash and Bitcoin, reflecting its learned policy based on observed market conditions and input features. Unlike discrete trading strategies that rely on fixed buy or sell signals, the PPO model operates in a continuous action space, allowing it to fine tune its position at each time step. This enables the agent to gradually increase or decrease exposure rather than making abrupt transitions, resulting in smoother changes in portfolio composition.

Throughout the trading period, the PPO agent exhibits varying levels of market participation, shifting between higher and lower exposure depending on perceived trends and risk conditions. During periods of upward price movement, the agent tends to increase its allocation to Bitcoin, while in uncertain or less favorable conditions it reduces exposure by allocating more capital to cash. This dynamic behavior contrasts with the Buy and Hold strategy, which maintains a constant full allocation regardless of market changes. The ability of the PPO agent to adjust its exposure continuously leads to a more flexible trading pattern, reflecting adaptive decision making in response to evolving market environments.

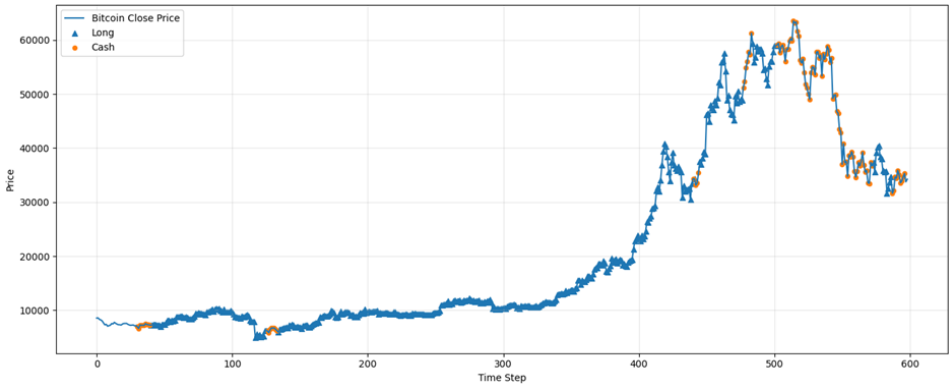

Figure 3 Trading Actions of PPO Agent over Time

Discussion

The results highlight a clear trade-off between absolute return and risk adjusted performance when comparing the PPO based strategy with the Buy and Hold approach. Although Buy and Hold achieves higher total

return and final portfolio value, this outcome is strongly influenced by prolonged bullish conditions in the Bitcoin market, where constant full exposure leads to maximum capital appreciation. In contrast, the PPO strategy demonstrates a more balanced performance by achieving a higher Sharpe ratio, indicating better efficiency in generating returns relative to risk. This suggests that the reinforcement learning agent is capable of learning a policy that prioritizes stability and controlled growth rather than purely maximizing profit. The similar levels of maximum drawdown observed in both strategies further indicate that while the PPO model improves return consistency, it does not significantly reduce exposure to extreme market downturns.

Another important observation is the impact of trading frequency on overall performance. The PPO agent exhibits highly active trading behavior, as reflected by the large number of rebalancing actions, which leads to substantially higher transaction costs. These costs reduce net returns and partially explain why the PPO strategy underperforms compared to Buy and Hold in terms of total profit. However, the dynamic allocation behavior of the PPO model, including its ability to adjust exposure based on market conditions, reflects a more realistic and adaptive trading approach. This adaptability may provide advantages in different market regimes, particularly in sideways or bearish conditions where passive strategies are less effective. Overall, the findings indicate that while reinforcement learning based trading does not necessarily outperform passive strategies in all scenarios, it offers significant benefits in terms of flexibility, risk management, and responsiveness to changing market environments.

Conclusion

This study presents a reinforcement learning based trading strategy for Bitcoin using Proximal Policy Optimization with a trend following and risk aware reward design. The experimental results demonstrate that while the proposed PPO strategy does not outperform the Buy and Hold approach in terms of total return, it achieves superior risk adjusted performance as indicated by a higher Sharpe ratio and more consistent portfolio growth. The model is able to dynamically adjust its market exposure, reflecting adaptive decision making in response to changing market conditions. However, the high trading frequency leads to increased transaction costs, which negatively impacts overall profitability. These findings suggest that reinforcement learning offers a promising framework for developing adaptive and risk sensitive trading strategies, although further improvements in transaction cost handling and trading efficiency are necessary to enhance its competitiveness against passive investment approaches.

Declarations

Author Contributions

Conceptualization: V.V.; Methodology: V.V.; Software: V.V.; Validation: V.V.; Formal Analysis: V.V.; Investigation: V.V.; Resources: V.V.; Data Curation: V.V.; Writing Original Draft Preparation: V.V.; Writing Review and Editing: V.V.; Visualization: V.V.; All authors have read and agreed to the published version of the manuscript.

Data Availability Statement

The data presented in this study are available on request from the corresponding author.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] R. P. N. Rao, “Reinforcement Learning: An Introduction; R. S. Sutton, A. G. Barto (Eds.),” *Neural Networks*, vol. 13, no. 1, pp. 133–135, Jan. 2000, doi: 10.1016/S0893-6080(99)00098-2.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, et al., “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015, doi: 10.1038/nature14236.

[4] Z. Jiang and J. Liang, “Cryptocurrency Portfolio Management with Deep Reinforcement Learning,” in *Proc. IntelliSys 2017*, London, UK, Sep. 2017, doi: 10.1109/IntelliSys.2017.8324237.

[5] X.-Y. Liu, H. Yang, J. Gao, and C. D. Wang, “FinRL: Deep Reinforcement Learning Framework to Automate Trading in Quantitative Finance,” *arXiv preprint arXiv:2111.09395*, Nov. 2021, doi: 10.48550/arXiv.2111.09395.

[6] T. E. Koker and D. Koutmos, “Cryptocurrency Trading Using Machine Learning,” *Journal of Risk and Financial Management*, vol. 13, no. 8, p. 178, Aug. 2020, doi: 10.3390/jrfm13080178.

[7] H. Yang and A. Malik, “Reinforcement Learning Pair Trading: A Dynamic Scaling Approach,” *Journal of Risk and Financial Management*, vol. 17, no. 12, p. 555, 2024, doi: 10.3390/jrfm17120555.

[8] S. Sun, R. Wang, and B. An, “Reinforcement Learning for Quantitative Trading,” *ACM Transactions on Intelligent Systems and Technology*, vol. 14, no. 3, p. 1-29, Jan. 2023, doi: 10.1145/3582560.

[9] J. Schulman, F. Wolski, P. Dhariwal, and A. Radford, “Proximal Policy Optimization Algorithms,” *arXiv preprint arXiv:1707.06347*, Jul. 2017, doi: 10.48550/arXiv.1707.06347.

[10] V. Kochliaridis, E. Kouloumpris, and I. Vlahavas, “Combining deep reinforcement learning with technical analysis and trend monitoring on cryptocurrency markets,” *Neural Computing and Applications*, vol. 35, no. 29, pp. 1–18, Apr. 2023, doi: 10.1007/s00521-023-08516-x.

[11] F. Liu, Y. Li, B. Li, and J. Li, “Bitcoin transaction strategy construction based on deep reinforcement learning,” *Applied Soft Computing*, vol. 113, no. December, p. 107952, Dec. 2021, doi: 10.1016/j.asoc.2021.107952.

[12] Z. Xu, “Dynamic Portfolio Optimization Using Reinforcement Learning in Cryptocurrency Markets,” *Academic Journal of Business & Management*, vol. 7, no. 4, 2025, doi: 10.25236/AJBM.2025.070428.

[13] D. S. Barra, H. F. Almeida, R. J. Feltrin, and S. R. Marin, “Reinforcement Learning Applied to a Cryptocurrency Portfolio in a Complexity Environment,” *Revista de Economia e Estatística*, vol. 36, no. 1, Dec. 2020, doi: 10.14393/REE-v36n1a2021-50850.

[14] A. A. Aigner and W. Schrabmair, “Power Assisted Trend Following,” Feb. 2020, doi: 10.13140/RG.2.2.20898.17605/1.

[15] N. Jarunde, “Trading Financial Markets—Fundamental vs Technical Analysis: Pros and Cons for Different Investment Styles,” Sep. 2023, doi: 10.13140/RG.2.2.22014.80964.

[16] R. Huang, “Research on Optimization of Cryptocurrency Trading Strategies Based on Reinforcement Learning—Combining Traditional Machine Learning and Deep Reinforcement Learning Methods,” *ITM Web of Conferences*, vol. 78, no. September, p. 13, Sep. 2025, doi: 10.1051/itmconf/20257801001.

[17] K. Kumlungmak and P. Vateekul, “Multi-Agent Deep Reinforcement Learning With Progressive Negative Reward for Cryptocurrency Trading,” *IEEE Access*, vol. PP, no. 99, pp. 1–1, Jan. 2023, doi: 10.1109/ACCESS.2023.3289844.

[18] S. Wang and D. Klabjan, “An Ensemble Method of Deep Reinforcement Learning for Automated Cryptocurrency Trading,” in *Proc. 2024 IEEE Int. Conf. Blockchain and Cryptocurrency (ICBC)*, May 2024, doi: 10.1109/ICBC59979.2024.10634436.

[19] C. S. J. Huang and Y.-S. Su, “Trading Strategy of the Cryptocurrency Market Based on Deep Q-Learning Agents,” *Applied Artificial Intelligence*, vol. 38, no. 1, Jul. 2024, doi: 10.1080/08839514.2024.2381165.

[20] J. H. Chun and S. J. Lee, “Cryptocurrency Futures Portfolio Trading System Using Reinforcement Learning,” *Applied Sciences*, vol. 15, no. 17, p. 9400, Aug. 2025, doi: 10.3390/app15179400.