Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 32(9): 5176–5200.
DOI: 10.3934/era.2024239
Received: 30 May 2024
Revised: 06 August 2024
Accepted: 21 August 2024
Published: 03 September 2024

Research article

Reinforcement learning for deep portfolio optimization

Ruyu Yan[1], **Jiafei Jin**[1,*] **and Kun Han**[2]

[1] School of Management, Harbin Institute of Technology, Harbin 150000, China
[2] Faculty of Computing, Harbin Institute of Technology, Harbin 150000, China

* **Correspondence:** Email: jinjiafei@hit.edu.cn.

Abstract: Portfolio optimization is an important financial task that has received widespread attention in the field of artificial intelligence. In this paper, a novel deep portfolio optimization (DPO) framework was proposed, combining deep learning and reinforcement learning with modern portfolio theory. DPO not only has the advantages of machine learning methods in investment decision-making, but also retains the essence of modern portfolio theory in portfolio optimization. Additionaly, it was crucial to simultaneously consider the time series and complex asset correlations of financial market information. Therefore, in order to improve DPO performance, features of assets information were extracted and fused. In addition, a novel risk-cost reward function was proposed, which realized optimal portfolio decision-making considering transaction cost and risk factors through reinforcement learning. Our results showed the superiority and generalization of the DPO framework for portfolio optimization tasks. Experiments conducted on two real-world datasets validated that DPO achieved the highest accumulative portfolio value compared to other strategies, demonstrating strong profitability. Its Sharpe ratio and maximum drawdown also performed excellently, indicating good economic benefits and achieving a trade-off between portfolio returns and risk. Additionally, the extraction and fusion of financial information features can significantly improve the applicability and effectiveness of DPO.

Keywords: portfolio optimization; transaction costs; risk management; deep learning; reinforcement learning

1. Introduction

Portfolio management process can be viewed as an information process and an execution process. The difference between information value and execution cost is the value investors want to capture. The goal of portfolio management is to dynamically allocate and maximize returns in portfolio. The success of investors largely depends on maintaining a good portfolio balance. However, this task is difficult for investors. With the development of big data and artificial intelligence (AI) trends, intelligent portfolio

selection methods based on deep learning (DL) [1] and machine learning (ML) methods [2, 3] have been proposed and have achieved results.

Although recent trends in AI have brought solutions to the investment industry, portfolio management is still largely based on linear models and modern portfolio theory (MPT). MPT suggests that portfolios focus more on risk than returns [4]. The covariance between various investment returns is introduced into the portfolios to measure risk. Portfolios are determined by the various securities and their weights. MPT relies on accurate predictions of market prices and certain restrictive assumptions (such as the assumption that the probability distribution of past asset returns fully represents the future), and studies the mean-variance portfolio selection problem. MPT does not assume transaction costs in the portfolio trading process [4]. Transaction costs are important in portfolio trading as they can easily erode the returns of trading strategies. Simultaneously, transaction costs analysis helps to better manage portfolios [5]. When considering transaction costs, a portfolio optimal strategy is achieved by optimizing the expected rebalancing of logarithmic returns [6], but this work overlooks the risk in the portfolio trading process. Risk management is essential within the framework of portfolio selection or asset allocation [7]. Therefore, it is imperative for portfolio investors to improve MPT and methods to more effectively achieve their objectives. Essentially, asset portfolio optimization involves continuous decision-making. That is to say, it continuously reallocates some funds into assets based on the historical information to achieve investment goals.

Reinforcement learning (RL) is an appropriate framework for dealing with complex data and challenging decision-making processes [8]. It can be effectively applied to solve problems related to decision process construction [9]. The process of asset trading can be naturally expressed as the process of RL. In the process of asset portfolio management, the agent takes action on the environment based on the observation of its state. As a result of accessing the state and taking action, the agent receives reward. In specific asset trading cases, the environment can be considered as the recent trading information of assets, actions are the trades that remove some assets held by the agent and obtain new assets, rewards are scalar functions of the gains or losses obtained by the agent taking these actions, and the agent can be thought of as an algorithm that interacts with the market by observing environment, balancing portfolios, and making decisions. Its objective is to maximize returns and minimize risk.

However, the complexity of portfolio selection also makes portfolio management challenging. Although attempts have been made to apply RL to financial portfolio management, these approaches may be limited in practice.

1) Environmental uncertainty. It includes two aspects: First, the inherent non-stationarity of financial market, which is characterized by high noise, randomness, and chaos. Second, the uncertainty caused by distribution shifts between training data and test data. Financial investments are highly sensitive to market fluctuations, and most existing models cannot adapt to the non-stationarity of the market and distribution shifts between data. To address these challenges, we propose a novel deep portfolio optimization (DPO) framework for portfolio optimization. First, the DPO framework is a universal framework with theoretical generalizability. It combines portfolio theory with RL learning paradigm and is not limited to any specific market. DPO can adapt to the inherent non-stationarity of financial markets and uses financial data information to explore optimal portfolio decisions. Second, our DPO framework employs RL and DL to optimize portfolio problems. DPO combines convolutional neural network (CNN) and recurrent neural network (RNN) modules within the RL

framework, utilizing these two modules to learn and extract features from financial information in a targeted manner. The CNN module learns local time features to study asset correlations, while the RNN module learns long-sequence time features to study sequential correlations. They address the lag issue in distribution shifts between data, improving the performance of DPO. Third, we empirically test the effectiveness and generalizability of the DPO framework. We tested and performed well on two real-world datasets: the cryptocurrency dataset and the Chinese A-share dataset. The performance of DPO surpasses baseline models in terms of accumulative portfolio value (APV). Empirical results confirm that DPO can address the challenges posed by environmental uncertainty.

2) Transaction costs and risk. The practicality principle of financial markets requires controlling transaction costs and risk. On one hand, neglecting transaction costs in investment decisions may lead to aggressive trading and introduce bias in estimating returns. On the other hand, ignoring the risk due to fluctuations in returns may lead to disastrous results in practice. Most existing methods only focus on one of these aspects without simultaneously constraining both influencing factors, which may limit their practical performance. Therefore, to assess the economic significance of portfolio performance, it is essential to incorporate transaction costs and risk into the analysis. We adopt the reward shaping method and propose a reward function that considers transaction costs and risk factors. Applying RL to optimize this reward function makes it easier to achieve higher returns and incur smaller risk losses. The performance of DPO on two real-world datasets also exceeds that of baseline models in terms of Sharpe ratio (SR) [10] and maximum drawdown (MDD) [11]. The empirical validation confirms the rationality and usefulness of the shaped reward function within the DPO framework.

Our contributions are as follows:

1) A novel DPO framework is proposed, combining portfolio theory, DL methods, and RL learning paradigm. DPO leverages the advantages of financial theory and RL, focusing on risk premiums and transaction costs to improve the robustness of portfolio optimization.

2) The performance of the DPO framework is improved by applying DL. Extracting and fusing features from financial information using CNN and RNN is highly effective in improving the performance of DPO. Empirical experiments are conducted on real datasets to make optimal portfolio decisions.

3) A novel reward function is proposed, which is grounded in MPT and thoroughly considers transaction costs and risk volatility in the portfolio decision-making process. By shaping the reward function in RL, DPO can achieve maximum cumulative returns while controlling transaction costs and risk.

The remaining parts of this paper are organized as follows. Section 2 reviews related work on portfolio optimization using DL methods and RL, focusing on transaction costs and risks in portfolio trading. Section 3 introduces the architecture of DPO, which includes DL modules, the RL framework, and the risk-cost reward function. Section 4 presents the experimental results. Section 5 presents the conclusions of the paper.

2. Related work

The prediction task of financial time series is very difficult and challenging due to the high noise, high frequency, and generally accepted semi-strong form of market efficiency in the stock

market [12]. Therefore, stock prediction holds significant importance. There is increasing attention from both academia and investors on portfolio forecasting. There has been a considerable amount of research endeavors to utilize AI and ML to predict stock market returns and volatility.

2.1. Portfolio optimization based on CNN or RNN

With the availability of large-scale market data and the rapid development of DL applications in big data, it is natural to utilize DL models to explore potential patterns in portfolio management. Financial information has the periodicity and tendency of time series. Schmidhuber [13] believed that deep artificial neural network (including RNN) was widely used in the fields of pattern recognition and ML. In temporal data analysis, RNN can be used to extract and capture these complex temporal features, and is used for prediction and classification of time series. Fischer and Krauss [14] used RNN to predict the trend of S&P 500 index components from 1992 to 2015. Compared with application benchmarks, RNNs have better prediction performance and are largely robust in their impact on microstructure. Vidal and Kristjanpoller [15] believed that predicting the future volatility of assets was important and significantly improved the prediction of portfolio volatility by adopting DL methods. It provided various information related to the characteristics of series, taking different lags of profitability as inputs, enabling it to learn and gain returns from the temporal structure. Ma et al. [16] utilized ML models such as random forest, support vector regression, long short-term memory (LSTM), deep multilayer perceptron, and CNN to optimize portfolios.

Existing techniques for training neural network models to predict market behavior have shown their effectiveness in asset price forecasting and asset allocation [17]. However, the lack of interaction between DL models and the market puts them at a disadvantage in decision-making problems such as portfolio management.

2.2. Portfolio optimization based on RL

Due to the continual updates of financial market information, numerous researches employ RL techniques to make timely and continual decisions. RL has already achieved some success in addressing continuous decision-making problems [18]. RL has been widely applied in the financial field. Almahdi and Yang [8] proposed a recurrent RL method that used a continuous risk adjusted performance objective function (Calmar ratio) to obtain buy and sell signals and asset allocation weights. They used a portfolio composed of exchange-traded funds traded on the most active exchanges. The results showed good performance of portfolio optimization. Jiang et al. [1] used a model-free deterministic policy gradient (DPG) to dynamically optimize a cryptocurrency portfolio. The research maximized cumulative returns through RL, utilizing price information and the DPG method to make more profitable decisions. Similarly, Jang and Seong [19] used the deep deterministic policy gradient (DDPG) and optimized asset portfolios. They addressed the portfolio problem by inputting Tucker decomposition of technical analysis and stock returns.

However, existing researches in portfolio optimization lack integration with actual economic benefits, focusing solely on technical aspects [16–18]. This gap leads to biases in investment returns and limits the effectiveness of portfolio decision-making.

2.3. Transaction costs and risk in portfolio optimization problems

In order to assess the realistic significance of portfolio performance, it is ultimately necessary to include transaction costs and risk factors in the analysis. Compared to previous research on financial markets, the strong liquidity of the cryptocurrency market and the Chinese market is the most important predictive factor [3, 9], which prompts us to carefully study the impact of transaction costs and risk factors.

Transaction costs are related to buying and selling business, whether it is creating an initial portfolio with cash or rebalancing an existing portfolio. In the financial market, transaction costs mainly consist of three parts: commission, stamp tax, and slippage. Due to liquidity issues, it is difficult to execute all trades at predetermined prices [3]. Obviously, neglecting these costs can lead to inefficient portfolio management, as they heavily impact realized returns [20].

Recently, RL has been applied to solve portfolio optimization problems. Ma and Li [21] adopted a robust portfolio selection problem using DL with limited attention. The agent can access risk-free assets and stocks in the financial market and achieve higher expected returns. However, these methods ignore asset correlation when extracting portfolio features and do not control the costs during the optimization process, resulting in limited performance. García-Galicia et al. [22] proposed managing and optimizing portfolios in the context of continuous-time RL. However, this work ignores the influence of risk factors on portfolio selection. Investors are concerned about risk and returns. MPT is about optimizing investor behavior [23]. Ban et al. [24] adopted two ML methods, which introduced performance-based regularization to optimize portfolio returns. Due to the drawbacks of investor irrationality in MPT, we shape a reward function to optimize risk and transaction costs, with the aim of constraining the estimated sample variance of portfolio returns and risk, so that the estimation error can develop in a less correlated direction. Therefore, we focus on the trade-off between portfolio risk and robustness, which improves the stability of portfolio allocation and reduces portfolio risk.

To sum up, this paper applies the RL learning paradigm to interact with market information for portfolio optimization. DL modules are utilized to analyze the temporal and asset correlations of financial information in investment strategies. Additionally, the focus is on the trade-off between portfolio returns and risk volatility to achieve maximum expected returns.

3. Methodology

This section describes the process of combining MPT with RL and constructing a portfolio optimization framework. To begin, the general architecture is introduced and the portfolio task is described in detail. Then, the DL modules are presented. Additionally, the RL framework applicable to portfolio tasks is introduced. Finally, the risk-cost reward function proposed by combining portfolio theory is described.

3.1. Problem formulation

Portfolio management is a fundamental financial planning task that aims to maximize returns or minimize risk through asset allocation. We consider a portfolio selection task for $n+1$ assets in the time t in a financial market, including one risk-free asset and n risky assets. In time t, we represent the prices of all assets as $p_t \in \mathbb{R}_+^{(n+1)\times d}$, where each row $p_{i,t} \in \mathbb{R}_+^d$ represents the features of the asset, and $d = 4$

represents the number of prices. Specifically, we consider four types of asset prices, namely opening price, highest price, lowest price, and closing price. The asset price information from two real-world markets (the cryptocurrency market and Chinese A-share market) exhibit both temporal correlations and asset correlations. They can be summarized into more prices to obtain more information, making the price vector $p_t = \{p_{t-k}, \ldots, p_{t-1}\}$, where k is the length of the price sequence. $p_{i,t}$ represents the closing price of the i asset at time t, where $i = \{1, \ldots, n\}$, n is the number of assets in the investment portfolio. Price vector p_t is composed of the closing prices of all n assets. Similarly, p_t^H and p_t^L represent the highest and lowest prices during time t. In portfolio problems, assets are not entirely used for investment. Specifically, when making portfolio decisions, it is not necessary to purchase every asset. Instead, we should observe market conditions and interact with market information to make optimal choices. Not all assets need to be included in the investment decisions.

In addition to the n asset portfolios, we introduce into p_t the price vector $p_{0,t}$ of an extra dimension (the first dimension indexed by 0), representing the risk-free asset price at time t. Assuming that the price of risk-free asset changes little and ignoring the influence of inflation or deflation factors, it can be considered that the asset is risk-free, and its price remains unchanged, that is, $p_{0,t} = 1$, $\{\forall t | p_{0,t} = 1\}$. It has little impact on the learning process and we exclude risk-free asset from the input. When the price vector $p_t \in \mathbb{R}^{m \times n \times 4}$ is normalized with respect to risk-free asset value for all time variables, $p_{0,t}$ remains constant for all continuous time t.

The change in asset prices over time during time t, $y_t = \frac{p_{t+1}}{p_t}$ is represented as the price relative vector, specifically represented as

$$y_t = \frac{p_{t+1}}{p_t} = (1, \frac{p_{1,t+1}}{p_{1,t}}, \ldots, \frac{p_{n,t+1}}{p_{n,t}})^\top \tag{3.1}$$

To facilitate mathematical expression of the process of asset reallocation in portfolio optimization, we introduce the weight score of asset reallocation

$$w_t = (w_{0,t}, w_{1,t}, \ldots, w_{n,t})^\top \tag{3.2}$$

where $w_{i,t}$ $(t \neq 0)$ is the weight score of the i asset, and $w_{0,t}$ is the weight score of risk-free asset at the end of time t. Assets will be reallocated into portfolios based on the weight assigned to each asset, that is, $\sum_{i=0}^{n} w_{i,t} = 1$.

In the experiments of this paper, the agent in RL sells or buys assets based on the difference of w_t and w_{t-1} between the time step $t-1$ and t in order to rebalance the portfolio.

3.2. DL modules

3.2.1. Correlation feature module

The correlation between assets in the financial market reveals the relationship between macro market trends and individual assets. We use CNN to asymptotically extract asset correlations without altering the structure of asset features. By employing highly reusable convolutional kernels, we can extract asset correlations quickly and with reduced risk of overfitting. The correlation feature module uses CNN to analyze the correlations between features at the current time point and adjacent time points in local observations, thereby extracting asset correlation features. This allows us to obtain more reliable price information.

The input is asset price information vectors, and the output is the correlation feature vectors of each asset. Considering the characteristics of stocks, we use a 1D CNN to extract asset correlation feature vectors, applying a convolutional kernel (size 1×3) to each stock for correlation feature vectors extraction. Fully considering the correlations between financial assets helps to obtain more information about asset movements, leading to better portfolio decisions.

3.2.2. Sequence feature module

Stock data in financial markets is characterized by high noise and temporal correlation. Considering these characteristics, we select LSTM to extract features from historical financial market information. The price sequence of assets reflects the price changes of each asset. We process historical price sequences using the sequence feature module. The sequence feature module uses LSTM to analyze the temporal correlation of financial data, extract temporal features, and optimize the portfolio.

We use LSTM to analyze and store the temporal information of financial assets. The input is asset price information vectors, and the output is the temporal feature vectors of each asset. Memory units and gates are used to store information over a long period, and the output represents the feature vectors of each financial asset. We concatenate the feature vectors of each asset to form an overall feature distribution, thereby achieving the extraction of feature vectors from financial temporal information. The sequence feature module can extract robust sequential features, helping to capture more price information and enabling the portfolio to achieve higher returns.

3.3. RL framework

Portfolio optimization is a continuous decision-making process. RL includes policy-based and value-based methods. Policy-based methods aim to directly learn and optimize the policy, enabling actions to be chosen in a given state to maximize cumulative rewards. Policy-based methods are particularly well-suited for handling problems with continuous action spaces, as they can flexibly output deterministic actions. Value-based methods suit for discrete action spaces. Therefore, in the portfolio optimization task, we select a policy-based method to make decisions.

We can formulate the optimization process of portfolio management as a generalized RL process, indicating that the next state only depends on the current state and action. We can describe the portfolio optimization problem as a tuple (S, A, P, R), where S is the state, A is the action, and R is the reward, determined by the reward function $S \times A \rightarrow R$, where P is the state transaction, representing the probability that the agent performs action a_t in state s_t and moves to the next state s_{t+1}. Action A does not affect the transaction state P, that is, the transaction state transition only depends on the environment.

DPG is an algorithm within the learning paradigm of RL. It is a policy-based method that focuses on learning and optimizing the gradient of deterministic policies. DPG can solve continuous decision-making problems and exhibits well robustness. We use the DPG method to learn deterministic portfolio strategies. We represent the optimal strategy at each step using a probability distribution function, sample actions according to this distribution to obtain the current optimal action value, and then solve the deterministic policy gradient to derive the optimal portfolio strategy.

We use deep neural networks to directly output the policy function $\pi(s)$, which represents the action to be taken in state s. The chosen policy is deterministic, meaning that the action to be executed in a

given state s is definite and unique, described as $a = \mu(s)$.

Due to the nonstationary nature of the financial market environment, we train DPO by directly optimizing the reward function and utilizing policy gradient methods. The policy is a mapping from state space to action space $\pi_\theta : S \rightarrow A$, The strategy gradient is represented by the parameter probability distribution $\pi_\theta(a|s) = P(a|s;\theta)$, and we select action a from the action space of state s according to the parameter vector θ. For a deterministic policy $a = \mu_\theta(s)$, this choice is generated deterministically by the strategy from a state. Due to returns r_t^γ is defined as the total discount returns from time step t,

$$r_t^\gamma = \sum_{k=t}^{\infty} \gamma^{k-t} r(s_k, a_k) \tag{3.3}$$

where r is the reward function and γ is the discount factor, and $0 < \gamma < 1$. We define the performance objective as

$$J(\mu_\theta) = \mathbb{E}[r_1^\gamma|\mu] \tag{3.4}$$

It is the expected $\rho^\mu(s)$ of the discounted state distribution.

$$J(\mu_\theta) = \int_s \rho^\mu(s) r(s, \mu_\theta(s)) ds = \mathbb{E}_{s\sim\rho^\mu}[r(s, \mu_\theta(s))] \tag{3.5}$$

Considering the time interval from 1 to T, the corresponding objective function is

$$J_T(\mu_\theta) = \sum_{t=1}^{T} \gamma^t r(s_t, \mu_\theta(s_t)) \tag{3.6}$$

The objective in the above equation is a typical Markov decision process objective function. However, it should be noted that this type of function does not match the optimization objectives of portfolio [25], as returns accumulates at time t and will be reconfigured at time $t + 1$. We use the cumulative product of portfolio value instead of summing up:

$$J_T = \prod_{t=1}^{T} J_0 r_t \tag{3.7}$$

It is more suitable for portfolio optimization tasks. Therefore, the objective function is

$$J_T(\mu_\theta) = J_0 \prod_{t=1}^{T} r(s_t, \mu_\theta(s_t)) \tag{3.8}$$

where J_0 is a constant.

3.4. Risk-cost reward function

In the portfolio optimization task, RL is a learning paradigm used to describe and solve the problem of an agent interacting with the financial environment to learn strategies for maximizing returns and minimizing risk. The reward function is an essential component of the learning objective function in RL, representing the feedback signal received by the agent upon achieving a goal in the environment.

It directly influences the agent's learning process. The effectiveness of portfolio optimization is a result of the combined influence of RL and the risk-cost reward function.

3.4.1. Mean-variance portfolio

We propose a reward function combined with MPT to construct a portfolio strategy that matches investors' expected returns and risk tolerance. At the theoretical level of portfolio theory, we demonstrate and test the importance of the proposed reward function in portfolio optimization, considering transaction costs and risk factors.

We discuss the mean-variance problem:

$$\min_{w_{i,t}}\{\sum_{i,j=1}^{n} w_{i,t}w_{j,t}\sigma_{i,j} - \sum_{i=1}^{n} w_{i,t}\upsilon_{i,t}\} \tag{3.9}$$

$$s.t. \sum_{i=0}^{n} w_{i,t} = 1 \tag{3.10}$$

$$w_{i,t} \geq 0 \tag{3.11}$$

In Eq (3.9), $w_{i,t} \in \mathbb{R}^N$ is the weight vector of the portfolio. $\sigma_{i,j} \in \mathbb{R}^{N\times N}$ is the covariance matrix of the estimated returns of assets i and j, and $\upsilon_{i,t} \in \mathbb{R}^N$ is the estimated mean of the estimated returns of assets. In other words, $\upsilon_{i,t} \in \mathbb{R}^N$ is the expected return in time t. The first two items in Eq (3.9) capture the portfolio's risk-returns trade-off: the first is the portfolio return variance $\sum_{i,j=1}^{n} w_{i,t}w_{j,t}\sigma_{i,j}$, and the second is the portfolio return mean $\sum_{i=1}^{n} w_{i,t}\upsilon_{i,t}$.

To the best of our knowledge, it is the first time we combine the classical mean-variance model with RL while considering transaction costs and risk factors during the trading process. The reward function focuses on transaction costs and risk, which not only reduces the impact of estimation errors but also minimizes frequent trading, thereby improving portfolio returns.

3.4.2. Rewards in portfolio

As mentioned earlier, the total asset weight vector $w_t = (w_{0,t}, w_{1,t}, \ldots, w_{n,t})^\top$ at time step t represents the asset weight. What the agent needs to do is to reallocate assets, that is, adjust $\tilde{a}_t$.

In time step t, we expect to reallocate the weight

$$a_t = (a_{0,t}, a_{1,t}, \ldots, a_{n,t})^\top \tag{3.12}$$

$$s.t. \sum_{i=0}^{n} a_{i,t} = 1 \tag{3.13}$$

where Eq (3.12) is the action vector in the model.

By taking action on time step t, the asset allocation vector will be affected by price changes y_t. At the end of the time, the allocation vector $\tilde{a}_t$ is

$$\tilde{a}_t = \frac{y_t \otimes a_t}{y_t \cdot a_t} \tag{3.14}$$

where $\otimes$ is the multiplication of elements. In the t step, after making decision a_t, we need to rebalance $\tilde{a}_t$ to a_t from the current portfolio strategy, where $\tilde{a}_t$ and a_t are the asset allocation vectors before and after rebalancing.

In addition, the reward function for each time step can be defined in terms of the profits obtained by the agent. The fluctuation in the value of each asset is $a_{i,t} \cdot y_{i,t}$. Therefore, the total returns for time step t are $a_t \cdot y_t$.

To start, we combine MPT with the RL learning paradigm, considering transaction costs and risk factors during the trading process. By leveraging the advantages of MPT and using the DPG algorithm, we obtain the optimal portfolio vector w_t. In addition, we combine the reward function with MPT, incorporating it during the function shaping process, so that the agent considers the impact of MPT on the optimal portfolio strategy during learning. The purpose of combining financial theory into the reward function shaping is to make the agent's decisions more realistic and economically meaningful, with practical applicability. Finally, in the portfolio optimization task, the optimization of the portfolio is the result of the combined effects of RL and the risk-cost reward function. The asset vector w_t is optimized through the action vector a_t. Specifically, w_t is derived from the reward function shaped by MPT and the RL learning paradigm, where the agent interacts with the environment by analyzing the financial market state, taking actions, and receiving rewards. Through continuous optimization, the agent achieves optimal portfolio decisions.

Transaction costs can easily erode the returns of trading strategies. They can generally be modeled by the norm of portfolio transaction vectors [5, 25]. We use the one-norm transaction costs of the portfolio transaction vectors. Using the DPO framework and mean-variance portfolio theory, we demonstrate that portfolio optimization with one-norm transaction costs can reduce the impact of estimation errors and achieve optimal portfolio strategies in real-world scenarios.

Consider transaction costs,

$$c_t = \delta \sum_{t=2}^{T} \|a_t - \tilde{a}_t\|_1 \tag{3.15}$$

$$r_t = r(s_t, a_t) = a_t \cdot y_t - \delta \sum_{t=2}^{T} \|a_t - \tilde{a}_t\|_1 \tag{3.16}$$

where c_t is the transaction cost incurred by the portfolio during trading.

Consider the risk factors,

$$\varphi^2(r_t) = \sum_{t=1}^{T} a_{i,t} a_{j,t} \sigma_{i,j} \tag{3.17}$$

$\varphi^2 \ln(r_t)$ is the trade-off between risk and returns. The objective of the agent is to maximize the accumulative portfolio value, which is equivalent to maximizing the sum of logarithmic values. Finally, the reward function with time step t is obtained:

$$\ln(r_t) = r(s_t, a_t) \tag{3.18}$$

The accumulative return R:

$$R(s_1, a_1, \ldots, s_T, a_T) = J_T = \frac{1}{T}\sum_{t=1}^{T} \ln(r_t) - \kappa\varphi^2 \ln(r_t) - \frac{\delta}{T-1}\sum_{t=2}^{T} \|a_t - \tilde{a}_t\|_1 \tag{3.19}$$

$$s.t. \sum_{i=1}^{n} = 1 \tag{3.20}$$

In the objective function 3.19, the first term is the returns of the portfolio during period T, the second term is the risk measure of the portfolio returns, and the third term is the one-norm transaction costs term. $\kappa \geq 0$, $\delta \geq 0$ are the risk aversion parameter and transaction costs parameter.

Considering the μ_θ policy, our goal is to maximize the objective function parameterized by θ, which can be written as

$$\mu_{\theta*} = \arg\max_{\mu_\theta} J_T(\mu_\theta) = \arg\max_{\mu_\theta} \left\{ \frac{1}{T}\sum_{t=1}^{T} \ln(r_t) - \kappa\varphi^2 \ln(r_t) - \frac{\delta}{T-1}\sum_{t=2}^{T} \|a_t - \tilde{a}_t\|_1 \right\} \tag{3.21}$$

$$\nabla_\theta \mu_\theta(\tau) = \frac{1}{T}\sum_{t=1}^{T} \nabla_\theta \ln \mu_\theta(a_t, s_t) \tag{3.22}$$

$$\theta \leftarrow \theta + \lambda \nabla_\theta \mu_\theta(\tau) \tag{3.23}$$

where λ is the learning rate. In Eq (3.21), the optimal policy $\mu_{\theta*}$ is learned by the agent, where $\mu_{\theta*}$ conforms to the objective function. The agent optimizes the objective function 3.19, which is constructed through the parameter θ. Due to the denominator T, the equation is properly normalized for data of different lengths T, which also makes it possible to train mini-batch in a sampling period. In order to achieve the expected returns of the portfolio, we adjust the portfolio weight to minimize risk loss $\kappa\varphi^2 \ln(r_t)$ and cost loss $\frac{\delta}{T-1}\sum_{t=2}^{T} \|a_t - \tilde{a}_t\|_1$. In other words, when κ and δ are sufficiently small, the returns of the strategy approach the theoretical optimal for the risk-cost reward.

3.5. DPO

1) Process of DPO. Financial markets consist of financial assets and their related information. It is crucial to optimize portfolios by fully utilizing information such as historical asset prices and factors influencing market changes in the financial market. Therefore, we propose DPO, which applies DL modules and combines MPT with an RL framework to make optimal portfolio decisions. Figure 1 shows our framework.

We use deep neural networks to extract features from financial information, construct a policy network, and optimize a portfolio using DPG to dynamically allocate assets based on market trends. Considering the characteristics of data and the scope of research, we need to study a decision-making process that takes into account the temporal and complexity of capital. Our DL modules include the sequence feature module and correlation feature module. They extract and learn features of asset information, respectively, and then integrate the temporal and correlation features of financial assets into the framework of RL to achieve the optimal portfolio strategy. Additionally, we focus on transaction costs and risk factors in the trading process, emphasizing the importance of traditional

financial theory. We propose a reward function based on MPT, aiming to enhance the stability of DPO framework's performance. A stable reward function prevents the agent from excessive trading, manages risk, and makes optimal portfolio decisions.

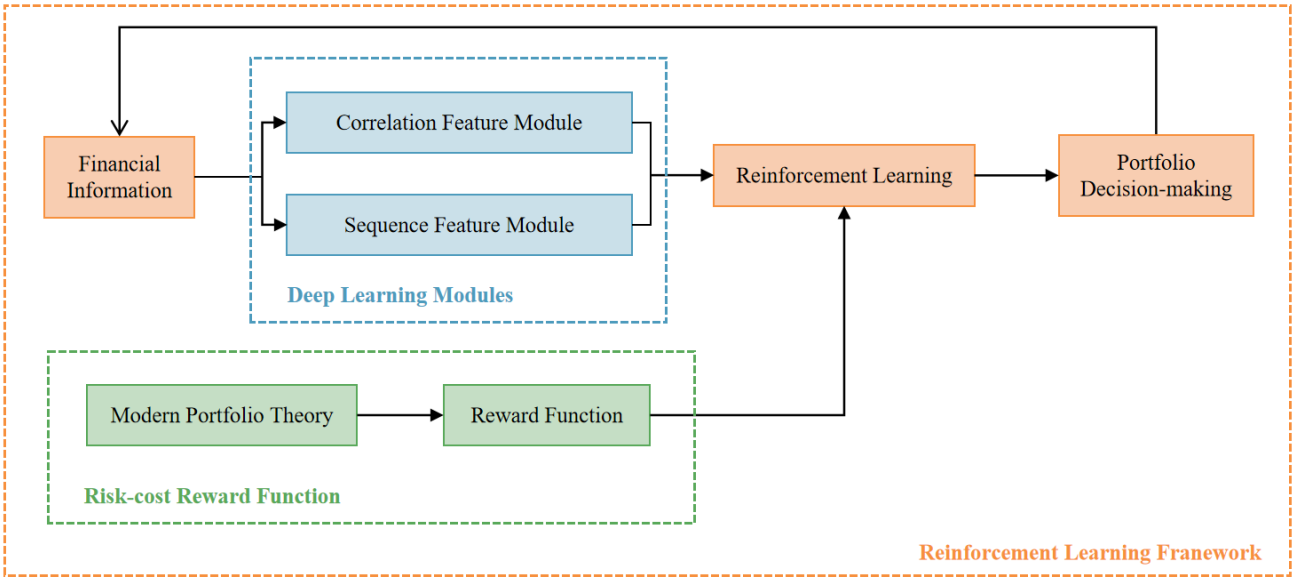

Figure 1. Process of DPO, which applies DL modules and combines MPT with the RL learning paradigm to make optimal portfolio decisions. Financial information includes the assets price information from two real-world markets. DL modules extract and fuse the temporal and asset correlation features of financial information. Reward function combines MPT, considering transaction costs and risk factors during the trading process. The DPG method is used to achieve the optimal portfolio decision-making. The optimization of portfolio is the result of the combined effects of DL modules, the risk-cost reward function, and the RL learning paradigm.

2) Architecture of DPO. In the preceding section, we introduced the DPO framework. In this section, we will provide a detailed description of the architecture of DPO, as shown in Figure 2. DPO combines MPT and utilizes the DL method and DPG learning paradigm to construct the policy network.

To begin, the input is the price vector of financial assets. The price vector consists of the normalized prices of stocks over time T. Next, the correlation feature module and the sequence feature module are used to extract features from the price vector. In the correlation feature module, considering the correlation between stock assets, we utilize CNN with a horizontal kernel to analyze the correlation between each stock at the current time point and adjacent time points, extracting information regarding asset correlation features. To provide more robust information for the current price features, the portfolio weights w_{t-1} from the previous trading period are incorporated into the correlation feature module, and a risk-free asset is introduced into the module to extract asset correlation features. In the sequence feature module, considering the temporal correlation of stock prices, we use LSTM to analyze the sequential correlation of each stock's continuous time series, extracting information regarding time series features. Similarly, the portfolio weights w_{t-1} from the

previous trading period are incorporated into the sequence feature module, and a risk-free asset bias is introduced into the module to extract asset sequence features. In addition, to enhance the effectiveness of information extraction by the DL modules, the element-wise addition approach is used to fuse feature vectors from price time series and asset correlation. Finally, the features from the DL modules are combined using the DPG method, ultimately outputting the optimal portfolio strategy through softmax. By considering transaction costs and risk factors in portfolio decision-making, we can effectively avoid biased trading induced by aggressive trades and profit fluctuations.

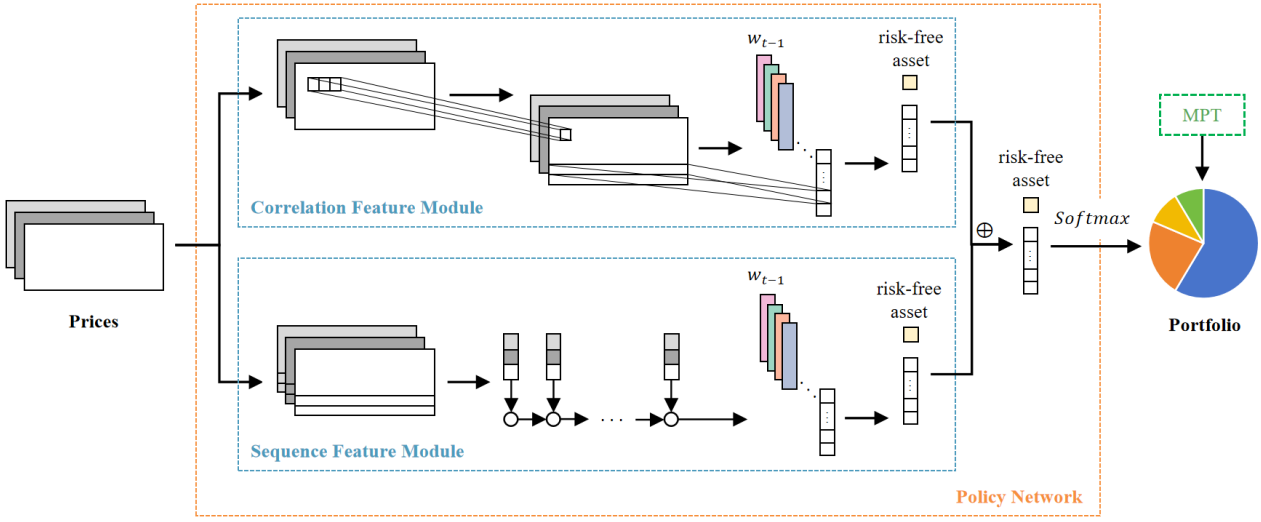

Figure 2. Architecture of DPO, where DL modules consist of the correlation feature module and sequence feature module. $\oplus$ is an element-wise addition approach to fuse the feature information from time series of price and assets correlation. Risk-cost reward function combines with MPT.

4. Results

4.1. Datasets and setup

DPO is a generalizable framework that is not limited to any particular market. To test its profitability and effectiveness, we empirically evaluated the applicability of the DPO framework on two real datasets: the cryptocurrency market and the Chinese A-share market. Additionally, we present the experimental setup.

4.1.1. Cryptocurrency

Cryptocurrencies have a considerable impact on emerging economies and the global economy. We evaluated DPO on a real cryptocurrency dataset. We set Bitcoin as risk-free cash [1,9] and select the 11 cryptocurrencies with the highest monthly trading volume. The cryptocurrency dataset contains prices for 12 different cryptocurrencies from 2015-07-01 to 2017-07-01. For each cryptocurrency, prices are recorded every 30 minutes. The dataset is divided into two sets: a training set and a testing set. We chronologically split 92% of the dataset as the training portion,with the remaining data used for testing.

If there is missing data for any cryptocurrency, the data for each date was filled in using the backfill method. The testing set is from 2017-05-03 12:00:00 to 2017-07-01 00:00:00. Except for cash assets, all assets contain the opening prices, closing prices, highest prices, and lowest prices.

4.1.2. Chinese A-share

The scale and uniqueness of the Chinese stock market make it highly attractive for academic research. We evaluated DPO on a real Chinese A-share dataset. We obtained the trading prices of all A-share stocks listed on Shanghai and Shenzhen stock exchanges every 5 minutes from EastMoney.com [https://www.eastmoney.com], and selected 9 stocks with high trading volume and good liquidity from 2023-06-27 to 2024-03-07. The trading hours of the Chinese stock market are from 9:30 am to 11:30 am and from 1:00 pm to 3:00 pm every Monday to Friday. No trading is allowed on Saturdays, Sundays, and holidays announced by the Shanghai Stock Exchange and Shenzhen Stock Exchange. We collect trading data every 5 minutes, resulting in 48 data points per trading day. For each stock, we record the prices every 30 minutes. Similar to the way the cryptocurrency dataset is processed, we chronologically split 95% of the dataset as the training portion and the remaining data as the testing portion. The testing set is from 2024-02-21 14:05:00 to 2024-03-07 14:35:00. We set the Chinese government bond as the risk-free rate. Except for the risk-free rate, all assets contain four prices.

In addition, due to the characteristics of the Chinese stock market, the temporal continuity of the Chinese A-share dataset is lower than that of the cryptocurrency dataset. Therefore, when conducting portfolio optimization experiments, we preprocess the Chinese A-share dataset by adopting an effective time concatenation method to ensure that the database stores a continuous 5-minute dataset. The advantage of this data processing method is that it ensures the temporal continuity of input data during the training process, thereby avoiding the negative impact of invalid data on the performance of the training model.

4.1.3. Setup

DPO was trained using a PC configured with i7-10700, Geforce GTX 1080Ti 11 GB, 32 GB RAM, 250 GB SSD, and 2 TB HDD. For both of datasets, the number of the total training steps is 8×10^4, batch size is set to 128, and the learning rate is 2.8×10^{-4} with the optimization as Adam.

For the correlation feature module, we set up a convolution operation with only 3 convolution kernels to get 3 feature maps, and the size of each kernel is 1×3. The stride is 1×1 and filled with padding. Then, we utilize 10 convolutional kernels with a size of $1 \times w$ for convolutional calculations to acquire 10 feature maps, each with a size of $1 \times n$, where w represents the width of the feature maps obtained from the previous convolutional operation, and n denotes the number of selected assets. Finally, we concatenate the 10 feature maps obtained at time t with the portfolio weights from the previous time $t-1$, and convolve them using a 1×1 kernel to ultimately obtain a portfolio feature map containing historical information. For the sequence feature module, we set up a multilayer perception network with 10 neurons. Then, we concatenate the weight information from time $t-1$ and perform the same $1 \times w$ convolution operation as in the correlation feature module. The activation function of both modules is ReLU, which is a nonlinear activation function capable of learning and representing more complex functions. It is worth noting that, due to variations in data volume and constraints, we

select the last 8% of the data from the cryptocurrency dataset for testing, and the last 5% of the data from the Chinese A-share dataset for testing.

In addition, we adopt Python as a programming language and implement the proposed DPO using the TensorFlow. Also, the Tflearn library is used to build, train, and evaluate DPO. We use Pandas library to handle and manipulate financial datasets, and utilize the NumPy package for scientific computing. SQLite3 provides a way to work with relational databases directly from Python. The time library module is used for handling time-related tasks.

4.2. Evaluation

We choose three metrics to measure the performance of portfolio optimization strategies: APV, SR, and MDD. APV is an intuitive measure of the cumulative change in asset value during the testing period. SR is the excess return generated by a portfolio for each unit of total risk it bears, which measures the relationship between returns and risk. MDD is the maximum loss and risk resistance that a portfolio may face.

1) APV. Given any portfolio management strategy, the APV of assets is evaluated for profitability considering transaction costs. In the final time range T, it can be expressed as

$$APV = \frac{\upsilon_T}{\upsilon_0} \tag{4.1}$$

APV is an accumulated value used to quantify the return of a strategy over a time period T. Normalize the investment portfolio without sacrificing generality, so that the initial return is $\upsilon_0 = 1$ and υ_T is the portfolio value of time T. Therefore, the higher the value of APV, the better the profitability of the portfolio. APV mainly considers portfolio returns and does not take into account the volatility of these returns.

2) SR. The SR is the average return per unit of volatility or total risk over the risk-free rate. It evaluates the ratio of average return to volatility (standard deviation). It balances the returns and risk of a portfolio, which can be expressed as

$$SR = \frac{\mathbb{E}[R_p - R_f]}{\sqrt{var(R_p - R_f)}} \tag{4.2}$$

where R_p is the portfolio return rate, R_f is the risk-free return rate, and the denominator is the standard deviation of the portfolio excess return rate. In the cryptocurrency market, we set $R_f = 2\%$ as the risk-free rate of return. In the Chinese A-share market, we use the Chinese government bond as the risk-free rate. Therefore, the higher the value of SR, the lower the risk of the portfolio and the better the profitability.

3) MDD. Although SR considers the volatility of portfolio values, it has no difference in market upward and downward volatility. However, the downward movement of the market is usually more important as it can effectively measure the stability of algorithms during market downturns. MDD represents the maximum loss from peak to trough.

$$MDD = \max_{t:\tau>t} \frac{\upsilon_t - \upsilon_\tau}{\upsilon_t} \tag{4.3}$$

where υ_t and υ_τ are the portfolio values of the portfolio at time t and time τ. Therefore, the lower the MDD value, the higher the stability of returns.

We compare several methods to verify the performance of our method. Best (Best) is holding only one of the best stocks during the training period. Anti-Correlation (Anticor) uses the principle of mean regression to continuously transfer wealth from high-performing stocks to anti-correlated underperforming stocks [26]. Weighted moving average mean regression (WMAMR) is a method of learning portfolio optimization using equal-weighted moving averages to explore price correlations in past periods [27]. Robust medical revolution (RMR) strategy is based on improved regression estimator to construct the optimal portfolios [28].

4.3. Experimental results

DPO is evaluated from the following three aspects: First, the profitability of the model on real datasets is assessed. Second, the actual economic performance of the model, considering transaction costs and risk factors, is evaluated. In addition, feature extraction and fusion are crucial in DPO. To verify the effectiveness of module selection and integration within the DL modules, the ablation experiment results of different feature extraction methods are compared.

4.3.1. Evaluation on profitability

We empirically test the performance of DPO on different datasets by comparing different models. Figure 3 shows the APV on the cryptocurrency dataset. Specifically, the results in Figure 3 show that the Best strategy has a high APV in the early stages, but with the change of trading time, the APV value fluctuates significantly and no longer shows a growing trend. This shows that the Best strategy has poor performance in profitability. When investors only hold one stock, it can lead to unstable returns and high risk, so they need to adopt a portfolio strategy. The APV value of Anticor is stable with an upward trend, but the overall level is relatively low, showing poor profitability. The APV values of WMAMR and RMR strategies are higher in the early stage, but show a downward trend in the later stage. This indicates that as the trading time and number of trades increase, when considering transaction costs, the two methods may cause excessive costs to erode portfolio returns due to excessive operations. The DPO method demonstrates good profitability and stability while considering transaction costs constraints. While other methods demonstrate declining trends over time, DPO shows the best performance. The results show the effectiveness and superiority of DPO in portfolio optimization.

In addition, we further validate the profitability of different methods on the Chinese A-share dataset. Figure 4 shows the APV on the Chinese A-share dataset. The results show that when the amount of data in the current period is relatively small, the APV of the Best strategy is significantly higher than that of other strategies. However, as the data volume increases and the trading time changes, the APV of the Best strategy shows a downward trend, which means that the risk resistance ability of the Best strategy is weak. This shows that adopting a portfolio strategy is more likely to achieve higher investment returns. The APV of the Anticor strategy has a low overall return and not a very good performance in portfolio optimization. In other words, we should dynamically adjust the proportion of different assets according to the market environment, information characteristics, and other factors. The APV of the WMAMR strategy shows a downward trend, and as the trading time and frequency increase, the APV value decreases more. This indicates that when considering transaction costs during the trading process, the profitability performance of the WMAMR strategy is poor. The APV value of

the RMR strategy is lower than that of WMAMR, and it also shows a similar trend as the WMAMR strategy in the later stage. At the same time, we have also verified that the WMAMR and RMR are not suitable for the Chinese A-share market and do not have universality. Therefore, it is crucial for portfolio strategies to consider the risk factors and transaction costs during the trading process. The APV of DPO has been steadily increasing. Due to the impact of transaction costs, the APV values of most strategies have decreased, while the profitability performance of DPO still performs well.

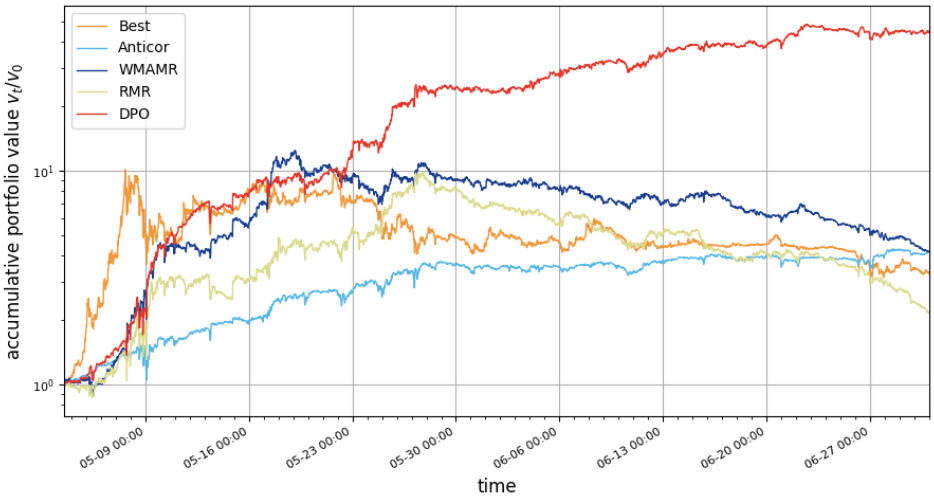

Figure 3. APV on cryptocurrency dataset. The changes in APV over time for different portfolio optimization methods. Best is the baseline strategy. Anticor, WMAMR, and RMR are state of the art strategies. DPO is our strategy.

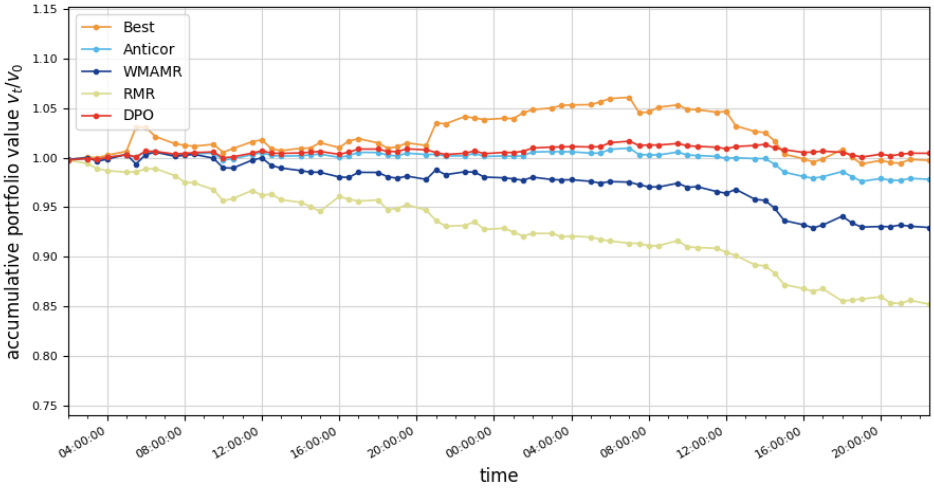

Figure 4. APV on Chinese A-share dataset. The changes in APV over time for different portfolio optimization methods. Best is the baseline strategy. Anticor, WMAMR, and RMR are state of the art strategies. DPO is our strategy.

We evaluate the profitability of DPO by comparing the APV of different models. From Figures 3 and 4, it can be seen that the portfolio returns of different models accumulate over time, with the DPO model having the highest APV. The results show that the performance of the DPO method surpasses that of other methods. DPO performs well on real-world datasets, demonstrating strong generalization capabilities.

Additionally, it is worth noting that while the DPO strategy outperforms other portfolio strategies on the Chinese A-share dataset, the returns on the Chinese A-share dataset are lower than those on the cryptocurrency dataset. We compare the APV of different methods on the two datasets separately, thoroughly validating the reasons for return differences caused by dataset discrepancies, as detailed below: 1) The Chinese A-share dataset has a small amount of data and a short time span, resulting in less wealth accumulation. Due to the trading time characteristics of the Chinese stock market, it is very difficult to collect 5-minute data, and our data is also very valuable and rare. In order to obtain as much data as possible, we collected data repeatedly and continuously over a long period of time, but the amount of data collected is still far less than the amount of cryptocurrency data. Therefore, it is difficult for us to conduct sufficient training, making it challenging to achieve higher returns during testing. 2) Compared to short-term investment tasks, the DPO strategy is more suitable for making portfolio decisions over the medium to long-term. DPO can process and learn market information characteristics, and make the best portfolio decisions by constantly learning from the current environment. When considering the transaction costs during the trading process, DPO will not take excessive tradings and cause economic losses, and can effectively make the trade-off risk and benefits. Therefore, learning more environmental information is more conducive to obtaining higher returns in the medium to long-term investment. DPO is more profitable on the cryptocurrency dataset with longer time spans. Due to the inherent differences in data volume and time span between the two datasets, DPO results vary between them. Therefore, we comprehensively validate the profitability of the DPO strategy in different financial environments, and the experimental results consistently demonstrate the method's excellent universality and representational performance.

4.3.2. Economic evaluation

We conducted experiments on two real datasets, the cryptocurrency market and the Chinese A-share market. We not only need to focus on the predicted returns of portfolios, but also consider actual frictions in the real market, such as transaction costs generated during trading and risk caused by price fluctuations. It is difficult to infer the economic contribution of a model solely from APV. To evaluate the economic performance of the models, we evaluate how the returns of each model are translated into the SR of the portfolio formed based on these returns. We also evaluate the MDD of the portfolio.

In the cryptocurrency market, considering the actual trading situation, the general transaction cost rate is 0.25% [25]. We consider the impact of transaction costs on the expected return of portfolios. Table 1 shows the distribution of metrics for different methods on cryptocurrency datasets. The results in Table 1 indicate that DPO achieves the maximum APV value within the range of transaction cost rates, which further confirms that the DPO method still has significant profitability in the real market considering transaction costs and risk.

Specifically, on the cryptocurrency dataset, Table 1 shows that among all the strategies adopted, the SR and MDD values of the Best strategy showed the worst performance, which is consistent with the analysis in the previous section that we found a large fluctuation in the Best return curve, indicating

that when investors only hold one asset, they bear high risk and unstable returns. Therefore, portfolio optimization is crucial. The SR value of Anticor is relatively high, indicating a strong resistance to risk fluctuations caused by price fluctuations over time. With a lower MDD value, the optimized portfolio is more stable than the Best strategy, but a lower APV value indicates a lower return The SR and MDD values of the WMAMR and RMR strategies show poor performance, indicating that they are unable to respond effectively to market downturns and have poor performance in adjusting for market and downside risks. The DPO method has the highest SR value, the lowest MDD value and the highest APV value. The results indicate that DPO can effectively control transaction costs, perform well in balancing market risk and returns, and has good representational and economic performance.

In the Chinese A-share market, a reasonable estimate of the transaction cost of the Chinese stock market during normal times is 25 basis points [3]. Table 2 shows the distribution of metrics for different methods on Chinese A-share datasets. Table 2 also shows the excellent performance of DPO considering risk factors and controlling transaction costs.

Table 1. Metrics of different methods on cryptocurrency dataset.

Strategies	APV	SR	MDD
Best	3.2994	0.0292	0.6871
Anticor	4.1815	0.0431	0.3134
WMAMR	4.1981	0.0341	0.6656
RMR	2.1710	0.0235	0.7810
DPO	44.3851	0.0750	0.3332

Table 2. Metrics of different methods on Chinese A-share dataset.

Strategies	APV	SR	MDD
Best	0.9973	–0.0031	0.0633
Anticor	0.9782	–0.1139	0.0332
WMAMR	0.9295	–0.2353	0.0759
RMR	0.8522	–0.4599	0.1457
DPO	1.0044	0.0311	0.0157

On the Chinese A-share dataset, Table 2 shows that the SR value of the Best strategy is higher than that of other strategies, indicating that the return of the Best strategy has little fluctuation in the short-term. However, compared to other strategies, Best has a higher MDD value, indicating that this strategy has a relatively weak ability to control downside risk. The performance of the SR and MDD values for the Anticor strategy is average, indicating relatively weak economic performance and unstable returns. However, the SR values for the WMAMR and RMR strategies are both negative, indicating that these two strategies have a weak ability to control risk. At the same time, the MDD of WMAMR and RMR strategies are higher than those of other strategies, and their economic performance in the Chinese market is inferior to that of other strategies. This is generally consistent with the trend of the return curves shown in the previous section. Therefore, we should adopt different portfolio strategies based on different market environment information to adapt to changes in different environments. The SR value of the DPO strategy is higher than that of other portfolio strategies, indicating that in the face of market fluctuations, the DPO strategy can effectively control risk and enable portfolio returns to

accumulate stably and continuously. The MDD value of the DPO strategy is the smallest, indicating that when facing a market downturn, it can be dynamically adjusted in a timely manner according to the market environment, reflecting the strong economic performance of the DPO strategy.

From Tables 1 and 2, we can see that compared with other strategies, the SR value of the DPO method is significantly higher than that of other strategies. Therefore, we believe that DPO can better balance between returns and risk, and obtain higher returns while controlling transaction costs. Tables 1 and 2 also reports the MDD distribution of each algorithm, and the results show that DPO has a very low MDD when making portfolio selection. The smaller the MDD of the portfolio, the lower the risk of the portfolio, which means that constraining the volatility of returns helps to control downside risk.

Additionally, it is noteworthy that the MDD of different portfolio optimization strategies is lower in the Chinese A-share dataset compared to the cryptocurrency dataset. We compared the MDD of different strategies across the two datasets, fully validating the reasons for the variations in metrics due to differences in the datasets, as detailed below: First, the real market environments of different datasets are distinct, and their data characteristics differ. Therefore, when measuring with metrics, results should be compared within the same dataset. Thus, we compare MDD within the same dataset to identify portfolio optimization strategies that can more robustly make the trade-off between returns and risk compared to others. Second, due to the characteristics of the Chinese market, the Chinese A-share dataset has fewer data points, a shorter time span, and lower price volatility. Given the unique trading hours of the Chinese stock market, collecting 5-minute data is particularly challenging, and our data is extremely precious and rare. To gather as much data as possible, we conducted multiple consecutive collections over a long period. However, the volume of collected data remains much smaller than that of the cryptocurrency dataset, and the time span of the dataset is also significantly shorter. Therefore, the short-term price data exhibits low volatility, and the limited data volume makes it difficult to achieve sufficient training. Nonetheless, testing on different real-world datasets holds significant importance. Due to the inherent differences in data volume and time span between the two datasets, the DPO results differ across the two datasets. However, the results consistently demonstrate the excellent effectiveness, applicability, and generalizability of DPO in real market environments.

4.3.3. Evaluation of ablation results

To evaluate the importance and usefulness of module selection and fusion in the DL modules, we separately use CNN and LSTM methods to extract features from financial information and conduct ablation experiments. By observing changes in model performance, we can determine that different feature extraction modules are crucial for the effectiveness of portfolio decision-making. This approach helps in understanding the internal workings of the model. The experimental results fully validate the effectiveness and applicability of different feature extraction modules for portfolio optimization. In addition, MPT has limitations that may prevent it from fully capturing the complexity and realities of real financial markets. However, MPT provides a foundational framework for portfolio optimization. DPO combines the essence of MPT with the advantages of DL methods. Our ablation experiments validate the effectiveness of combining MPT with different DL methods and RL learning paradigm for portfolio optimization, while also leveraging the advantages of RL learning paradigm within the MPT framework.

We compare DPO with extraction methods that use only one module. The only difference between these variants is the method of feature extraction. DPO-L uses only the sequence feature module,

employing LSTM to extract the temporal features of financial information. DPO-C uses only the correlation feature module, employing CNN to extract the correlation features of financial assets. DPO uses both the sequence feature module and the correlation feature module, simultaneously considering the temporal correlation of market information and the correlation of assets, combining these information features into the RL framework to make optimal portfolio decisions.

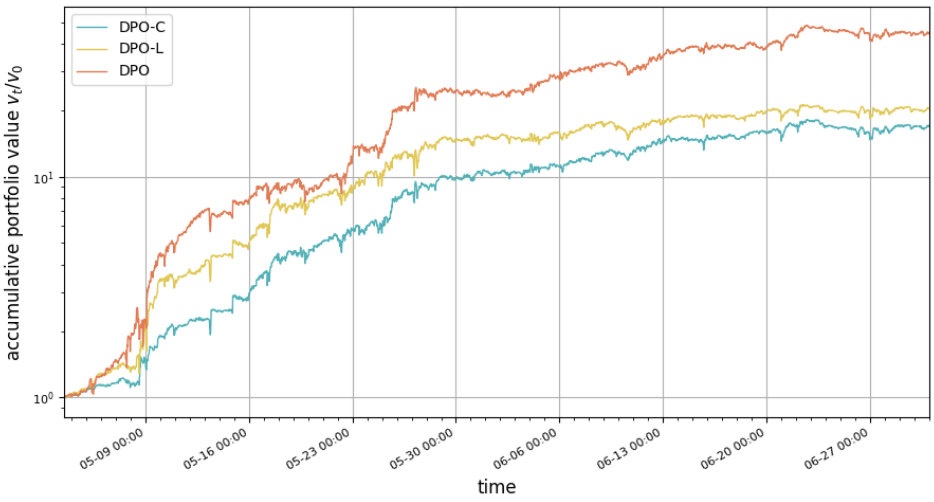

Figure 5. APV of ablation experiments on cryptocurrency dataset. The changes in APV over time for different ablation experiment methods. DPO-C uses only the correlation feature module. DPO-L uses only the sequence feature module. DPO uses both the sequence feature module and the correlation feature module.

We validated through ablation experiments that module selection and fusion in the DL module enable the formulation of more effective portfolio strategies and result in better profitability. Figure 5 shows the returns from the ablation experiments in the cryptocurrency dataset. It illustrates the APV for different feature extraction methods on the cryptocurrency dataset. As shown in Figure 5, the APV of DPO is the highest. The results show that extracting and integrating features from both the sequence feature module and the correlation feature module significantly enhance the profitability of DPO in the cryptocurrency market. Similarly, Figure 6 shows the returns from the ablation experiments in the Chinese A-share dataset. It illustrates the APV for different feature extraction methods on the Chinese A-share dataset. As shown in Figure 6, the APV of DPO consistently exceeds that of DPO-L and DPO-C, indicating the best profitability and stability. The results demonstrate that simultaneously considering the temporal and correlation features of financial assets in portfolio optimization significantly improves profitability, highlighting the importance of module selection and fusion.

First, we evaluate the profitability of DPO with different feature extraction modules. Upon observing the results in Figures 5 and 6, we find that the APV of DPO-L is higher than that of DPO-C. This shows that financial price sequence features extracted based on LSTM are more effective in improving the profitability of portfolio compared to the financial asset correlation features extracted by CNN. Furthermore, when we combine both the sequence feature module and the

correlation feature module into the DPO framework, the APV is the highest in both the cryptocurrency market and the Chinese A-share market. This result demonstrates that appropriate selection and fusion of DL modules in portfolio optimization can significantly enhance the performance of portfolio decisions.

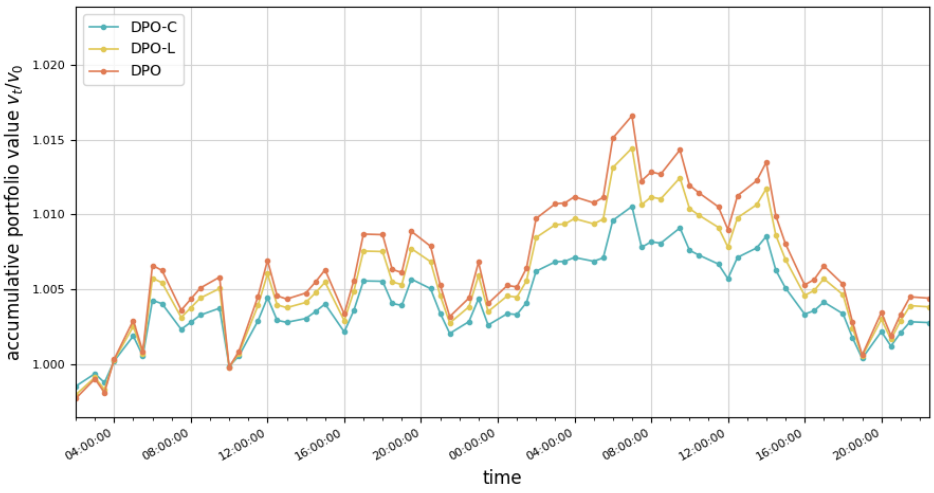

Figure 6. APV of ablation experiments on Chinese A-share dataset. The changes in APV over time for different ablation experiment methods. DPO-C uses only the correlation feature module. DPO-L uses only the sequence feature module. DPO uses both the sequence feature module and the correlation feature module.

Second, we evaluate the economic performance of DPO with different feature extraction modules. Table 3 presents the metrics for the ablation experiments on the cryptocurrency dataset, while Table 4 presents the metrics for the ablation experiments on the Chinese A-share dataset. The results in Tables 3 and 4 show that on real-world datasets, the APV values of DPO, DPO-L, and DPO-C are higher than those of other strategies. This shows the necessity of employing feature extraction and fusion module methods. In financial markets, focusing on the temporal features of asset information and asset correlation is beneficial for improving portfolio return predictions.

Table 3. Metrics for ablation experiments on cryptocurrency dataset.

Strategies	APV	SR	MDD
DPO-C	16.9725	0.0722	0.1901
DPO-L	20.4285	0.0643	0.2633
DPO	44.3851	0.0750	0.3332

In addition, Table 3 displays the SR and MDD of different feature extraction modules on the cryptocurrency dataset. The highest SR is observed for DPO, indicating that DPO effectively balances between returns and risk by extracting and processing features from price time series and asset correlation. DPO achieves maximum portfolio returns while minimizing risk. Table 4 presents the SR and MDD of different feature extraction methods on the Chinese A-share dataset. This demonstrates

that the DPO strategy can extract features from different market price information, effectively control risk caused by price fluctuations, and exhibit good generalization.

Table 4. Metrics for ablation experiments on Chinese A-share dataset.

Strategies	APV	SR	MDD
DPO-C	1.0028	0.0302	0.0100
DPO-L	1.0038	0.0308	0.0137
DPO	1.0044	0.0311	0.0157

5. Conclusions

The proposed DPO, which combines MPT with DL methods and the RL learning paradigm, effectively makes the trade-off between portfolio returns and risk across different financial markets. Specifically, DPO extracts and fuses time series features and asset related features of financial market information by applying DL modules. In addition, a novel risk-cost reward function is proposed by considering the transaction costs and risk factors of portfolios. The results show that DPO can achieve maximum predictive returns while controlling transaction costs and risk. Compared with other baseline strategies, DPO achieves more competitive profitability and better economic performance. It makes timely decisions based on the features and trends of financial market information, dynamically reconfigures the portfolio, and achieves portfolio optimization. We not only elucidate more effective and universal portfolio optimization methods based on RL and MPT but also highlight a new direction of applying more advanced DL methods to quantitative financial research.

In future studies, we will focus on current limitations and further research. Despite the challenges in collecting 5-minute data due to the unique trading hours of the Chinese stock market, our data is extremely valuable and rare. To achieve better returns accumulation and economic performance, we will accumulate and collect data over a longer period. In addition, further work will explore the influence of multi-source information, such as price series information, financial market text information, and assisted question-answering information for prompt-guided AI models (for example, GPT, which is Generative Pre-trained Transformer and is a DL model developed by OpenAI for natural language processing tasks), on portfolio decision-making to achieve more stable performance of the algorithm. We will explore integrating different novel DL modules to extract information from various types of sources and fuse multi-source information features. This is aimed at enhancing the versatility and generalization of the model.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work is founded by Ministry of Science and Technology Talent Exchange foundation of China (No. DL2021179008L).

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. Z. Jiang, D. Xu, J. Liang, A deep reinforcement learning framework for the financial Portfolio management problem, preprint, arXiv: 1706.10059.
2. S. Gu, B. Kelly, D. Xiu, Empirical asset pricing via machine learning, *Rev. Financ. Stud.*, **33** (2020), 2223–2273. https://doi.org/10.1093/rfs/hhaa009
3. M. Leippold, Q. Wang, W. Zhou, Machine learning in the Chinese stock market, *J. Financ. Econ.*, **145** (2022), 64–82. https://doi.org/10.1016/j.jfineco.2021.08.017
4. H. Markowitz, Portfolio selection, *J. Financ.*, **7** (1952), 71–91.
5. A. V. Olivares-Nadal, V. DeMiguel, Technical note—A robust perspective on transaction costs in portfolio optimization, *Oper. Res.*, **66** (2018), 733–739. https://doi.org/10.1287/opre.2017.1699
6. C. H. Hsieh, On asymptotic log-optimal portfolio optimization, *Automatica*, **151** (2023), 110901. https://doi.org/10.1016/j.automatica.2023.110901
7. R. Kan, X. Wang, G. Zhou, Optimal portfolio choice with estimation risk: No risk-free asset case, *Manage. Sci.*, **68** (2022), 2047–2068. https://doi.org/10.1287/mnsc.2021.3989
8. S. Almahdi, S. Y. Yang, An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown, *Expert Syst. Appl.*, **87** (2017), 267–279. https://doi.org/10.1016/j.eswa.2017.06.023
9. Y. Zhang, P. Zhao, Q. Wu, B. Li, J. Huang, M. Tan, Cost-sensitive portfolio selection via deep reinforcement learning, *IEEE T. Knowl. Data Eng.*, **34** (2020), 236–248. https://doi.org/10.1109/TKDE.2020.2979700
10. W. F. Sharpe, Capital asset prices: A theory of market equilibrium under conditions of risk, *J. Financ.*, **19** (1964), 425–442. https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1964.tb02865.x
11. B. V. de M. Mendes, R. C. Lavrado, Implementing and testing the maximum drawdown at risk, *Financ. Res. Lett.*, **22** (2017), 95–100. https://doi.org/10.1016/j.frl.2017.06.001
12. E. F. Fama, Efficient capital markets: A review of theory and empirical work, *J. Financ.*, **25** (1970),383–417. https://jstor.66557.net/stable/pdf/2325486
13. J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Networks*, **61** (2015), 85–117. https://doi.org/10.1016/j.neunet.2014.09.003
14. T. Fischer, C. Krauss, Deep learning with long short-term memory networks for financial market predictions, *Eur. J. Oper. Res.*, **270** (2018), 654–669. https://doi.org/ 10.1016/j.ejor.2017.11.054
15. A. Vidal, W. Kristjanpoller, Gold volatility prediction using a CNN-LSTM approach, *Expert Syst. Appl.*, **157** (2020), 113481. https://doi.org/10.1016/j.eswa.2020.113481
16. Y. Ma, R. Han, W. Wang, Portfolio optimization with return prediction using deep learning and machine learning, *Expert Syst. Appl.*, **165** (2021), 113973. https://doi.org/10.1016/j.eswa.2020.113973

17. T. H. Nguyen, K. Shirai, J. Velcin, Sentiment analysis on social media for stock movement prediction, *Expert Syst. Appl.*, **42** (2015), 9603–9611. https://doi.org/10.1016/j.eswa.2015.07.052
18. L. Chen, H. Zhang, J. Xiao, X. He, S. Pu, S. F. Chang, Counterfactual critic multi-agent training for scene graph generation, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (ICCV), (2019), 4613–4623. https://doi.org/10.1109/ICCV.2019.00471
19. J. Jang, N. Seong, Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory, *Expert Syst. Appl.*, **218** (2023), 119556. https://doi.org/10.1016/j.eswa.2023.119556
20. P. Beraldi, A. Violi, M. Ferrara, C. Ciancio, B. A. Pansera, Dealing with complex transaction costs in portfolio management, *Ann. Oper. Res.*, **299** (2021), 7–22. https://doi.org/10.1007/s10479-019-03210-5
21. Y. Ma, Z. Li, Robust portfolio choice with limited attention, *Electron. Res. Arch.*, **31** (2023), 3666–3687. https://doi.org/10.3934/era.2023186
22. M. García-Galicia, A. A. Carsteanu, J. B. Clempner, Continuous-time reinforcement learning approach for portfolio management with time penalization, *Expert Syst. Appl.*, **129** (2019), 27–36. https://doi.org/10.1016/j.eswa.2019.03.055
23. H. M. Markowitz, Foundations of portfolio theory, *J. Financ.*, **46** (1991), 469–477. https://www.jstor.org/stable/2328831
24. G. Y. Ban, N. El Karoui, A. E. Lim, Machine learning and portfolio optimization, *Manage. Sci.*, **64** (2018), 1136–1154. https://doi.org/10.1287/mnsc.2016.2644
25. Y. Ye, H. Pei, B. Wang, P. Y. Chen, Y. Zhu, J. Xiao, et al., Reinforcement-learning based portfolio management with augmented asset movement prediction states, in *Proceedings of the AAAI Conference on Artificial Intelligence*, (AAAI), (2020), 1112–1119. https://doi.org/10.1609/aaai.v34i01.5462
26. A. Borodin, R. El-Yaniv, V. Gogan, Can we learn to beat the best stock, *J. Artif. Intell. Res.*, (NeurIPS), **21** (2004), 579–594.
27. L. Gao, W. Zhang, Weighted moving average passive aggressive algorithm for online portfolio selection, in *2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, IEEE (IHMSC), (2013), 327–330. https://doi.org/10.1109/IHMSC.2013.84
28. D. Huang, J. Zhou, B. Li, S. C. Hoi, S. Zhou, Robust median reversion strategy for online portfolio selection, *IEEE T. Knowl. Data Eng.*, **28** (2016), 2480–2493. https://doi.org/10.1109/TKDE.2016.2563433

AIMS Press