Enhancing A Stock Timing Strategy by Reinforcement Learning

Yaoming Li, Yun Chen

Abstract—**The generation of the stock timing strategy is a crucial task for stock trading. Vast previous studies focus on proposing an end-to-end stock timing strategy based on reinforcement learning. However, it is difficult to explain why the model works, and it takes courage to believe in a black box in real stock trading. In this paper, we propose PPO Enhancement Strategy to modify the trading signal of the base stock trading strategy instead of predicting the direction of stock price directly. The base strategy can be formed by technical analysis, fundamental analysis, and other interpretable models, so that it can increase the interpretability of the trading model. In order to make the result of the PPO Enhancement Strategy robust, we perform extensive experiments on two market Indices and four stocks from American stock markets. The proposed PPO Enhancement Strategy outperforms the benchmarks, the Buy-and- Hold Strategy and the Moving Average Strategy, in terms of different evaluation criteria.**

Index Terms—**reinforcement learning, enhancing stock trading strategy, moving average strategy, proximal policy optimization, stock trading.**

I. Introduction

STOCK investment is always an hot topic in finance research. And stock timing is an important part of that. However, because of the uncertainty of the moving direction of stock price, it is difficult to make a decision when to buy or sell stock in the market.

Vast researches([1], [2], [3], [4], [5], [6], [7], [8], [9]) have studied stock timing strategies. That mainly includes three categories: technical analysis [10], [11], fundamental analysis [12], [13], and trading based on statistics and machine learning [14], [15]. Among that, technical analysis tries to predict stock moving direction based on technical indicators such as momentum indicator(Average Directional Movement Index; Absolute Price Oscillator; Momentum; Percentage Price Oscillator), volume Indicators(Chaikin A/D Line; On Balance Volume), and pattern recognition(Two Crows; Three Inside Up/Down; Three Advancing White Soldiers; Abandoned Baby) from the price and trading volume. Fundamental analysis decides to buy or sell stocks by analyzing companies' value through fundamental indicators such as Earnings per Share (EPS), Price to Earnings Ratio (P/E), Free Cash Flow (FCF), and so on. The machine learning methods try to apply logistic regression, support vector machine, deep neural networks, and so on into timing strategies.

In recent years, with the development of artificial intelligence(AI), especially, after AlphaGo([16]) defeated humans on Go, more and more researchers try to apply deep learning and reinforcement learning into stock portfolio management([17], [18], [19], [20]) and stock trading. [18] proposes a reinforcement learning framework for cryptocurrency portfolio management and finds that the portfolio based on reinforcement learning gets a ten times return. [19] proposes a portfolio management model(system-DeepBreath) based on reinforcement learning, and claims that the return of investment achieved by DeepBreath outperforms current expert investment strategies while minimizing the market risk. [20] proposes a portfolio management model based on reinforcement learning in a market with a dynamic number of assets and finds that the proposed method outperforms state-of-the-art methods, achieving average daily returns of over 24% on the cryptocurrency markets. [21] uses data augmentation models, based on deep Q-learning, proximal policy optimization and soft actor-critic, to guide stock timing, and finds that proximal policy optimization (PPO) is the best algorithm to achieve a better performance based on different measures. [22] proposes a stock timing strategy using reinforcement learning with a Gated Recurrent Unit as stock states extraction and finds that the trading strategy outperforms turtle strategies and other deep reinforcement learning trading strategies. [6] uses another two innovative states extraction methods, K-Means and candlestick, to extract stock states from the historical stock data, and trains the agent based on the Q-learning algorithm. The authors claim that the proposed two timing strategies based on Q-learning outperform the Buy-and-Hold strategy and the timing strategy based on a decision tree.

Most of the works focus on how to propose an end-to-end trading model to defeat the market. However, deep learning and reinforcement learning is a black box so that in real trading it takes courage to trust completely the trading model based on reinforcement learning. To fill this gap, this paper proposes a reinforcement learning algorithm to enhance the traditional stock trading strategy instead of predicting the direction of stocks directly. In this way, the base trading strategy can be a white box and is easy to understand. Reinforcement learning just modifies the trading signals from the base trading strategy. Thus this type of trading strategy can be used more confidently, compared with a black-box trading strategy based on deep learning or reinforcement learning. In our model, the agents are trained by a reinforcement learning algorithm and decide whether to follow or modify the base trading signals. The base trading strategy can be formed by technical analysis such as average strategy or machine learning such as logistic regression, that is, it can be a white box and interpretable. Our main contributions are as follows: 1) We propose a reinforcement learning framework to improve the performance of the base trading strategy. 2) In order to enhance the base stock trading strategy, we define the

Manuscript received March 3, 2021; revised August 13, 2021.

Yaoming Li is a PhD candidate of School of Information Management and Engineering, Shanghai Key Laboratory of Financial Information Technology, Shanghai University of Finance and Economics, Shanghai 200433, China (corresponding author, e-mail: lym@163.sufe.edu.cn)

Yun Chen is a professor of Shanghai Key Laboratory of Financial Information Technology, Shanghai University of Finance and Economics, Shanghai 200433, China (e-mail: chenyun@sufe.edu.cn)

actions of the reinforcement learning agent as ”following the base strategy” and ”reversing the base strategy” rather than ”the longing stock signal” and ”shorting stock signal” like a general end-to-end trading strategy. 3) We apply the state augment methods into reinforcement learning. The states of reinforcement learning combines the base stock timing strategy's trading signals with the stock's financial features. 4) We perform extensive experiments to demonstrate reinforcement learning can enhance the performance of the base stock trading strategy. The results show the PPO Enhancement Strategy can get decent performance compared with the base stock trading strategy(Moving Average Strategy) and the Buy-and-Hold Strategy on different stocks and indices in terms of different evaluation criteria.

The rest of this paper are organized as follows: Section II describes the preliminaries and reviews the background on reinforcement learning, proximal policy optimization algorithm, as well as the base stock timing strategy(Moving Average Strategy). Section III explicates the details of applying reinforcement learning into enhancing the base stock timing strategy. Section IV discusses the details of experiments and their results. Section V concludes the paper and points out some future research directions.

II. The Background

In this section, we introduce the approaches used in this paper, including reinforcement learning, proximal policy optimization, and the base stock timing strategy.

A. Reinforcement learning

The problems can be solved by reinforcement learning, can be modeled as a Markov Decision Probleme(MDP). There are fives essential elements (S, A, r, P, γ) in a MDP, where S is the Markov state set, A is the set of actions, $r(s, a) = \mathbb{E}(r_{t+1}|s_t = s, a_t = a)$ is a reward function used to calculate the reward of executing action a at state s, $P(s_{t+1}|s_t, a_t)$ is the transition probability from s_t to s_{t+1} by action a_t and $\gamma \in (0, 1)$ is a discount factor. Reinforcement learning includes an agent and an environment, and the interaction between them is shown in Fig 1. The environment outputs state s_t at time t, then the agent selects a action a_t based on s_t, and executes the action in the environment. At last, the environment gives the agent a feedback r_{t+1} as a reward and the next state s_{t+1}. Keep looping this process, we can get a trajectory $\tau = (s_1, a_1, r_1, s_2, ..., a_T, r_T, s_{T+1})$. After collecting these trajectories, we can train the agent by reinforcement learning algorithm to improve its performance. The goal of the agent is to maximize the expectation of total accumulated rewards is defined as,

$$Reword_t = \sum_{i=0}^{T-t-1} \gamma^i r_{t+i+1} \tag{1}$$

The agent, also called a policy, $\pi_\theta(a|s) : S \times A \rightarrow [0, 1]$ is a conditional probability distribution of action a under the condition state s, where θ is parameters of the distribution. It observes states from the environment and decides actions by the stochastic policy $\pi_\theta(a|s)$. The environment in reinforcement learning receive the action from the agent and feedbacks a reward to evaluate the action and the next state to the agent. The states space consists of financial features of stocks such as open price, high price, low price, close price, volume, $MA10$, and $MA20$.

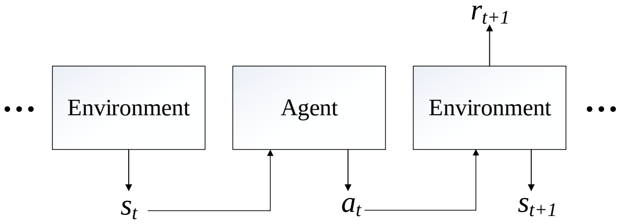


Fig. 1. Interaction between the agent and the environment.

B. Proximal policy optimization

Proximal Policy Optimization(PPO) [23], [24] is a model-free off-policy reinforcement learning algorithm, in which the agent aims to obtain the optimal policy to maximize the objective function by policy gradient algorithm. PPO is based on the similar ideas of Trust Region Policy Optimization(TRPO) [23] and becomes a popular reinforcement learning methods due to its state-of-the-art performance and efficiency in different scenarios, compared with the other online methods such as policy gradient algorithm [25], [26] and Actor Critic Algorithm(A2C) [27]. In order to make the samples sampled from the old policy can be used to train the agent, PPO uses a few other tricks to keep new policies close to old by optimizing a surrogate objective function L, as follow

$$L(\theta) = \mathbb{E}_t\big[\min(b_t(\theta)A_t, clip(b_t(\theta), 1 - \varepsilon, 1 + \varepsilon)A_t)\big] \tag{2}$$

where $b_t(\theta)$ denotes the probability ratio $b(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$, $\pi_\theta(a_t|s_t)$ is the policy, θ is the parameters of the policy, ε is the clip parameter to control the difference between the new policy and the old policy and A_t is the generalized advantage estimation function [28] given by

$$A_t = \sum_{l=1}^{\infty} (\lambda\gamma)^l (r_t + \gamma V(s_{t+l+1})) - V(s_l) \tag{3}$$

where γ is the discount factor, λ is a factor for trade-off of bias and variance for Generalized Advantage Estimator, and $V(s_t)$, the value function at s_t, is the expectation of return at the state s_t under the policy π_θ. Then the policy is updated via

$$\theta_{k+1} = argmin_\theta - L(\theta) \tag{4}$$

From (2, 4), when $b(\theta)$ is in $(1 - \varepsilon, 1 + \varepsilon)$, we update the policy by the samples sampled from the old policy, that is we do not need generate new samples from the new policy. Therefore, PPO not only improves the efficiency of sample utilization but also the stability of the strategy.

C. The base stocks timing strategy

We consider the moving average strategy([10], [29], [30]) as the base strategy which is often used in practice. It is very popular, can be tailored according to any time frame, and is suitable for long-term and short-term investors. The moving average (MA) is a simple technical analysis tool that smooths out stocks' prices by filtering out the noise from random short-term price fluctuations. Investors can choose a specific period such 10 days, 20 days or any period the stock investors to calculate the average. A moving average has different computing methods, simple moving average(SMA) and the exponential moving average(EMA). For a 10 days' SMA, it is the arithmetic mean of the last 10 days' stock closing price. However, EMA's calculation is more complex than SMA, as it put more weights on the recent prices. For simplicity, this paper constructs a moving average timing strategy by SMAs, which is one of the main moving average strategies using two moving averages: one is long-term average and the other short-term average. The long-term average is 20 days moving average($MA20$), and the short-term averate is 10 days($MA10$). When the $MA10$ crosses above the $MA20$, it comes to a so called "golden cross" point. This is a buy signal as it indicates that the trend is shifting up. On the other hand, when the $MA10$ crosses below the $MA20$, it is a sell signal called "dead/death cross" point, as it indicates that the trend is shifting down. We then can define the moving average timing strategy as follows

$$\text{policy} = \begin{cases} 1 & \text{if MA10} > \text{MA20} \\ -1 & \text{if MA10} < \text{MA20} \end{cases}. \tag{5}$$

The policy is that 1 means long signal, while -1 means short signal.

III. Enhance Stock Timing Strategy Using Reinforcement Learning

This section will introduce reinforcement learning to enhance the performance of the base stock timing strategy. For this purpose, firstly, we define the essential ingredients(states space, actions space, rewards function) in reinforcement learning. It is popular in practical trading strategy to integrate stock features with asset prices for a better prediction. In our framework, we consider stock features including opening price($Openprice_t$), high price($Highprice_t$), low price($Lowprice_t$), the volume($Volume_t$), short-term moving average($MA10_t$), and long-term moving average($MA20_t$) that are often used in timing strategies. In the raw data, because the value of the features changes too much, we firstly use the Z-score to standardize the features, as follows

$$O_t = \frac{Openprice_t - mean(Openprice)}{std(Openprice)} \tag{6}$$

$$H_t = \frac{Highprice_t - mean(Highprice)}{std(Highprice)} \tag{7}$$

$$L_t = \frac{Lowprice_t - mean(Lowprice)}{std(Lowprice)} \tag{8}$$

$$V_t = \frac{Volume_t - mean(Volume)}{std(Volume)} \tag{9}$$

$$MA10_t = \frac{MA10_t - mean(MA10)}{std(MA10)} \tag{10}$$

$$MA20_t = \frac{MA20_t - mean(MA20)}{std(MA20)} \tag{11}$$

Where $mean(\cdot)$ is a function to get the mean of the features, and $std(\cdot)$ is a function to get the standard deviation of the features so that the stock features at time t in the environment can be represented as

$$S_t^* = [O_t, H_t, L_t, V_t, MA10_t, MA20_t]. \tag{12}$$

Besides the stock features above, we also augment the stock features by combining the trading signal generated from the Moving Average Strategy. The augmented state is defined as

$$s_t = (S_t^*, P_t)$$

where S_t^* is observable states. Stock features as above(open price, high price, low price, volume, MA10, and MA20)are defined as in (6-11) and P_t is the trading signal and it is defined as in (5). The dimension of states space is seven. In this paper, P_t is generated by Moving Average Strategy, but it can be more generally generated by other strategies such as machine learning classifier, and so on.

Secondly we need to define the action space of the reinforcement learning agent. The action space has 2 dimensions: action 0 means following the base timing strategy, on the other hand, action 1 means making the opposite decision compared with the base timing strategy. It is defined, as follows

$$\text{action} = \begin{cases} 0 & \text{follow base timing strategy} \\ 1 & \text{'long' when the base timing} \\ & \text{strategy outputs 'short' singal.} \\ & \text{ 'short' when the base timing} \\ & \text{strategy outputs 'long' singal.} \end{cases} \tag{13}$$

The agent is a neural network with a structure like Fig 2. It has four fully connected hidden layers. The activation function is RELU. The input layer is the current state s_t. There are two units in the output layer, one representing the action to determine whether to follow the base timing strategy and the other representing the value of the action to evaluate the pros and cons of the action.

The reward function is the logarithm of the total return every day when the agent's action is 0. It is defined, as follows

$$r_t = \begin{cases} ln(C_{t+1}/C_t) & \text{if policy} = 1 \\ ln(2 - C_{t+1}/C_t) & \text{if policy} = -1 \end{cases} \tag{14}$$

and when the action is 1, that is, the base timing strategy is reversed, the reward function is as follows

$$r_t = \begin{cases} ln(2 - C_{t+1}/C_t) & \text{if policy} = 1 \\ ln(C_{t+1}/C_t) & \text{if policy} = -1 \end{cases} \tag{15}$$

where C_t is the closing price of stocks at time t. Fig.3 shows the training framework for this agent. Firstly, we use the base timing strategy such as moving average strategy, machine learning timing strategy to obtain the trading signals, P_t. Secondly, we combine the trading signal P_t as a feature with the stock finance features(opening price, high price, low price, the volume, MA10, and MA20) representing the states of the reinforcement learning agent. Thirdly, we construct a neural policy network as an agent, as shown in the Fig 2. Lastly, we use the proximal policy optimization algorithm to train agents. In detail, the yellow blocks represent the features $MA10$ and $MA20$ used in the base strategy. The blue blocks represent other trading features (opening price, high price, low price, the volume). The green blocks represent

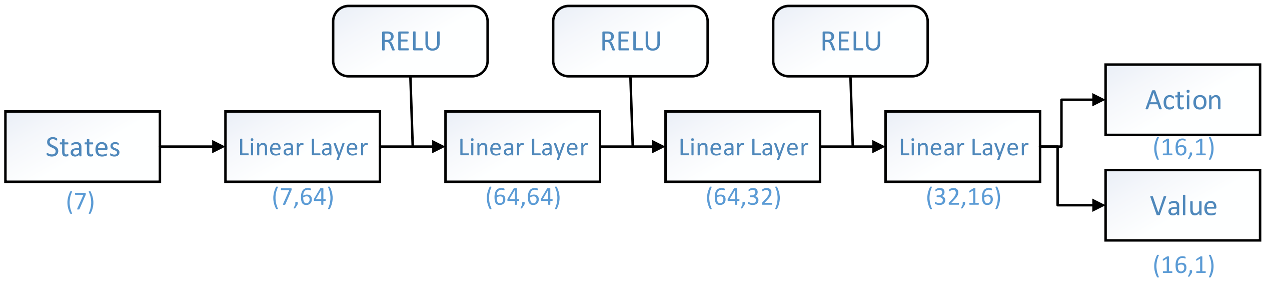


Fig. 2. The Structure of Agent

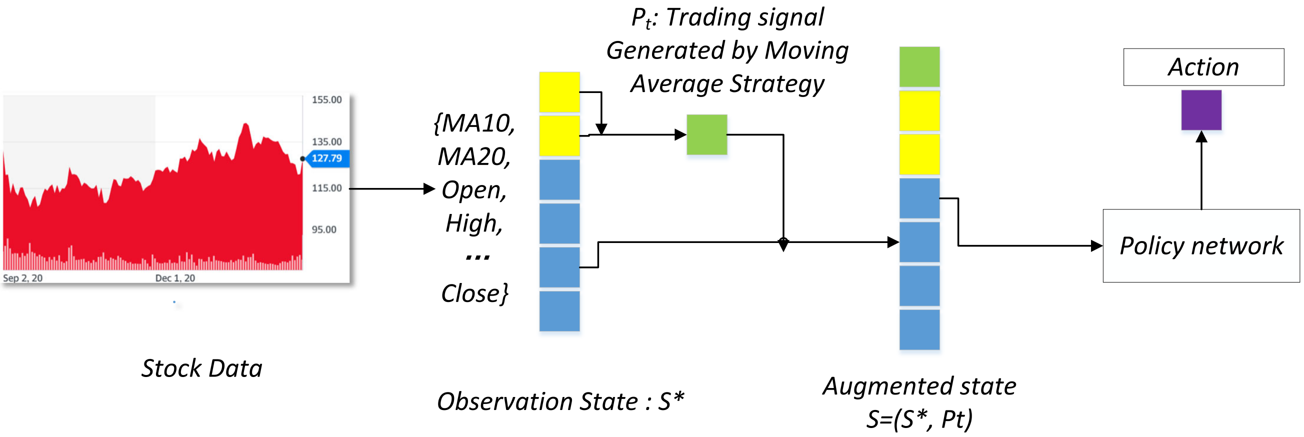

155.00
135.00
127.79
115.00
95.00
Sep 2, 20
Dec 1, 20
Stock Data
{MA10, MA20, Open, High, ... Close}
P_t: Trading signal Generated by Moving Average Strategy
Observation State : S*
Augmented state S=(S*, Pt)
Action
Policy network

Fig. 3. The Train Framework

the trading signal formed by the moving average strategy, while the purple blocks represent the action formed by the reinforcement learning algorithm.

IV. Experiments and Discussion

In this section, we provide the details of experiments conducted with the proposed method, including data sets, performance evaluation metrics, and experimental results. In experiments, the base timing strategy is a moving average strategy(Crossovers). The moving average strategy is a trading strategy often used in practice, which is composed of short-term (10-day) moving average (MA 10) and long-term (20-day) moving average (MA 20). When MA 10 crosses MA 20 upwards, investor buy long stocks, while when MA 10 crosses MA 20 upwards, investors buy short stocks.

A. *Experimental dataset*

We conducted experiments on the daily data of two stock indices and four stocks, shown in Table I. Firstly, we use reinforcement learning algorithms to train the agent in the training set, so as to enhance the base timing strategy. Then, we measure the performance of the agent on different stocks during the testing period. In order to make the experiments more robust, we chose two stocks with upward trend(Apple, MSFT) and two stocks with downward trends(COTY, ICON) . Fig 4 plots of their daily closing price during the whole trading periods with the X axis presenting the trading days and the Y axis presenting the price of different stocks.

All the stock indices and individual stocks come from the American Stock Exchanges. The experimental dataset is obtained from Yahoo finance through Python API.

B. *The parameter settings*

Regarding the proximal policy optimization settings, the PPO Enhancement Strategy consists of a reinforcement learning decision-making module. In the RL decision-making module, a policy is a fully connected network. We set the network with four fully connected hidden layers, and the activation function is the RELU function. The input layer is the current states s_t, and there are two units in the output layer: one is the action to decide whether to follow the basic stock timing strategy, and the other is the value function of the current states. The hidden layers are linear, and the first two hidden layers both have 64 units, the third hidden and fourth have 32 units and 16 units respectively. The details of the parameters are shown in the Table IV in the Appendix.

C. *Performance evaluation metrics*

To evaluate the proposed stock timing strategy based on reinforcement learning, two benchmarks(the Buy-and-Hold trading strategy(without timing) and the moving average strategy) are compared. The Buy-and-Hold is a traditional trading strategy for value investment, the investors hold the stock after buying. Thus their profit directly reflects the market trends. The moving average strategy is often used in trend trading. In order to have an objective evaluation of

TABLE I
EXPERIMENT DATASET

Stock name	Period	Total Period	Training period	Testing period	Trend
DOW JONES	From	2016-01-04	2016-01-04	2020-01-01	Uptrend
	To	2020-10-09	2019-12-30	2020-10-09	
NASDAQ	From	2016-01-04	2016-01-04	2020-01-01	Uptrend
	To	2020-10-09	2019-12-30	2020-10-09	
APPLE	From	2016-01-04	2016-01-04	2020-01-01	Uptrend
	To	2020-10-09	2019-12-30	2020-10-09	
MSFT	From	2016-01-04	2016-01-04	2020-01-01	Uptrend
	To	2020-10-09	2019-12-30	2020-10-09	
COTY	From	2016-01-04	2016-01-04	2020-01-01	Downtrend
	To	2020-10-09	2019-12-30	2020-10-09	
ICON	From	2016-01-04	2016-01-04	2020-01-01	Downtrend
	To	2020-10-09	2019-12-30	2020-10-09	

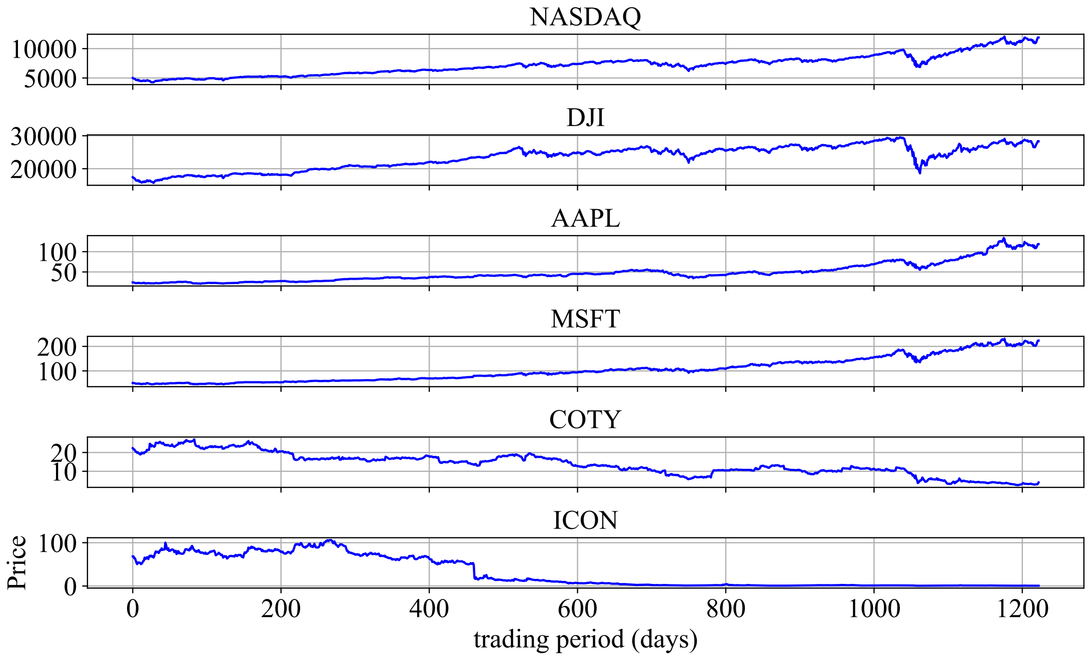


Fig. 4. Experimental dataset of stocks

our proposed model, six performance evaluation metrics are used for the performance comparisons, such as accumulated return(%AR), average annual return(%AAR), maximum drawdown, standard deviation, and Sharpe ratio(SR). %AR and %AAR measure the earning power of the trading strategy. Standard deviation and maximum drawdown measure the risk of the trading strategy. SR measures the return of strategy under risk. Mathematically, assuming the daily net return of the stock trading strategy is $R_t, t = 1, \cdots, T$, then the accumulated return

$$\%AR = (\prod_{t=1}^{T}(1 + R_t) - 1) \times 100 \tag{16}$$

. The average annual return $AAR = \frac{252}{T}\sum_{t=1}^{T} R_t$. The Standard Deviation

$$\sigma = (252)^{1/2}\sqrt{\frac{\sum_{t=1}^{T}(R_t - \bar{R})^2}{T}} \tag{17}$$

, where $\bar{R} = 1/T\sum_{t=1}^{T} R_t$. The maximum drawdown(MDD) is the maximum observed loss from a peak to a trough of a portfolio before a new peak is attained. Maximum drawdown is an indicator of downside risk over a specified time period, and it is defined as

$$MDD = max(\frac{p_i - p_j}{p_i}) \tag{18}$$

where p_i is the net value of the trading strategy at time i and j is the value at a time after i. and Sharpe ratio is equal to $\frac{AAR-r_f}{\sigma}$, where r_f is the risk-free rate, set at 0.03.

D. Experiment result

We discuss the results of experiments in this section. The benchmarks are the Buy-and-Holding strategy and the Moving Average Strategy(Crossovers). Table I shows the division of experimental data set in training and testing data. We perform extensive experiments on two stock indices (DOW JONES and NASDAQ) and four individual stocks(AAPL, MSFT, COTY, ICON) with different trends(upward trend and downward trend). The agent is trained by proximal policy optimization algorithm, and the hyperparameters are set in sectionIV-B. The comparison of %AR of stock timing

TABLE II
PERFORMANCE OF ALL THE TRADING STRATEGIES IN TERMS OF % ACCUMULATED RETURN.

Stock name	Buy-and-Hold	Moving Average Strategy	PPO Enhancement Strategy	Trend
DOW JONES	-2.1	2.2	23.9	Uptrend
NASDAQ	26.9	29.0	59.4	Uptrend
APPLE	50.0	61.8	92.8	Uptrend
MSFT	35.4	4.1	92.2	Uptrend
COTY	-70.0	-44.8	5.7	Downtrend
ICON	-52.3	-35.6	39.3	Downtrend

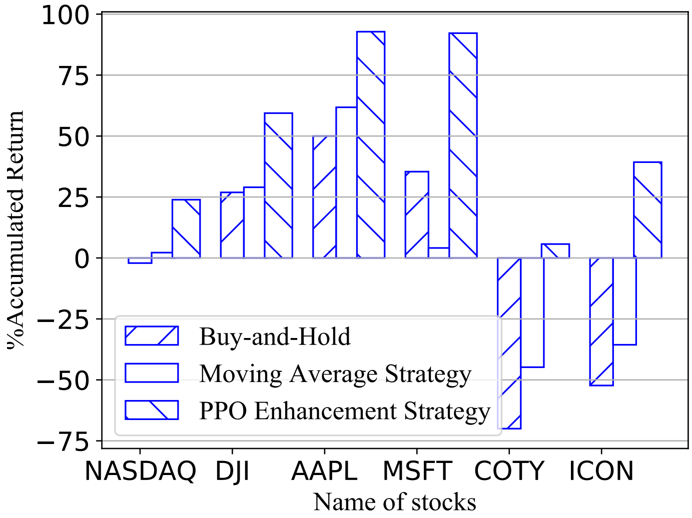


Fig. 5. The %Accumulated return of different strategies.

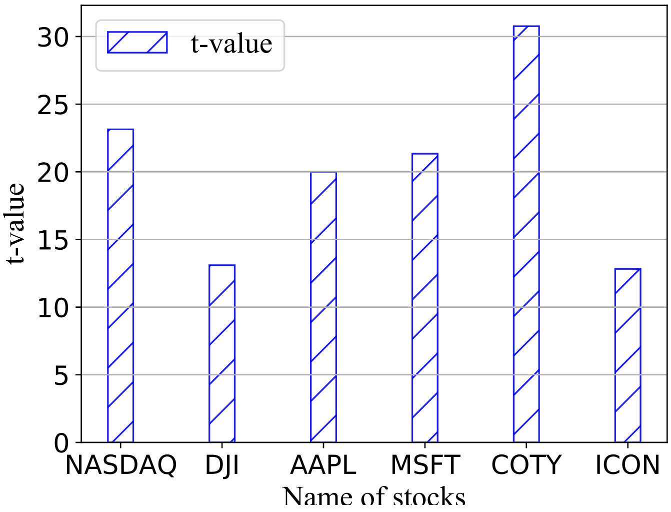


Fig. 6. Result of the Students t-Test (t-values) between Moving Average Strategy and the PPO Enhancement Strategy

strategy on two stock indices and four individual stocks with different price trends are recorded in Table II and the comparison of the performance of the three different trading strategies in terms of percentage accumulated return in the test period shown in Fig 5. From the Table II and Fig 5, we can find that the Moving Average Strategy performs slightly better than the Buy-and-Hold Strategy on the stock indices and individual stock AAPL. In the downtrend stocks, the Moving Average Strategy avoids 25.2% and 16.7% loss compared with the Buy-and-Hold Strategy respectively on stock COTY and ICON. However, it performs worse than the Buy-and-Hold Strategy on the stock MSFT. Though the Moving Average Strategy is used frequently in trends trading, it is not a good timing strategy according to the results in experiments. The PPO Enhancing Strategy based on the base strategy(the Moving Average Strategy) performs much better than the base strategies on the two indices and four stocks in terms of %accumulated return, whether the trend is upward or downward. Fig 8 shows the net value among different timing strategies. The green, blue, and red lines represent the net value during the test period of the PPO Enhancing Strategy, the Buy-and-Hold Strategy, and the Moving Average Strategy respectively. From Fig 8, the proposed PPO Enhancing Strategy outperforms the two benchmarks on four stocks and two indices in testing datasets.

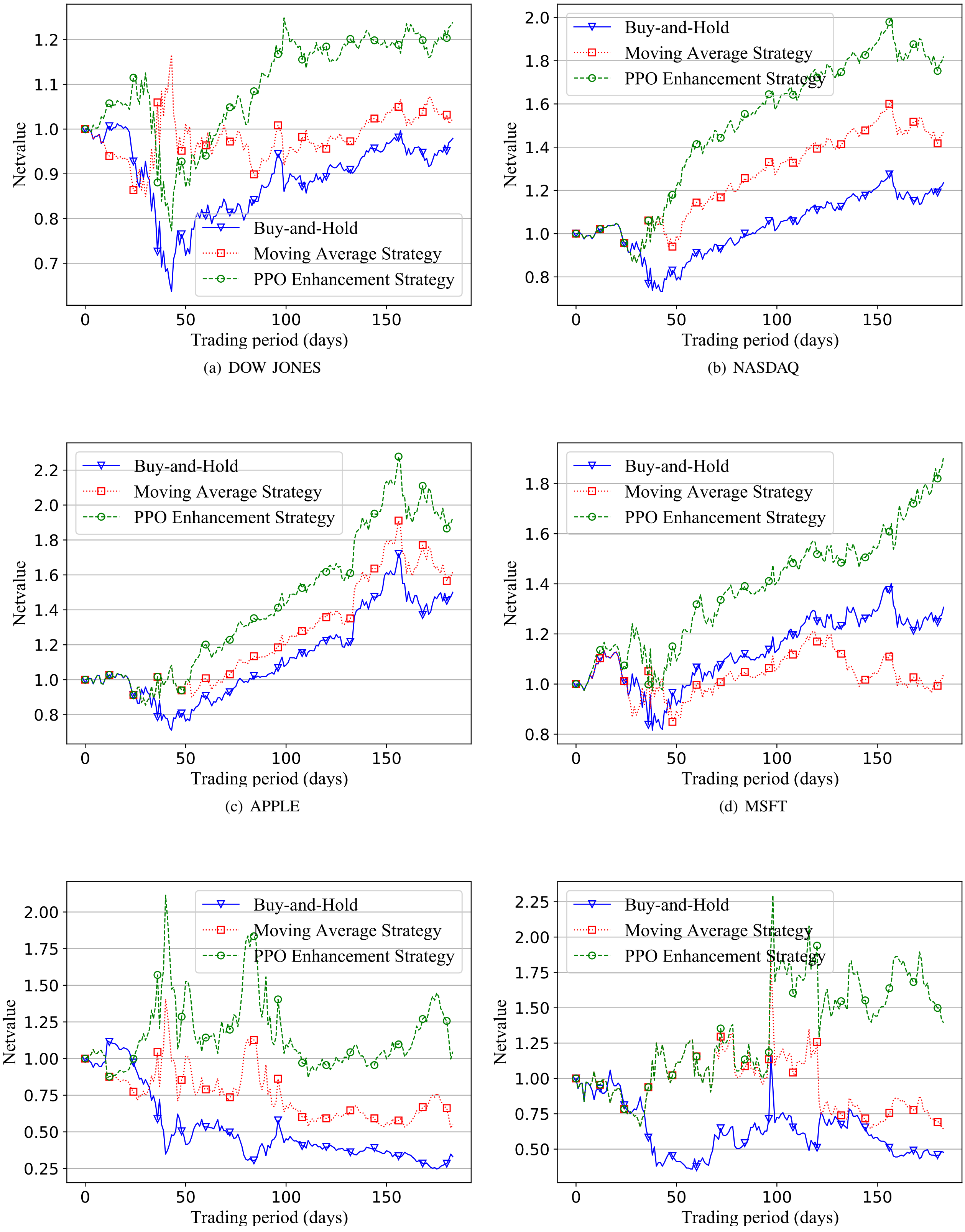
We perform students t-test on the daily value of the PPO Enhancement Strategy and the two benchmarks to check whether the results are statistically different. Fig 6 and Fig 7 show the t-value of different stocks. All the t-values are higher than 1.96 (the level of significance). So that the results are statistically significant, that is, the PPO Enhancement Strategy outperforms the two benchmarks.

In order to verify the robustness and reliability of the proposed PPO Enhancing Strategy, a relatively comprehensive comparison is shown in Table III. It concludes that on two indices and four stocks, the proposed PPO Enhancement Strategys accumulated return, annual return, Sharpe ratio, and Sortino ratio is better than two benchmarks, which shows that the model has good profitability and generalization ability compared with the benchmarks. The maximum drawdown of the proposed PPO Enhancement Strategy is smaller than other strategies. The standard deviation of the three strategies are almost the same, that is, the PPO Enhancement Strategy revises the base timing strategy very few times during the trading period, so the turnover rate of PPO Enhancement Strategy is almost the same as the base timing strategy.

V. Conclusion

In this paper, we propose a PPO Enhancement Strategy to improve the performance of a base stock timing strategy, instead of an end-to-end timing strategy. In this way, the base stock timing strategy can be modeled by the technical analysis and fundamental analysis so that it can be a white box strategy. This can increase the interpretability of the model.

The proposed PPO Enhancement Strategy based on proximal policy optimization algorithm can improve the perfor-



(a) DOW JONES

(b) NASDAQ

(c) APPLE

(d) MSFT

(e) COTY

(f) ICON

Fig. 8. The Net Value of Different Timing Strategy

TABLE III
TRADING PERFORMANCE ON TEST DATASET OF TREE STOCK TRADING STRATEGIES.

Stock name	Performance evaluation metrics	Buy-and-Hold	Moving Average Strategy	PPO Enhancement Strategy
Nasdaq	Annual Return	0.34	0.70	1.28
	Accumulated Return	0.23	0.47	0.81
	Annual Std	0.40	0.40	0.399
	Shrap Ratio	0.93	1.53	2.27
	Stino Ratio	1.27	2.28	3.69
	Maximum Drawdown	-0.30	-0.17	-0.17
DJI	Annual Return	-0.03	0.03	0.34
	Accumulated Return	-0.02	0.02	0.24
	Annual Std	0.42	0.42	0.42
	Shrap Ratio	0.14	0.28	0.91
	Stino Ratio	0.19	0.41	1.32
	Maximum Drawdown	-0.37	-0.23	-0.31
AAPL	Annual Return	0.75	0.94	1.47
	Accumulated Return	0.50	0.62	0.93
	Annual Std	0.51	0.51	0.51
	Shrap Ratio	1.35	1.55	2.03
	Stino Ratio	2.00	2.26	3.07
	Maximum Drawdown	-0.31	-0.18	-0.18
MSFT	Annual Return	0.45	0.06	1.43
	Accumulated Return	0.31	0.04	0.91
	Annual Std	0.49	0.50	0.49
	Shrap Ratio	0.99	0.36	2.06
	Stino Ratio	1.46	0.50	3.19
	Maximum Drawdown	-0.28	-0.25	-0.22
COTY	Annual Return	-0.78	-0.56	0.07
	Accumulated Return	-0.67	-0.45	0.05
	Annual Std	1.13	1.13	1.13
	Shrap Ratio	-0.78	-0.16	0.62
	Stino Ratio	-1.10	-0.23	0.90
	Maximum Drawdown	-0.78	-0.63	-0.59
ICON	Annual Return	-0.64	-0.45	0.58
	Accumulated Return	-0.52	-0.36	0.39
	Annual Std	1.42	1.42	1.42
	Shrap Ratio	-0.06	0.24	0.99
	Stino Ratio	-0.10	0.39	1.66
	Maximum Drawdown	-0.66	-0.65	-0.44

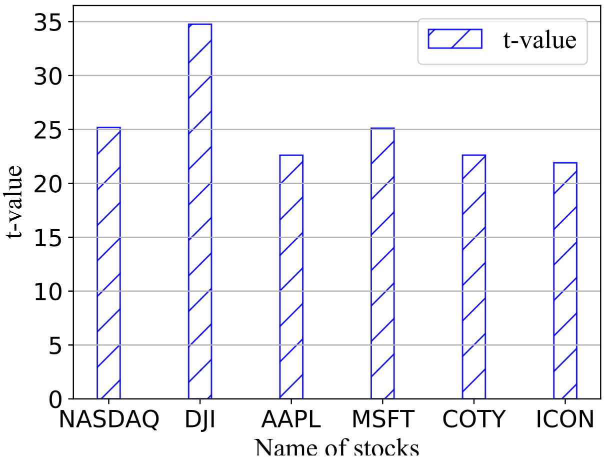


Fig. 7. Result of the Students t-Test (t-values) between Hold-and-Buy and the PPO Enhance Strategy

mance of the base stock timing strategy. In our experiments, the base strategy is a Moving Average Strategy, a very frequently used trading strategy. The actions of the reinforcement learning agent are defined as following or reversing the base strategy, rather than longing or shorting stocks like other end-to-end trading strategies developed by reinforcement learning or deep learning. In order to utilize the information of the base strategy, we apply state augmentation methods, regarding the trading signal of the base strategy as a part of states in the reinforcement learning environment.

The experimental results in the Section IV show that the performance of the proposed PPO Enhancement Strategy is independent of stock price trends and its performance is better than the two benchmarks: the Buy-and-Hold Strategy and the Moving Average Strategy in terms of different evaluation criteria(return, risk, and risk-adjusted return criteria). We find the standard deviation of the PPO Enhancement Strategy and the base timing strategy is almost the same by analyzing their standard deviation in Table III. that is, the PPO Enhancement Strategy revises the base timing strategy very few times during the trading period, so the turnover rate of PPO Enhancement Strategy is almost the same as the base timing strategy.

In the future, this study can be further improved in several ways. Firstly, in this paper, we use reinforcement learning to improve the performance of the stock time strategy. We will use the same ideas to apply reinforcement learning into portfolio selection. Secondly, the experiments performed on daily data, and the proposed model can use for other more high-frequency stock data such as hourly data, tick data. Thirdly, we only consider the base stock trading strategy with the Moving Average Strategy, however, other stock timing strategies such as the trading strategy based on decision trees, logistic regression also can be enhanced by our methods.

TABLE IV
THE HYPER-PARAMETERS OF THE PROXIMAL POLICY OPTIMIZATION ALGORITHM.

Parameters	value
learning rate	0.0003
varepsilon	0.2
gamma	0.8
gae_lambda	0.95
buffer size	128
batch_size	64
max_grad_norm	5
activate function	RELU
input layer	7
the first hidden layer	64
the second hidden layer	64
the third hidden layer	64
the forth hidden layer	16
output layer	2

APPENDIX

Table IV shows the hyperparameters of experiments in Section IV.

Fig 9 shows the training process of enhancing the moving average strategy using proximal policy optimization algorithm. The x-axis represents training timesteps, and y-axis represents the mean reward of episode.

REFERENCES

[1] T. Vaga, *Profiting from chaos: using chaos theory for market timing, stock selection, and option valuation*. Tonis Vaga, 1994.

[2] K. Lee and G. Jo, "Expert system for predicting stock market timing using a candlestick chart," *Expert Systems with Applications*, vol. 16, no. 4, pp. 357–364, 1999.

[3] M. Gomes, V. Magnani, T. Albanez, and M. Valle, "dataset on share issuance, abnormal returns and market timing in the Brazilian stock market," *Data in Brief*, vol. 25, p. 104251, 2019.

[4] K. Baltakys, M. Baltakienė, H. Kärkkäinen, and J. Kanniainen, "neighbors matter: Geographical distance and trade timing in the stock market," *Finance Research Letters*, vol. 31, 2019.

[5] Q. Zhuge, L. Xu, and G. Zhang, "Lstm neural network with emotional analysis for prediction of stock price." Engineering Letters, vol. 25, no. 2, pp167-175, 2017.

[6] J. B. Chakole, M. S. Kolhe, G. D. Mahapurush, A. Yadav, and M. P. Kurhekar, "A q-learning agent for automated trading in equity stock markets," *Expert Systems with Applications*, vol. 163, p. 113761, 2021.

[7] A. S. Girsang, F. Lioexander, and D. Tanjung, "Stock price prediction using lstm and search economics optimization." IAENG International Journal of Computer Science, vol. 47, no. 4, pp758-764, 2020.

[8] A. Wibowo *et al.*, "Weight-dropped long short term memory network for stock prediction with integrated historical and textual data." IAENG International Journal of Computer Science, vol. 47, no. 3, pp367-377, 2020.

[9] Z. Ge, W. Wang, and D. Chen, "Predicting index returns from the market structure disagreement: Evidence from China." Engineering Letters, vol. 28, no. 4, pp1063-1074, 2020.

[10] A. Gunasekarage and D. M. Power, "The profitability of moving average trading rules in south asian stock markets," *Emerging Markets Review*, vol. 2, no. 1, pp. 17–33, 2001.

[11] K. Haegeman, E. Marinelli, F. Scapolo, A. Ricci, and A. Sokolov, "Quantitative and qualitative approaches in future-oriented technology analysis (fta): From combination to integration?" *Technological Forecasting and Social Change*, vol. 80, no. 3, pp. 386–397, 2013.

[12] W. Kross and D. A. Schroeder, "An empirical investigation of the effect of quarterly earnings announcement timing on stock returns," *Journal of Accounting Research*, pp. 153–176, 1984.

[13] H. DeAngelo, L. DeAngelo, and R. M. Stulz, "Fundamentals, market timing, and seasoned equity offerings," *NBER Working Paper*, no. w13285, 2007.

[14] V. H. Shah, "Machine learning techniques for stock prediction," *Foundations of Machine Learning— Spring*, vol. 1, no. 1, pp. 6–12, 2007.

[15] S. Shen, H. Jiang, and T. Zhang, "Stock market forecasting using machine learning algorithms," *Department of Electrical Engineering, Stanford University, Stanford, CA*, pp. 1–5, 2012.

[16] E. Gibney, "Google ai algorithm masters ancient game of go," *Nature News*, vol. 529, no. 7587, p. 445, 2016.

[17] T.-S. Quah, "Using neural network for DJIA stock selection." Engineering Letters, vol. 15, no. 1, pp126-133, 2007.

[18] Z. Jiang, D. Xu, and J. Liang, "A deep reinforcement learning framework for the financial portfolio management problem," arXiv. org, Tech. Rep., 2017.

[19] F. Soleymani and E. Paquet, "Financial portfolio optimization with online deep reinforcement learning and restricted stacked autoencoderdeepbreath," *Expert Systems with Applications*, vol. 156, p. 113456, 2020.

[20] C. Betancourt and W.-H. Chen, "Deep reinforcement learning for portfolio management of markets with a dynamic number of assets," *Expert Systems with Applications*, vol. 164, p. 114002, 2021.

[21] Y. Yuan, W. Wen, and J. Yang, "Using data augmentation based reinforcement learning for daily stock trading," *Electronics*, vol. 9, no. 9, p. 1384, 2020.

[22] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, and H. Fujita, "Adaptive stock trading strategies with deep reinforcement learning methods," *Information Sciences*, 2020.

[23] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International Conference on Machine Learning*, 2015, pp. 1889–1897.

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[25] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[26] D. Aberdeen *et al.*, "Policy-gradient algorithms for partially observable markov decision processes," 2003.

[27] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in *Advances in Neural Information Processing Systems*, 2000, pp. 1008–1014.

[28] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[29] M. Metghalchi, J. Marcucci, and Y.-H. Chang, "Are moving average trading rules profitable? evidence from the European stock markets," *Applied Economics*, vol. 44, no. 12, pp. 1539–1559, 2012.

[30] T. Kilgallen, "Testing the simple moving average across commodities, global stock indices, and currencies," *The Journal of Wealth Management*, vol. 15, no. 1, pp. 82–100, 2012.

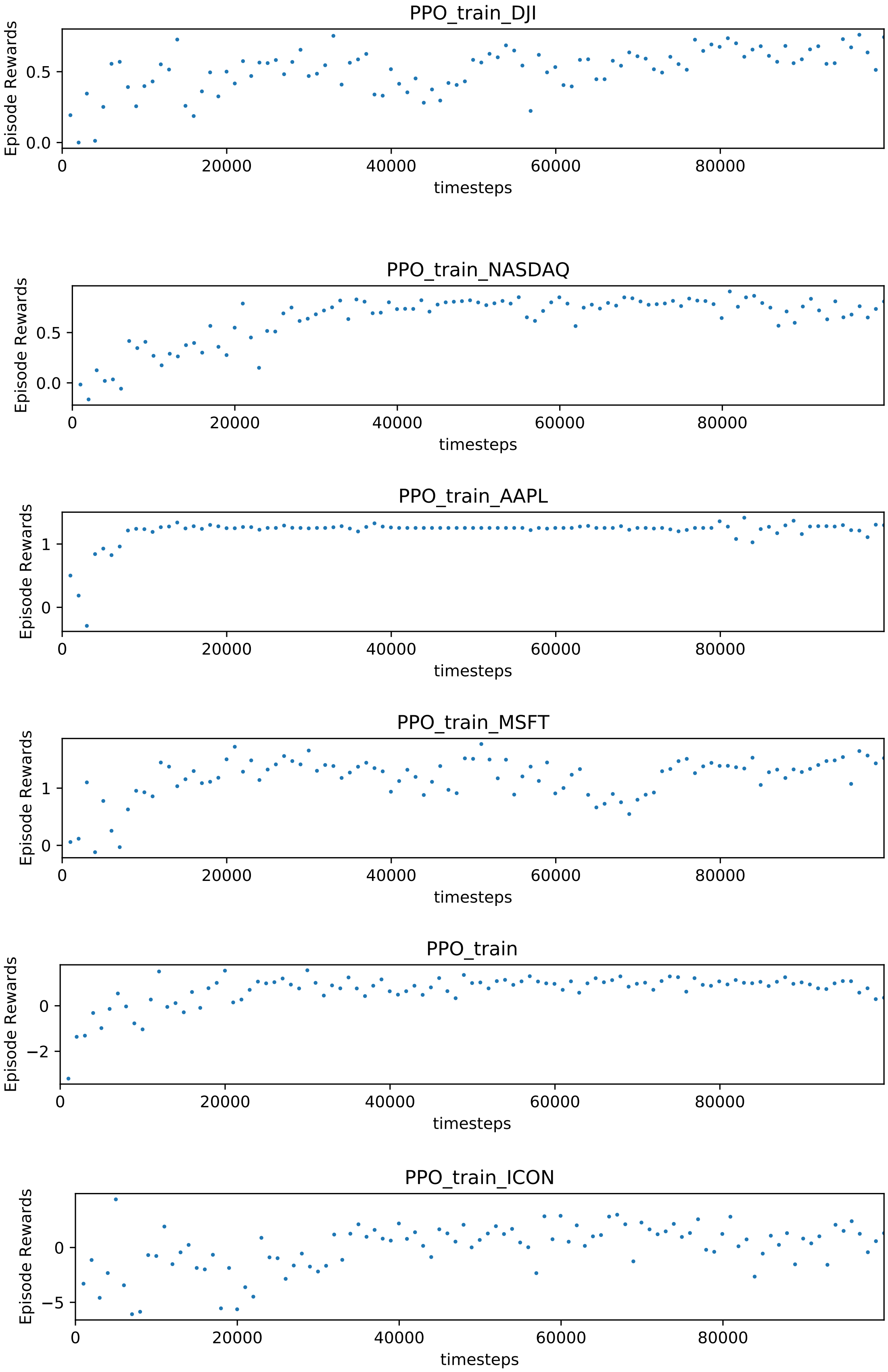


Fig. 9. The training process of different stocks