Omega 129 (2024) 103169

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Robust online portfolio optimization with cash flows ☆

Benmeng Lyu [a,b], Boqian Wu [d], Sini Guo [c,*], Jia-Wen Gu [b], Wai-Ki Ching [a]

[a] Department of Mathematics, The University of Hong Kong, 999077, Hong Kong, China
[b] Department of Mathematics, Southern University of Science and Technology, Shenzhen, 518055, China
[c] School of Management, Beijing Institute of Technology, Beijing, 100081, China
[d] Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, 7522 NB, The Netherlands

ARTICLE INFO

Keywords:
Decision making
Cash flow
Linear programming
Transaction costs
Robust optimization

ABSTRACT

One fundamental issue in finance is portfolio selection, which seeks the best strategy for assigning capital among a group of assets. There has been growing interest in online portfolio selection where the investment strategy is frequently readjusted in a short time as new financial market data arrives constantly. Numerous effective algorithms have been extensively examined both in terms of theoretical analysis and empirical evaluation. Previous online portfolio selection algorithms that incorporate transaction costs are limited by the fact that they often approximate the transaction remainder factor instead of calculating it precisely. This could lead to suboptimal investment performance. To address this issue, we present an innovative method that considers transaction costs and resolves the accurate transaction remainder factor and the optimal portfolio allocation simultaneously for each period. In addition, we take into account the open-end fund, which permits constant cash inflows, and develop a framework for online portfolio selection. We also incorporate the uncertainty set to minimize the impact of the prediction error during the prediction process. Utilizing the framework presented in this innovative model, we develop a novel algorithm for online portfolio selection that incorporates transaction costs and continuous cash inflows with the objective of maximizing cumulative wealth. Numerical experiments show that the proposed algorithms are able to handle transaction costs and constant cash inflows effectively.

1. Introduction

Portfolio selection has been extensively researched in the financial field, which aims to achieve certain objectives like long-term cumulative return maximization and risk minimization. Two well-known portfolio selection theories have emerged. In mean–variance theory, the return of the risky asset is typically considered to be governed by a probability distribution. Portfolio managers make decisions by balancing the portfolio allocation's future expected return and the corresponding risk in a specific time period [1]. The Kelly investment theory, as proposed in [2], offers an alternative perspective to addressing the portfolio selection problem. The objective is to maximize the future expected logarithmic return within multiple trading periods by sequentially distributing capital among a group of assets. More research works on portfolio selection can be explored in [3–5].

Due to the inherent sequential structure of portfolio selection tasks, researchers have put forth a trading technique known as the "Online Portfolio Selection (OLPS)" algorithm. Different from traditional mean–variance theory, estimating the distribution function of future asset returns in advance is not necessary for online portfolio selection. It uses forecasting approaches to predict the future performance of assets and strategically distributes capital to these assets in order to maximize long-term investment returns [6]. Based on the historical information of investment strategies and risky asset prices, the investor will adjust a new portfolio allocation at the beginning of each period. In the past few years, an increasing number of researchers have developed various types of online portfolio selection algorithms. A detailed survey of OLPS can be found in [7].

The fundamental concept behind these algorithms, regardless of their specific formulations, is to implicitly or explicitly predict the risky assets' future performance and optimize the allocation of the portfolio in each period. A variety of algorithms are intricately linked to the framework. The first category of the algorithms is commonly known as "Benchmark", which encompasses the Buy-and-Hold (BAH) strategy [6], the Best Stock (Best) strategy [6] and the Constant Rebalanced Portfolios (CRP) strategy [6]. These strategies are frequently employed

☆ Area: Data-Driven Analytics. This manuscript was processed by Associate Editor Xiang Fang.

* Corresponding author.
E-mail addresses: benmeng@connect.hku.hk (B. Lyu), b.wu@utwente.nl (B. Wu), guosini@bit.edu.cn (S. Guo), gujw@sustech.edu.cn (J.-W. Gu), wching@hku.hk (W.-K. Ching).

https://doi.org/10.1016/j.omega.2024.103169
Received 24 January 2024; Accepted 29 July 2024
Available online 2 August 2024

as baselines for comparative analysis. The second classification of online portfolio selection algorithms is "Follow the winner", which is based on the momentum principle assuming that assets currently exhibiting favorable performance are likely to continue performing well in the future and aims to enhance the allocation of capital towards these more successful assets. For example, the Universal Portfolios (UP) algorithm [8], the Exponential Gradient (EG) algorithm [9] and the Online Newton Step (ONS) algorithm [10]. Conversely, algorithms known as "Follow the loser" are based on the mean reversion principle, which is often characterized by transferring the wealth from good-performing assets (winners) to poor-performing assets (losers). This type of algorithms includes the Anticorrelation (Anticor) algorithm [11], the Confidence Weighted Mean Reversion (CWMR) algorithm [12], the Passive Aggressive Mean Reversion (PAMR) algorithm [13], the Online Moving Average Reversion (OLMAR) algorithm [14] and the Robust Median Reversion (RMR) algorithm [15]. Apart from the two types of Follow-the-winner/loser algorithms, another type, "Pattern Matching" considers both winners and losers at the same time. In general, the pattern matching algorithms can be divided into two distinct procedures: sample selection and portfolio optimization [7]. In the first procedure, sample selection, we divide the risky assets into different groups based on their historical price patterns and predict the future prices group by group. In portfolio optimization, we derive the optimal portfolio allocation based on the previously predicted results. The category of online portfolio selection algorithms referred to as "Meta-learning" is the fifth one [16]. This technique establishes many foundational experts, each of which is equipped with distinct algorithms and generates a singular portfolio vector [17]. The output vectors of all the experts are then combined to form the final portfolio.

Despite the success of the aforementioned online portfolio selection algorithms in either theoretical or empirical contexts, most of them fail to consider transaction costs. Transaction costs in real trading refer to the expenses associated with buying or selling risky assets. These costs can significantly impact investment returns and are incurred by investors when executing trades. This oversight inevitably results in suboptimal performance when making decisions on portfolio selection. Transaction costs normally consist of price impact costs and commission costs. However, the price impact costs are related to the security's market micro-structure, which is beyond our scope. As the analysis of online portfolio selection algorithms has become increasingly important, there has been a growing focus placed on the significance of taking transaction costs into consideration. Blum and Kalai [18] firstly computed a target portfolio with the UP algorithm and then paid for the transaction costs proportionally from each stock. Albeverio et al. [19] presented a new online universal strategy that takes transaction costs into account and employs a new prediction method for price relative vectors based on "cross rates". The Online Lazy Updates (OLU) algorithm proposed in [20] aims to incorporate transaction costs into the online portfolio by measuring the difference between two successive portfolios and using lazy updates. The authors of the study in [21] incorporated the transaction costs in the net profit in each period. Jiang et al. [22] proposed an iterative formula for estimating the value of the transaction remainder factor w_{t-1}, which is the net proportion of the initial wealth that remains after subtracting transaction costs incurred at the beginning of period t. Li et al. [23] introduced the Transaction Cost Optimization (TCO) algorithm. They investigated the relationship between the transaction remainder factor w_{t-1} and two successive portfolios' variations and utilized the convex optimization to derive two closed-form formulas for updating the portfolio. Guo et al. [24] derived the analytical formula of the transaction remainder factor, and provided theoretical evidence of its existence and uniqueness. Although some of the above online portfolio selection studies try to solve the transaction remainder factor in real online decision making, only approximated solution with lower and upper bounds is obtained, or an exact algorithm is provided to calculate the transaction remainder factor when the investment strategy is already known, see for example, [23,24]. Currently, there is still no effective method of precisely solving the transaction remainder factor and investment strategy simultaneously.

In addition, none of the algorithms has ever considered the case of cash inflows and outflows, which is a novel and promising research direction. In the financial market, according to their investment structure, mutual funds are typically categorized into two groups: closed-end funds or open-end funds. For the purpose of raising capital for its initial investments, the closed-end fund will conduct a single initial public offering (IPO) in which it will issue a specific amount of shares. However, no new shares will be generated, and no extra capital will be added to the fund [25]. In contrast, open-end funds accept a constant flow of new investment capital, which is the most common type of mutual fund investment [26]. In the past, online portfolio selection algorithms solely took into account closed-end funds. After the initial investment is made, there will be no additional cash coming in or going out over the entire investment process. For the first time, we provide a novel online portfolio selection model that is grounded in the utilization of open-end funds. Here, we assume that the open-end fund has constant cash inflows during a given investment period.

Optimizing investment strategy with precise transaction costs and real-world dynamic capital is a critical and thorny issue in online portfolio selection. This study aims to tackle this issue by contributing to the literature in the following ways:

- The investment strategy and transaction remainder factor are solved precisely and simultaneously by the change of variables approach and linear programming, which overcomes the limitation that the approximated transaction remainder factor fails to ensure the true optimal strategy.
- A novel online portfolio selection framework incorporating constant cash flows is constructed, strengthening the practicability of investment strategy in real-world portfolio management, especially for open-end funds.
- The robust online portfolio selection algorithm is proposed based on price relative vector uncertainty sets, which is beneficial to lessening the impact of the prediction errors on investment decision making and improving the portfolio resilience.

This work is organized as follows: Section 2 presents the basic setting and mathematical formulation of online portfolio selection with transaction costs and constant cash inflows. Section 3 defines the optimization problem and introduces a new framework for online portfolio selection. Section 4 demonstrates the effectiveness and practicality of the proposed algorithm through numerical experiments. Finally, Section 5 summarizes the findings and suggests future research directions.

2. Problem formulation

In OLPS, a portfolio manager reallocates his capital in a sequential manner to achieve some financial objectives [14]. In particular, at the beginning of each period, a new portfolio allocation is determined according to the prediction of risky assets' performances in the upcoming period. Subsequently, the manager adjusts the existing allocation to align with the newly determined allocation. After the rebalancing, the portfolio's wealth will then be subject to market changes until the end of the period. It is noted that the portfolio allocation changes dynamically, with an increase in the allocation of outperforming assets and a decrease in the allocation of underperforming assets. During the rebalance, we make the assumption that the transaction costs are incurred when the weights of risky assets are changed for each period. Additionally, a fixed amount of cash is invested at the end of each period.

In the analysis, we examine a portfolio selection task spanning n periods. The market consists of $(m + 1)$ assets, encompassing one cash asset and m risky assets [27]. Incorporating a cash asset in the portfolio contributes to diversified investment to effectively reduce risk. The closing price vector $\mathbf{p}_t \in \mathbb{R}_+^{m+1}$ represents the prices of all assets during the tth period. Each element of this vector $p_{t,i}$, corresponds to the closing price of the asset i. The price changes for period t are represented by the *price relative vector* $\mathbf{x}_t \in \mathbb{R}_+^{m+1}$, where $x_{t,i} = \frac{p_{t,i}}{p_{t-1,i}}$ indicates the ratio of the price change of asset i from period $t-1$ to period t. For instance, $x_{t,i} = 1.2$ means that the price of asset i has increased by 20% from the previous period. It is important to note that the cash asset is assumed to have no price change in all the periods, indicating that the corresponding ratio is $x_{t,0} = 1$ for each period.

The allocation over the portfolio at period t is specified by the *portfolio vector* $\mathbf{b}_t = (b_{t,0}, b_{t,1}, \ldots, b_{t,m})$. The element $b_{t,i}$ is the weight of total capital in the ith risky asset at period t, while the symbol $b_{t,0}$ represents the percentage of total capital in the cash asset. Generally, it is assumed that short selling is not allowed. And then $\mathbf{b}_t \in \Delta_m$, where

$$\Delta_m = \left\{ \mathbf{b}_t : \mathbf{b}_t \in \mathbb{R}_+^{m+1}, \quad \sum_{i=0}^{m} b_{t,i} = 1 \right\}.$$

The investing process is identified by a portfolio selection strategy that is characterized by adhering to a series of mappings, that is

$$\mathbf{b}_t : \mathbb{R}_+^{(m+1)\times(t-1)} \to \Delta_m, \quad t = 2, 3, \ldots, n,$$

where $\mathbf{b}_t$ is the tth portfolio vector. The portfolio typically begins with a uniform allocation, i.e., $\mathbf{b}_1 = (\frac{1}{m+1}, \ldots, \frac{1}{m+1})$.

2.1. Transaction costs and constant cash inflows

The portfolio manager establishes a new portfolio $\mathbf{b}_t$ at the beginning of the tth period. The previous allocation $\bar{\mathbf{b}}_{t-1}$ at the end of the $(t-1)$th period is rebalanced to arrive at the new portfolio $\mathbf{b}_t$. It should be noted that, due to price changes during the $(t-1)$th period, the allocation $\bar{\mathbf{b}}_{t-1}$ is different from $\mathbf{b}_{t-1}$, where

$$\bar{b}_{t-1,i} = \frac{b_{t-1,i}x_{t-1,i}}{\mathbf{b}_{t-1}\mathbf{x}_{t-1}^\top}, \quad i = 0, 1, \ldots, m.$$

Since the investment of the cash inflow K, the allocation shifts from $\bar{\mathbf{b}}_{t-1}$ to

$$\hat{\mathbf{b}}_{t-1} = \frac{S_{t-1}\bar{\mathbf{b}}_{t-1} + K\mathbf{e}}{S_{t-1} + K} \tag{1}$$

where the cumulative wealth at the end of the $(t-1)$th period is denoted by S_{t-1}, and the unit vector $\mathbf{e} = (1, 0, \ldots, 0)$ states a $(m+1)$-dimensional vector where the elements $e_0 = 1, e_i = 0, i = 1, \ldots, m$. The amount of capital in the portfolio becomes $S_{t-1} + K$.

In Fig. 1, the market movement during the $(t-1)$th period derives the cumulative wealth and the portfolio vector to S_{t-1} and $\bar{\mathbf{b}}_{t-1}$. The cash inflow K redistributes the portfolio vector into $\hat{\mathbf{b}}_{t-1}$ with corresponding wealth $S_{t-1} + K$. The portfolio vector rebalances to $\mathbf{b}_t$ as a result of the asset selling and buying activity, and the portfolio wealth shrinks by a factor w_{t-1}.

Let γ_s and γ_b denote the transaction cost rates for selling and buying assets, respectively. In this work, we assume that transaction costs for buying and selling risky assets are proportional. The transaction remainder factor w_{t-1} is the net proportion of the wealth that remains after the transaction costs are deducted [22,23], which is defined by

$$1 = w_{t-1} + \gamma_b \sum_{i=1}^{m} (b_{t,i}w_{t-1} - \hat{b}_{t-1,i})^+ + \gamma_s \sum_{i=1}^{m} (\hat{b}_{t-1,i} - b_{t,i}w_{t-1})^+,$$

where $(\cdot)^+ = \max\{\cdot, 0\}$.

It is crucial to emphasize that there is no transaction cost incurred when trading the cash asset. The sales take place when the proportion that was present prior to the rebalancing is higher than the proportion that was present thereafter, i.e. $\hat{b}_{t-1,i} - b_{t,i}w_{t-1} > 0$. On the other hand, the purchases occur when $b_{t,i}w_{t-1} - \hat{b}_{t-1,i} \geq 0$. In accordance with the established conventions of previous research [19,20,28,29], we set that $\gamma_s = \gamma_b = \gamma \in [0, 1]$. We have

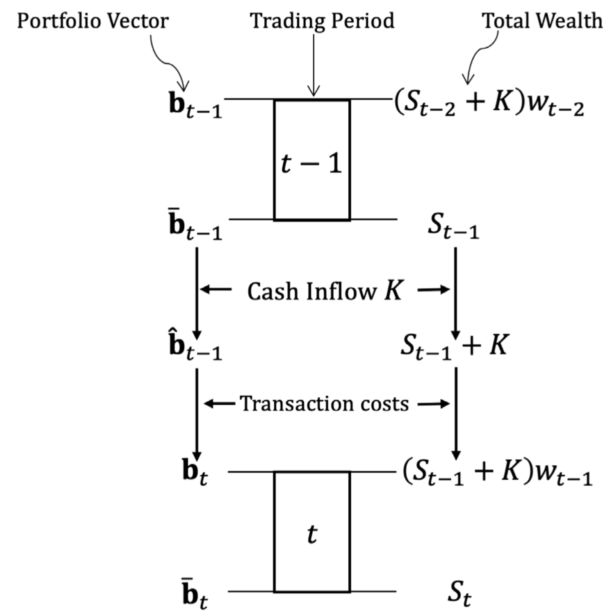

Fig. 1. Illustration of the effect of constant cash inflows and transaction costs.

$$1 = w_{t-1} + \gamma \sum_{i=1}^{m} |b_{t,i}w_{t-1} - \hat{b}_{t-1,i}|. \tag{2}$$

This indicates that the total value of the transaction remainder factor and the transaction expenses are always equal to one. Through the process of rebalancing, the cumulative wealth is changed into $(S_{t-1} + K) \times w_{t-1}$.

During the period t, the allocation $\mathbf{b}_t$ causes a change of the cumulative wealth with a factor of

$$\mathbf{b}_t\mathbf{x}_t^\top = \sum_{i=0}^{m} b_{t,i}x_{t,i}.$$

In summary, the wealth changes from $S_{t-1}+K$ to $(S_{t-1}+K)\times w_{t-1}\times(\mathbf{b}_t\mathbf{x}_t^\top)$ during the period t. Owing to the fact that we invest sequentially and work with relative prices, the cumulative wealth increases in a multiplicative manner. The following expression can be used to specify the cumulative wealth at the end of period n:

$$S_n = S_0 \prod_{t=1}^{n} w_{t-1} \times (\mathbf{b}_t\mathbf{x}_t^\top) + K \sum_{k=2}^{n} \prod_{t=k}^{n} w_{t-1} \times (\mathbf{b}_t\mathbf{x}_t^\top), \tag{3}$$

where the initial wealth S_0 is set to 1.

The objective of this task is for the manager to develop a portfolio strategy $(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n)$ that is based on the price relatives $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, with the goal of maximizing the cumulative wealth S_n. For each period t, the manager accesses the previous portfolio vectors $(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{t-1})$ and price relative vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{t-1})$. He/She then utilizes historical data to establish a new portfolio vector $\mathbf{b}_t$ for the subsequent price relative vector $\mathbf{x}_t$. This process is iterated until the last period, at which stage the portfolio strategy is evaluated in accordance with the cumulative wealth S_n.

Several general assumptions are used in the preceding model. To begin with, we make the assumption that each unit of the asset can be divided into smaller parts without any restrictions and that any desired amount of shares, including fractional amounts, can be purchased or sold at the most recent closing price during any trading period. Furthermore, we make the assumption that the actions of the market and the values of stocks remain unaffected by any trading method [20].

Algorithm 1 Online Portfolio Selection with Transaction Costs and Constant Cash Inflows

Input: An OLPS algorithm; Historical price relatives $\mathbf{x}_1^t$; Transaction cost rate γ; Constant cash inflows K.
Output: Final cumulative wealth S_n.
Begin
Initialize $\mathbf{b}_1 = (\frac{1}{m+1}, \dots, \frac{1}{m+1}), S_0 = 1$.
for t=2, …, n **do**
1. Invest the cash inflow K;
2. Record current allocation $\hat{\mathbf{b}}_{t-1} = \frac{S_t \bar{\mathbf{b}}_{t-1} + K\mathbf{c}}{S_t + K}$;
3. Calculate next portfolio vector $\mathbf{b}_t$ by **Algorithm 2**;
4. Derive the transaction reminder factor w_{t-1} by **Algorithm 2**;
5. Update the cumulative wealth after transaction cost is deducted: $S' = (S_{t-1} + K) \times w_{t-1}$;
6. Receive market price relatives vector $\mathbf{x}_t$;
7. Update the cumulative wealth S_t:

$S_t = (S_{t-1} + K) \times w_{t-1} \times (\mathbf{b}_t \mathbf{x}_t^\top)$;

8. Record the allocation at the end of period t : $\bar{b}_{t-1,i} = \frac{b_{t,i} x_{t,i}}{\mathbf{b}_t \mathbf{x}_t^\top}, i = 0, 1, \dots, m$;

end for
End

3. Robust online portfolio optimization

In previous algorithms such as OLU [20], TCO [23], the objective was to minimize $||\hat{\mathbf{b}}_{t-1} - \mathbf{b}_t||_1$ to incorporate transaction costs, which is equivalent to maximizing the transaction remainder factor w_{t-1} or estimate the value of w_{t-1} using an iterative formula. This was based on the relationship between the transaction remainder factor and transaction costs in Eq. (2). We propose a novel method that maximizes the expected value function while precisely solving the transaction remainder factor.

The nonlinear term $|b_{t,i} w_{t-1} - \hat{b}_{t-1,i}|$ is difficult to tackle in portfolio optimization, hence, we use the change of variables approach to transform it into a linear one that is easier to solve [30]. Assume that there exist $u_{t,i}$ and $v_{t,i}$ satisfying

$$\begin{cases} |b_{t,i} w_{t-1} - \hat{b}_{t-1,i}| = u_{t,i} + v_{t,i} \\ b_{t,i} w_{t-1} - \hat{b}_{t-1,i} = u_{t,i} - v_{t,i} \qquad i = 0, 1, \dots, m \\ u_{t,i} \geq 0, v_{t,i} \geq 0. \end{cases} \tag{4}$$

In case that $b_{t,i} w_{t-1} - \hat{b}_{t-1,i} \geq 0$, we have $u_{t,i} \geq 0, v_{t,i} = 0$; otherwise, $u_{t,i} = 0, v_{t,i} > 0$. Therefore, for any $b_{t,i} w_{t-1} \geq 0, \hat{b}_{t-1,i} \geq 0$, there always exists corresponding solutions of $u_{t,i}$ and $v_{t,i}$ that fall into three different cases:
(1) $u_{t,i} > 0, v_{t,i} = 0$;
(2) $u_{t,i} = 0, v_{t,i} = 0$;
(3) $u_{t,i} = 0, v_{t,i} > 0$.

Then we can transform Eq. (2) into the following equation:

$$w_{t-1} = 1 - \gamma \sum_{i=1}^{m} (u_{t,i} + v_{t,i}) = 1 - \gamma (\mathbf{u}_t + \mathbf{v}_t)\mathbf{1} + \gamma (u_{t,0} + v_{t,0}). \tag{5}$$

And the portfolio vector $\mathbf{b}_t$ is expressed as follows:

$$\mathbf{b}_t = \frac{\hat{\mathbf{b}}_{t-1} + \mathbf{u}_t - \mathbf{v}_t}{w_{t-1}} = \frac{\hat{\mathbf{b}}_{t-1} + \mathbf{u}_t - \mathbf{v}_t}{1 - \gamma (\mathbf{u}_t + \mathbf{v}_t)\mathbf{1} + \gamma (u_{t,0} + v_{t,0})}, \tag{6}$$

where $\mathbf{1}$ is the $(m+1)$-dimensional column vector of all ones.

3.1. Price relative vector prediction

In order to implement the investment strategies, we predict the price relative vector $\tilde{\mathbf{x}}_t$ for period t. When it comes to the future trend of asset values, having a higher forecast accuracy means that we are able to have a better understanding of the future trend. Following this, one can make adjustments to the investment portfolio in accordance with the prediction results and allocate more wealth to assets whose prices may rise. In order to make a prediction of $\tilde{\mathbf{x}}_t$, a variety of methods have been suggested [9,13–15]. Here we employ two popular methods: Robust Median Reversion (RMR) [15] and Exponential Moving Average (EMA) [7].

RMR: We calculate the L_1-median of historical prices in order to make a prediction about $\tilde{\mathbf{x}}_t$. In statistics, the L_1-median $\boldsymbol{\mu}$, which is often referred to as spatial median, is the solution to the following problem for a k-historical price window:

$$\boldsymbol{\mu} = \underset{\boldsymbol{\mu}}{\operatorname{argmin}} \sum_{i=1}^{k} \left\| \mathbf{p}_{t-i} - \boldsymbol{\mu} \right\|.$$

The next price is predicted using the robust L_1-median estimator $\boldsymbol{\mu}$ in RMR. Then, the predicted value of the price relative vector is

$$\tilde{\mathbf{x}}_t(k) = \frac{\tilde{\mathbf{p}}_t}{\mathbf{p}_{t-1}} = \frac{\boldsymbol{\mu}}{\mathbf{p}_{t-1}}.$$

EMA: Different from RMR, where only a window of historical data is taken into consideration, the exponential moving average (EMA) is proposed by taking the exponentially weighted average of all historical prices.

$$\begin{aligned} EMA_1(\alpha) &= \mathbf{p}_1, \\ EMA_t(\alpha) &= \alpha \mathbf{p}_t + (1-\alpha) EMA_{t-1}(\alpha) \\ &= \alpha \mathbf{p}_t + (1-\alpha)\alpha \mathbf{p}_{t-1} + (1-\alpha)^2 \alpha \mathbf{p}_{t-2} + \cdots + (1-\alpha)^{t-1} \mathbf{p}_1, \end{aligned}$$

where $\alpha \in (0, 1)$ is the decaying factor. Then, the predicted value of the price relative vector can be updated by

$$\begin{aligned} \tilde{\mathbf{x}}_t(\alpha) &= \frac{EMA_{t-1}(\alpha)}{\mathbf{p}_{t-1}} = \frac{\alpha \mathbf{p}_{t-1} + (1-\alpha) EMA_{t-2}(\alpha)}{\mathbf{p}_{t-1}} \\ &= \alpha \mathbf{1} + (1-\alpha) \frac{EMA_{t-2}(\alpha)}{\mathbf{p}_{t-2}} \frac{\mathbf{p}_{t-2}}{\mathbf{p}_{t-1}} = \alpha \mathbf{1} + (1-\alpha) \frac{\tilde{\mathbf{x}}_{t-1}}{\mathbf{x}_{t-1}}. \end{aligned}$$

Appendix A compared the prediction accuracy of two different methods. It should be noted that the estimation errors of both methods differ a lot on different stocks. The presence of a significant estimation error, which indicates a lack of precision in predicting price relative, can have a major impact on the optimal allocation of a portfolio. Hence, it is important to integrate the uncertainty regarding the accuracy of predictions into the portfolio optimization procedure. Robust optimization is a method that effectively captures and models uncertainty [31]. Compared with the traditional portfolio optimization approaches, the theoretical framework of robust optimization suggests that the inputs are subject to errors, hence falling inside an uncertainty set surrounding their nominal prediction [32]. The proposed approach expands upon traditional portfolio optimization models by integrating uncertainty through a rigorous methodology within the modeling framework.

There are several ways illustrated in [33] to model the uncertainty. The box uncertainty set is a polytope where each dimension of the uncertain coefficients is bounded within a specific range. Ellipsoidal uncertainty sets incorporate second moment information, such as the variance and correlation of uncertain parameters. In our model, the uncertainty set $\mathcal{U}$ is designed to capture the range of potential deviations between the predicted price relative vectors and true vectors. Following the guidance of robust optimization frameworks [33], we use box constraints to define this set, providing a balance between computational efficiency and comprehensive coverage of uncertainties. One straightforward approach is to implement protection such that the predicted price relative $\tilde{x}_{t,i}$ is not far away from their true value $x_{t,i}$. Then, the uncertainty set is defined as follows:

$$\mathcal{U} = \left\{ \mathbf{x}_t \mid |x_{t,i} - \tilde{x}_{t,i}| \leq \varepsilon_{t,i}, \quad i = 0, \dots, m \right\} \tag{7}$$

where $\varepsilon_{t,i}$ represents the upper bound for the acceptable difference between the predicted and actual price relative. This formulation ensures that all possible realizations of the price relative within the interval $[\tilde{x}_{t,i} - \varepsilon_{t,i}, \tilde{x}_{t,i} + \varepsilon_{t,i}]$ are considered during the optimization process. Box uncertainty sets are advantageous due to their simplicity and tractability, as highlighted by [33]. The parameter $\varepsilon_{t,i}$ can be dynamically adjusted based on historical prediction errors, calculated as follows:

$$\varepsilon_{t,i} = \begin{cases} 0 & t = 2, \\ \frac{1}{t-2}\sum_{s=2}^{t-1} |x_{s,i} - \tilde{x}_{s,i}| & t = 3, \ldots, n, \end{cases}$$

where $i = 0, 1, \ldots, m$. This method calculates $\varepsilon_{t,i}$ as the average prediction error over past periods to reflect historical volatility. By recalibrating $\varepsilon_{t,i}$ at each period, we ensure that the model remains sensitive to recent market environment while maintaining robustness. Note that the uncertainty set $\mathcal{U} \subset \mathbb{R}_+^{m+1}$, which means $\tilde{x}_{t,i} - \varepsilon_{t,i} \geq 0$.

3.2. Mathematical formulation

We now formulate the framework for online portfolio selection with transaction costs and constant cash inflows. This framework can be used to develop online portfolio selection algorithms based on various trading principles. It is straightforward to derive the expression of the portfolio's cumulative wealth in Eq. (3) by combining transaction costs and constant cash inflows. The objective of this task is to maximize the cumulative wealth S_n, which is equivalent to the maximization of the portfolio's expected return in each period.

In our study, we add a regularization term in our objective function, which is a technique commonly used to control the stability [34]. By penalizing large changes in portfolio weights, the L_1-penalty term effectively controls the transaction volume to avoid high transaction costs of frequent trading.

Objective: As for period t:

$$\mathbf{b}_t = \arg\max_{\mathbf{b}\in\Delta_m} \ w_{t-1} \times (\mathbf{b}\tilde{\mathbf{x}}_t^\top) - \lambda \|\mathbf{b}w_{t-1} - \hat{\mathbf{b}}_{t-1}\|_1. \tag{8}$$

The first term is the expected return, and the second term is the L_1-penalty term for period t. Here $\lambda \geq 0$ is used to balance the expected return and L_1-penalty term. It is known as the trade-off parameter. We remark that there is no specific risk term in the objective, which is usually incorporated in traditional portfolio selection studies. This is due to online portfolio selection is essentially different from traditional portfolio selection. Most of traditional portfolio selection studies require the statistical assumption on the uncertain return of asset, which is usually assumed to follow a certain distribution function. As a comparison, online portfolio selection focuses on the timely investment decision when market data is renewed constantly and does not require any statistical assumption. The key lies in precisely predicting the future returns and making decisions in a very short time based on the latest market data, and readjusting the strategy when new data arrives. In this relatively high-frequency strategy readjusting setting, it is difficult to precisely measure the short-time investment risk. Therefore, most of online portfolio selection studies do not consider risk term in the model, see, for example, [8–11,13,14,21,23,35].

By using the box uncertainty set in Eq. (7), we can formulate the robust model of the online portfolio selection problem.

Objective: As for period t:

$$\mathbf{b}_t = \arg\max_{\mathbf{b}\in\Delta_m} \ w_{t-1} \times (\mathbf{b}\tilde{\mathbf{x}}_t^\top) - \lambda \|\mathbf{b}w_{t-1} - \hat{\mathbf{b}}_{t-1}\|_1 - w_{t-1} \times (\mathbf{b}\boldsymbol{\varepsilon}_t^\top). \tag{9}$$

In the worst case, the expected return for asset i is $\tilde{x}_{t,i} - \varepsilon_{t,i}$, which is greater than 0 according to Section 3.1. The objective in Problem (9) is to maximize the expected portfolio return in the worst case, where assets with a larger estimation error are penalized and will have smaller weights [36]. Our robust optimization model inherently accounts for risk by considering the worst-case scenarios within the defined uncertainty set. By optimizing the portfolio for the worst-case predicted returns, we ensure that the portfolio is resilient to prediction errors and market uncertainties. This approach provides a safeguard against adverse market conditions and manages risk implicitly.

By using Eqs. (4)–(6), Problem (9) can be transformed into the following problem:

$$\begin{aligned} (\mathbf{u}_t, \mathbf{v}_t) = \arg\max_{\mathbf{u},\mathbf{v}\in\mathbb{R}^{m+1}} \ & (\hat{\mathbf{b}}_{t-1} + \mathbf{u} - \mathbf{v})(\tilde{\mathbf{x}}_t - \boldsymbol{\varepsilon}_t)^\top - \lambda(\mathbf{u}+\mathbf{v})\mathbf{1} \\ \text{s.t.} \quad & (\mathbf{u}-\mathbf{v})\mathbf{1} + \gamma(\mathbf{u}+\mathbf{v})\mathbf{1} - \gamma(u_0 + v_0) = 0, \\ & \hat{\mathbf{b}}_{t-1} + \mathbf{u} - \mathbf{v} \geq \mathbf{0}, \\ & \mathbf{u} \geq \mathbf{0}, \mathbf{v} \geq \mathbf{0} \end{aligned} \tag{10}$$

where the constraint condition

$$(\mathbf{u}-\mathbf{v})\mathbf{1} + \gamma(\mathbf{u}+\mathbf{v})\mathbf{1} - \gamma(u_{t,0} + v_{t,0}) = 0$$

is a transformation of $\sum_{i=0}^m b_i = 1$.

Theorem 1. *Problem (9) achieves the optimal solution $\mathbf{b}_t^\star$ if and only if Problem (10) achieves the optimal solution $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$, where $\mathbf{b}_t^\star$ and $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ satisfies Eq. (4).*

The proof is given in Appendix B. Note that Problem (9) is the robust version of Problem (8) by adding the uncertainty term $w_{t-1} \times (\mathbf{b}\boldsymbol{\varepsilon}_t^\top)$. But it is difficult to solve Problem (9) directly since the transaction remainder factor w_{t-1} is an implicit function of decision vector $\mathbf{b}_t$ (see Eq. (2)). Therefore, we turn to solve Problem (10), which is clearly a standard linear programming model with linear objective and constraints. Theorem 1 guarantees the equivalence of Problem (9) and Problem (10). When the optimal solution $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ of Problem (10) is solved, the optimal solution $\mathbf{b}_t^\star$ of Problem (9) and corresponding transaction remainder factor w_{t-1} can be obtained immediately according to Eq. (4). The transformation from nonlinear programming into a linear programming makes Problem (9) computationally tractable due to large-scale linear programming can be solved easily by using solvers such as Matlab, Python, Gurobi and Cplex.

Algorithm 2 Transaction Costs and Constant Cash Inflows

Input: Last adjusted portfolio vector $\hat{\mathbf{b}}_{t-1}$; Historical price relatives $\mathbf{x}_1^{t-1}$; Window size k; Trade-off parameter λ.
Output: Next portfolio $\mathbf{b}_t$ and transaction remainder factor w_{t-1}.
Begin
1. Predict next price relative vector $\tilde{\mathbf{x}}_t$.
2. Derive $\mathbf{u}_t, \mathbf{v}_t$ by solving the optimization problem (10):

$$\begin{aligned} (\mathbf{u}_t, \mathbf{v}_t) = \arg\max_{\mathbf{u},\mathbf{v}\in\mathbb{R}^{m+1}} & (\hat{\mathbf{b}}_{t-1} + \mathbf{u} - \mathbf{v})(\tilde{\mathbf{x}}_t - \boldsymbol{\varepsilon}_t)^\top - \lambda(\mathbf{u}+\mathbf{v})\mathbf{1} \\ \text{s.t.} \quad & (\mathbf{u}-\mathbf{v})\mathbf{1} + \gamma(\mathbf{u}+\mathbf{v})\mathbf{1} - \gamma(u_0 + v_0) = 0, \\ & \hat{\mathbf{b}}_{t-1} + \mathbf{u} - \mathbf{v} \geq \mathbf{0}, \\ & \mathbf{u} \geq \mathbf{0}, \mathbf{v} \geq \mathbf{0}; \end{aligned}$$

3. Calculate transaction remainder factor w_t:

$$w_{t-1} = 1 - \gamma(\mathbf{u}_t + \mathbf{v}_t)\mathbf{1} + \gamma(u_{t,0} + v_{t,0});$$

4. Update portfolio $\mathbf{b}_t$:

$$\mathbf{b}_t = \frac{\hat{\mathbf{b}}_{t-1} + \mathbf{u}_t - \mathbf{v}_t}{1 - (\mathbf{u}_t + \mathbf{v}_t)\mathbf{1} + u_{t,0} + v_{t,0}};$$

End

Based on Theorem 1, we propose a general algorithm named Transaction Costs and Constant Cash Inflows (TCI), which is displayed in **Algorithm 2**. Then two online portfolio selection algorithms can be designed according to the selection of prediction methods in step 1 of TCI. When the RMR method is selected, we denote this algorithm as TCIR. In case that EMA method is selected, we denote this algorithm as TCIE.

4. Numerical experiments

The purpose of this section is to illustrate the superiority of our algorithm to existing online portfolio selection algorithms through numerical experiments. The numerical experiments are designed from the following aspects:

Table 1
Details of four public data sets.

Dataset	Type	Assets number	Range	Trading days	Region
NYSE-O	Stock	36	1962.03.07-1984.12.31	5651	US
NYSE-N	Stock	23	1985.01.01-2010.06.30	6431	US
TSE	Stock	88	1994.01.04-1998.12.31	1259	Canada
MSCI	Index	24	2006.04.01-2010.03.31	1043	Global

All related datasets are available at https://github.com/OLPS/OLPS.

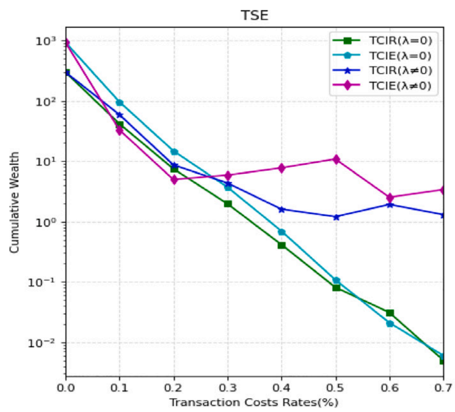

Fig. 2. Cumulative wealth and average turnover with different transaction cost rates.

- **The Effectiveness of L_1-regularization**: Evaluate the impact of L_1-regularization on the performance of our proposed TCIE and TCIR in Section 4.2.
- **Transaction Costs Analysis**: Analyze how transaction costs affect the algorithms' performance in Section 4.3.
- **Constant Cash Inflows Analysis**: Investigate the effect of constant cash inflows on algorithms' performance in Section 4.4.
- **Parameter Sensitivity**: Conduct sensitivity analysis to explore how variation in trade off parameter affects the performance and robustness of our proposed algorithms in Section 4.5.

To quantitatively analyze the performance of the algorithms, the employed performance metrics include cumulative wealth, Sharpe ratio, Information ratio, and turnover.

4.1. Data sets

We make use of four public data sets in the numerical experiments. These include the NYSE-O and its subsequent data set, the NYSE-N, the TSE, and the MSCI in that order [7]. The details of four public data sets are described in Table 1. These publicly accessible datasets guarantee the ability to replicate results and ensure that our evaluation of existing algorithms is impartial.

4.2. The effectiveness of L_1-regularization

To verify the effectiveness of the L_1-penalty term, we employ the data set TSE and analyze the performance of our algorithms. When the trade off parameter $\lambda = 0$, there is no L_1-penalty term in the objective function. We treat the case $\lambda = 0$ as a comparison and set $\lambda = 5 * \gamma$ in Problem (10). The detailed parameter analysis of λ can be found in Section 4.5. In particular, we set the window size to $k = 5$ in RMR and the decaying factor to $\alpha = 0.5$ in EMA.

To measure the performance of our algorithms, we employ two performance metrics: Cumulative wealth and average daily turnover. Cumulative wealth refers to the total value obtained through the successive accumulation of returns, commencing with an initial investment of one dollar. A greater cumulative wealth indicates that an algorithm attains a larger net profit subsequent to the deduction of transaction costs. Portfolio turnover quantifies the rate at which assets are bought and sold within a portfolio. The metric is calculated by taking the average of the absolute differences in portfolio weights at each rebalancing interval [37]. The average daily turnover measures the changes in the portfolios over each period, which is defined as follows:

$$AT_n = \frac{1}{2n} \sum_{t=1}^{n} \left\| \hat{\mathbf{b}}_{t-1} - \mathbf{b}_t \omega_{t-1} \right\|_1 . \tag{11}$$

Lower turnover indicates a more effective portfolio allocation strategy in the case of transaction costs [38].

Using our algorithms, the cumulative wealth and average daily turnover achieved on TSE are illustrated in Fig. 2. One can see that when $\lambda = 0$, the cumulative wealth achieved by TCIR and TCIE decreases rapidly as the transaction cost rate rises. As a comparison, with the existence of the L_1-penalty term ($\lambda \neq 0$), these two algorithms perform significantly better, of which the cumulative wealth decreases slowly and suffers fewer losses due to the increase in transaction cost rate. It can also be seen that the average daily turnover decreases rapidly in TCIR ($\lambda \neq 0$) and TCIE ($\lambda \neq 0$). By incorporating the L_1-penalty term to control the transaction volumes and average daily turnover, the performance of TCIR and TCIE is significantly improved even under high transaction costs.

4.3. Transaction costs analysis

Transaction costs play an important role in our model and have a great impact on the cumulative wealth. In the numerical experiments, we primarily employ the two TCI algorithms, i.e., TCIR and TCIE. As a comparison, we choose 13 other algorithms including BAH [6], Best [6], BCRP [6], UCRP [6], UP [8], EG [9], ONS [10], Anticor [11], CWMR [12], PAMR [13], OLMAR [14], RMR [15] and TCO [23]. The default parameter values specified in the reference papers are used for other algorithms.

A summary of these algorithms is presented in Table 2 in terms of algorithm categories, prediction methods, indicating whether transaction cost and cash flows are considered. As for the method of incorporating transaction cost in these algorithms, we follow the work of Li and Hoi [7] where transaction cost is considered in evaluating the algorithm performance but not directly used in decision process. The newly introduced cash flow in each period is equally distributed among each asset to maintain the algorithms' structures.

Table 2
Summary of algorithms in numerical experiments.

Categories	Algorithms	Prediction Method	Optimizing Transaction Cost	Incorporating Cash Inflows
Benchmarks	BAH [6]	No	No	No
	Best [6]	No	No	No
	BCRP [6]	$\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}_i, i = 1, \ldots, n$	No	No
	UCRP [6]	No	No	No
Follow-the-Winner	UP [8]	No	No	No
	EG [9]	$\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}_{t-1}$	No	No
	ONS [10]	$\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}_i, i = 1, \ldots, t-1$	No	No
Follow-the-Loser	Anticor [11]	No	No	No
	PAMR [13]	$\tilde{\mathbf{x}}_t = 1/\tilde{\mathbf{x}}_{t-1}$	No	No
	CWMR [12]	$\tilde{\mathbf{x}}_t = 1/\tilde{\mathbf{x}}_{t-1}$	No	No
	OLMAR [14]	$\tilde{\mathbf{x}}_t(w) = \frac{1}{w}\left(\mathbf{p}_{t-1}/\mathbf{p}_{t-1} + \mathbf{p}_{t-2}/\mathbf{p}_{t-1} + \cdots + \mathbf{p}_{t-w}/\mathbf{p}_{t-1}\right)$	No	No
	RMR [15]	$\tilde{\mathbf{x}}_t(k) = \boldsymbol{\mu}/\mathbf{p}_{t-1}$	No	No
	TCO [23]	$\tilde{\mathbf{x}}_t = 1/\tilde{\mathbf{x}}_{t-1}$	Yes	No

Table 3
Cumulative wealth achieved by various algorithms with transaction cost rates (0%, 0.25%, and 0.5%) and zero constant cash inflow.

Algorithms	NYSE-O			NYSE-N			TSE			MSCI		
	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%
BAH	14.21	14.17	14.14	18.23	18.18	18.14	1.60	1.59	1.58	0.90	0.89	0.89
Best	0.39	0.14	0.05	9.91	4.27	1.96	0.30	0.22	0.16	0.42	0.24	0.13
BCRP	248.50	180.25	130.74	120.76	99.54	82.04	6.59	6.32	6.06	1.49	1.48	**1.47**
UCRP	26.67	22.66	19.26	31.82	26.34	21.80	1.57	1.49	1.43	0.92	0.89	0.87
UP	26.75	22.66	19.41	31.35	26.13	21.82	1.56	1.49	1.42	0.92	0.90	0.88
EG	26.70	22.81	19.50	31.26	26.14	21.86	1.57	1.50	1.43	0.92	0.90	0.88
ONS	25.15	23.76	22.44	27.86	25.68	23.67	0.38	0.36	0.35	0.88	0.84	0.80
Anticor	9.98E + 07	1295.00	0.02	1.72E + 06	29.99	0.00	22.55	1.81	0.14	4.21	0.80	0.15
CWMR	5.72E + 15	2.86E + 05	0.00	9.18E + 05	0.00	0.00	186.55	1.79	0.02	**15.56**	0.19	0.00
PAMR	5.08E + 15	2.60E + 05	0.00	1.30E + 06	0.00	0.00	257.86	2.09	0.02	14.99	0.18	0.00
OLMAR	7.66E + 16	4.89E + 08	2.85	4.21E + 08	1.70	0.00	59.00	1.10	0.02	14.56	0.53	0.02
RMR	1.77E + 17	6.13E + 08	1.92	6.24E + 08	1.30	0.00	245.17	3.65	0.05	15.26	0.49	0.02
TCO	6.76E + 12	6.70E + 08	2.79E + 04	2.43E + 07	4.21E + 03	56.83	153.05	**9.57**	0.91	5.33	1.54	0.75
TCIR	7.10E + 16	**2.68E + 11**	**1.47E + 07**	**1.27E + 09**	6.21E + 04	**865.64**	299.13	4.05	1.20	10.37	1.08	1.20
TCIE	**8.41E + 18**	1.87E + 11	9.62E + 04	1.13E + 09	**2.50E + 05**	488.36	**932.39**	5.45	**10.85**	14.91	**1.88**	0.94

Table 3 presents the cumulative wealth achieved by different algorithms under different transaction cost rates $0\%, 0.25\%$, and 0.5%, and zero cash inflow. Several interesting results can be derived following from Table 1. Firstly, in case the transaction cost rate is 0, TCIR achieves the largest cumulative wealth on NYSE-N and the second largest cumulative wealth on TSE. TCIE attains the largest cumulative wealth on both NYSE-O and TSE. The superior performance of TCIE over TCIR can be ascribed to the higher prediction accuracy of the price relative vector since accurately predicting the future returns of risky assets is beneficial to averting the potential investment loss and increasing the final return.

In general, TCIE and TCIR perform much better than other online portfolio selection algorithms in the case of zero transaction cost rates. Secondly, under the case of nonzero transaction cost rate, the performance of "Benchmark" strategies (i.e., BAH, Best, BCRP, UCRP) and "Follow the winner" algorithms (i.e., UP, EG, ONS) degrades slowly and suffers fewer losses. As a comparison, "Follow the loser" algorithms (i.e., Anticor, CWMR, PAMR, OLMAR, RMR) and TCO suffer significant losses due to the increase in transaction cost rate. Lastly, with a transaction cost rate of 0.25%, TCIR is able to reach the largest cumulative wealth on NYSE-O and the second-largest wealth on NYSE-N. It also achieves the best performance on NYSE-O and NYSE-N with a transaction cost rate of 0.5%. TCIE achieves the largest cumulative wealth on NYSE-N and MSCI with a transaction cost rate of 0.25%. Therefore, our algorithms (i.e., TCIR, TCIE) outperform many existing OLPS algorithms in cumulative wealth no matter whether the transaction cost rate is 0 or not.

Fig. 3 shows the trend of cumulative wealth on NYSE-O, NYSE-N, TSE, and MSCI with 0.25% transaction cost rate and zero cash inflow. It is clear that TCIR and TCIE consistently surpass other algorithms on the NYSE-N data set and achieve the largest and the second largest cumulative wealth on NYSE-O and NYSE-N. TCIE and TCIR perform worse than some algorithms on TSE and MSCI. TCIE only exceeds other algorithms in the later period of the data set MSCI and achieves the largest cumulative wealth.

Investors attribute significance to transaction costs due to their pivotal role in influencing cumulative returns. Transaction costs have a detrimental impact on returns, and over a period of time, substantial transaction costs can result in the loss of thousands of dollars. This loss is not only due to the direct expenses incurred but also because these charges deplete the amount of cash that can be allocated for investment purposes [39]. To better demonstrate the effectiveness of TCIR and TCIE for transaction costs, we implemented several transaction cost rates, spanning from 0% to 0.7%, and analyzed the relationship between cumulative wealth and transaction costs. For a comparison, we choose six OLPS algorithms with outstanding performance in cumulative wealth. The results are shown in Fig. 4.

It is clear that most of the cumulative wealth decreases as the transaction cost rate increases. There are some special cases in our algorithms where the cumulative wealth increases as the transaction cost is larger (i.e., the cumulative wealth of TCIE on data set TSE). The main reason is that the L_1-penalty term improves the performance by limiting transaction volumes under high transaction cost rates. Regarding the performance of our algorithms, the cumulative wealth achieved by TCIR and TCIE on TSE is not as good as we expected, with low transaction cost rates (less than 0.4%). When transaction cost rates are high (greater than 0.4%), TCIE achieves the largest or the second-largest cumulative wealth on TSE. The performance of TCIR and TCIE is significantly better on NYSE-O, NYSE-N, and MSCI. They achieve the largest or the second largest cumulative wealth for all the different transaction cost rates and outperform all previous algorithms on MSCI in most cases except when the transaction cost rates are 0% and 0.1%.

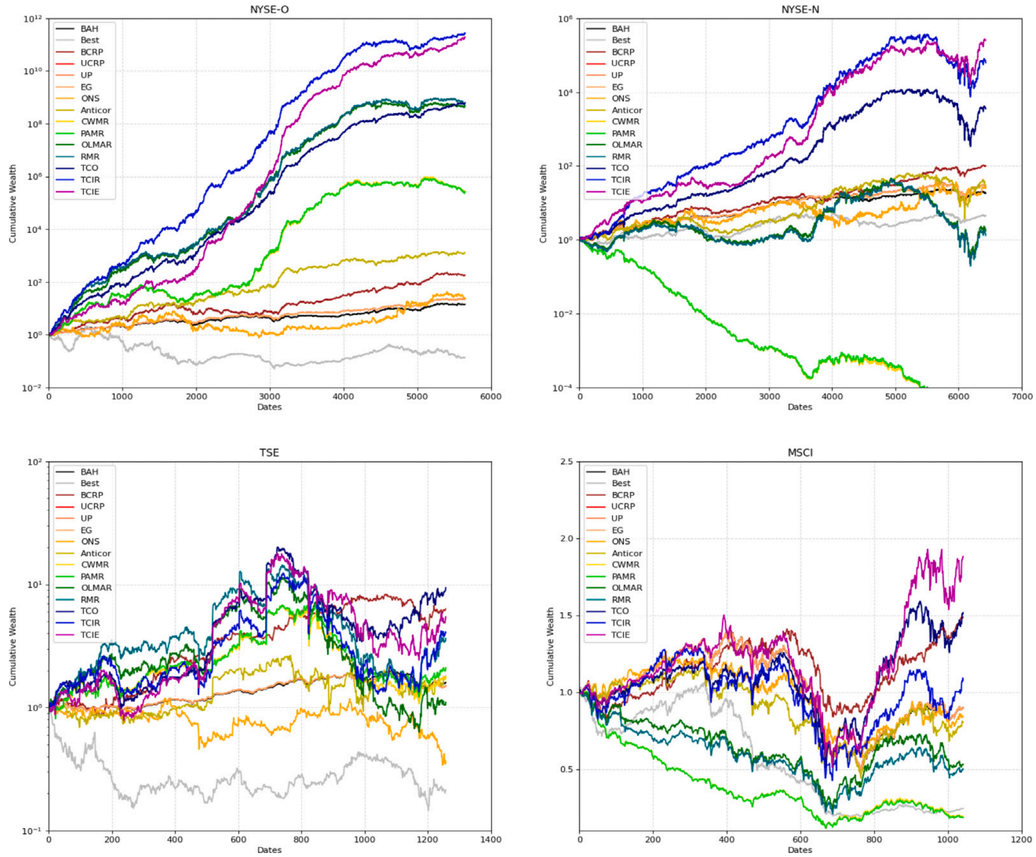

Fig. 3. Cumulative wealth on different data sets.

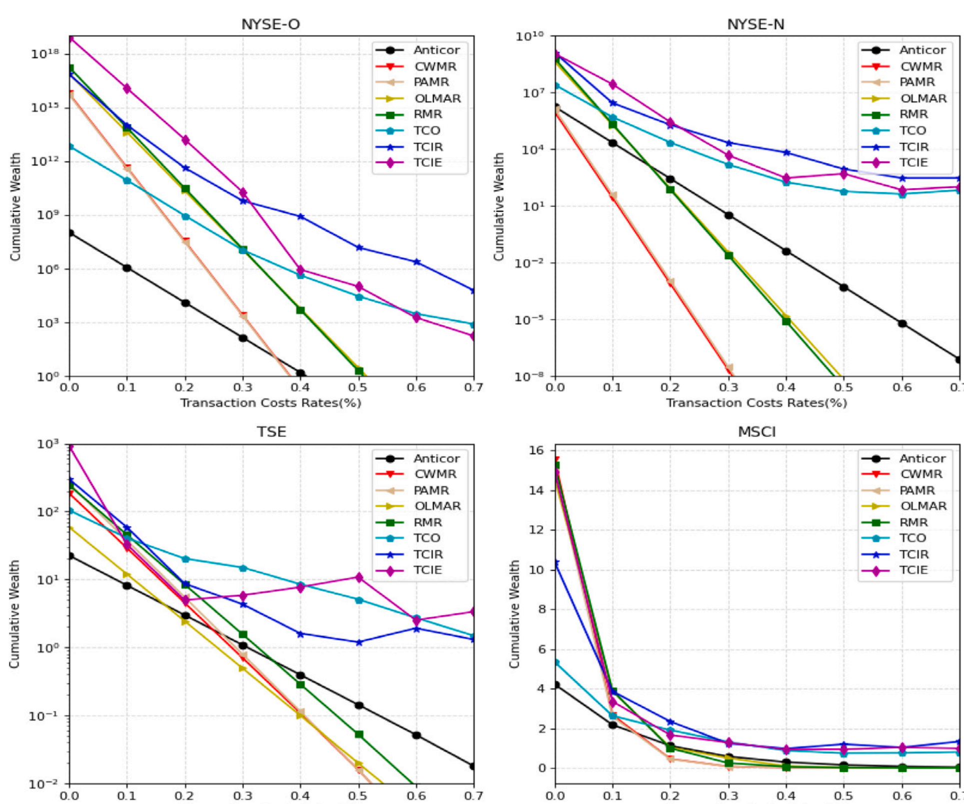

Fig. 4. Cumulative wealth with different transaction cost rates.

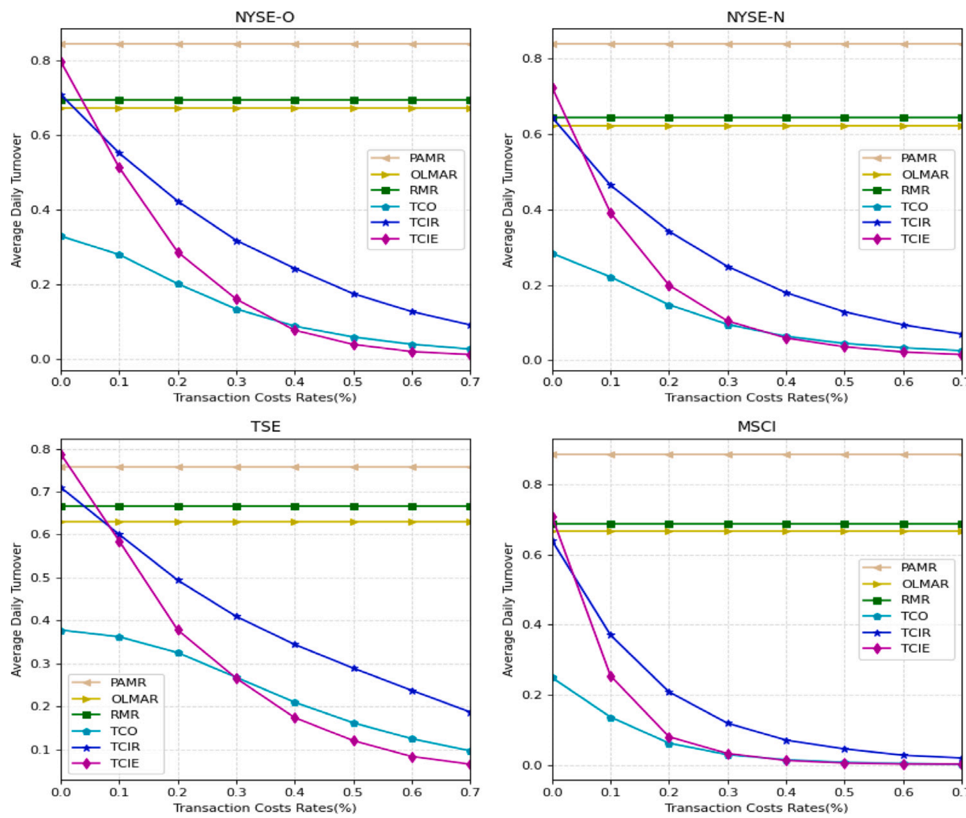

Fig. 5. Average daily turnover with different transaction cost rates.

In most cases, our algorithms (TCIR, TCIE) perform best in the case of non-zero transaction cost rates, which exhibit strong effectiveness and practicability.

The portfolio turnover cannot be neglected in the investment process since a high turnover rate represents a large transaction cost in frequent asset trading. Fig. 5 shows the average daily turnover of some algorithms on the four public data sets. The "Follow the loser" algorithms (Anticor, CWMR, PAMR, OLMAR, RMR) do not consider the transaction costs. Therefore, their average daily turnovers are nearly constant on all the data sets. By contrast, TCO achieves a low turnover all the time, which effectively reduces the influence of transaction costs. Our algorithms (TCIR, TCIE) maintain a high level of turnover in order to capitalize on the assets' volatility when the transaction cost rate is low. As the rates increase, their turnovers consistently decrease to almost zero. By comprehensively analyzing Figs. 4 and 5, one can observe that such trade-off results in our algorithms accumulating significant cumulative wealth when transaction costs are not zero.

4.4. Constant cash inflows analysis

Cash flow refers to the movement of funds resulting from capital injections or withdrawals that occur during the investment procedure. This phenomenon is an inherent and unavoidable aspect of real-world applications. In each period, the insurer will generate premium capital and incur claim payout alongside investment income [40].

4.4.1. Cumulative wealth

In our model, we study the problem of online portfolio selection with a controllable constant cash inflow. With the inflow of cash in each period, the investor's cumulative wealth will increase to a certain extent. In addition, the investment strategy may also be affected. We design a series of experiments and compare their performance to investigate the impact of constant cash inflows on OLPS algorithms.

Table 4 presents the cumulative wealth achieved by different algorithms with constant cash inflow $K = 0.1$ and transaction cost rates of 0%, 0.25%, and 0.5%, respectively. It can be seen clearly that the cumulative wealth achieved by all algorithms increases due to cash inflows. When the transaction cost is zero, the performance of all algorithms is very similar to the case of zero cash inflow. The cumulative wealth achieved by "Follow the loser" algorithms is far greater than that achieved by "Benchmark" strategies and "Follow the winner" algorithms. Our algorithms (TCIR, TCIE) achieve the best performance on NYSE-O, NYSE-N, and TSE under zero transaction cost, which is consistent with previous results under zero cash inflow. TCIR and TCIE dominate other algorithms and suffer fewer losses in most cases on the four data sets when the transaction costs are non-zero. This demonstrates that our proposed TCIR and TCIE are able to effectively tackle the practical online portfolio selection issues when non-zero transaction cost rates and constant cash inflows are involved.

Fig. 6 shows the trend of cumulative wealth on different data sets under the transaction cost rate of 0.25% and constant cash inflow of 0.1. The performances of all algorithms are similar to the zero cash inflow case. TCIE achieves the largest cumulative wealth on NYSE-O, NYSE-N, and MSCI.

4.4.2. Sharpe ratio

Not only are the investors interested in cumulative wealth, but they are also interested in other performance indicators that can be utilized in the evaluation of an OLPS algorithm. One of the most often used metrics in the field of finance is the Sharpe ratio, which compares the performance of an investment to that of a risk-free asset after taking into account the investment's individual level of risk. The Sharpe ratio measures the additional amount of return that an investor obtains for each unit of increase in risk. It is demonstrated in the following manner [41]:

$$SR = \frac{\bar{R} - r_f}{\sigma}. \tag{12}$$

Table 4
Cumulative wealth achieved by various algorithms with transaction cost rates (0%, 0.25% and 0.5%) and constant cash inflow ($K = 0.1$).

Algorithms	NYSE-O			NYSE-N			TSE			MSCI		
	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%
BAH	2393.30	2387.32	2381.34	2554.50	2548.10	2541.72	155.21	154.82	154.43	99.97	99.72	99.47
Best	641.02	482.33	374.46	2040.29	1452.53	1076.31	110.57	103.67	97.61	78.88	65.30	54.97
BCRP	2.07E + 04	1.61E + 04	1.26E + 04	1.35E + 04	1.15E + 04	9.79E + 03	345.46	335.44	325.75	141.55	140.83	**140.11**
UCRP	3893.12	3460.65	3081.39	4191.71	3604.29	3104.19	150.03	145.79	141.69	100.93	99.63	98.35
UP	3883.47	3462.28	3097.40	4128.42	3587.50	3103.84	150.06	145.97	141.77	100.98	99.69	98.39
EG	3904.08	3484.09	3114.12	4131.99	3583.06	3111.56	150.06	146.02	142.11	100.89	99.65	98.43
ONS	6441.42	6416.18	6391.22	4299.60	4197.89	4099.64	60.75	60.51	60.28	94.66	93.76	92.86
Anticor	2.42E + 09	6.89E + 04	77.83	7.50E + 07	5470.29	88.77	1016.38	184.63	58.45	238.17	98.59	51.31
CWMR	8.05E + 16	8.73E + 06	16.43	4.34E + 07	48.25	18.46	3529.13	118.17	26.61	**701.73**	64.08	18.53
PAMR	7.18E + 16	7.92E + 06	15.63	6.13E + 07	49.01	18.63	4759.22	130.66	26.82	679.49	63.49	18.65
OLMAR	9.09E + 17	8.32E + 09	220.60	1.62E + 10	649.63	49.77	996.29	69.48	23.85	640.36	91.85	28.20
RMR	1.99E + 18	9.64E + 09	162.06	2.43E + 10	539.51	46.75	3491.68	153.53	38.42	699.36	95.27	29.10
TCO	1.01E + 14	1.29E + 10	3.16E + 06	9.88E + 08	2.90E + 05	467.69	2666.12	**443.84**	112.73	323.79	152.40	88.16
TCIR	7.09E + 17	3.37E + 12	**6.17E + 08**	**5.21E + 10**	3.23E + 06	**6.65E + 04**	5174.64	225.38	85.74	487.15	123.56	123.52
TCIE	**9.21E + 19**	**5.01E + 12**	3.53E + 06	4.76E + 10	**1.07E + 07**	4.56E + 04	**1.46E + 04**	287.85	**586.15**	680.91	**173.68**	94.58

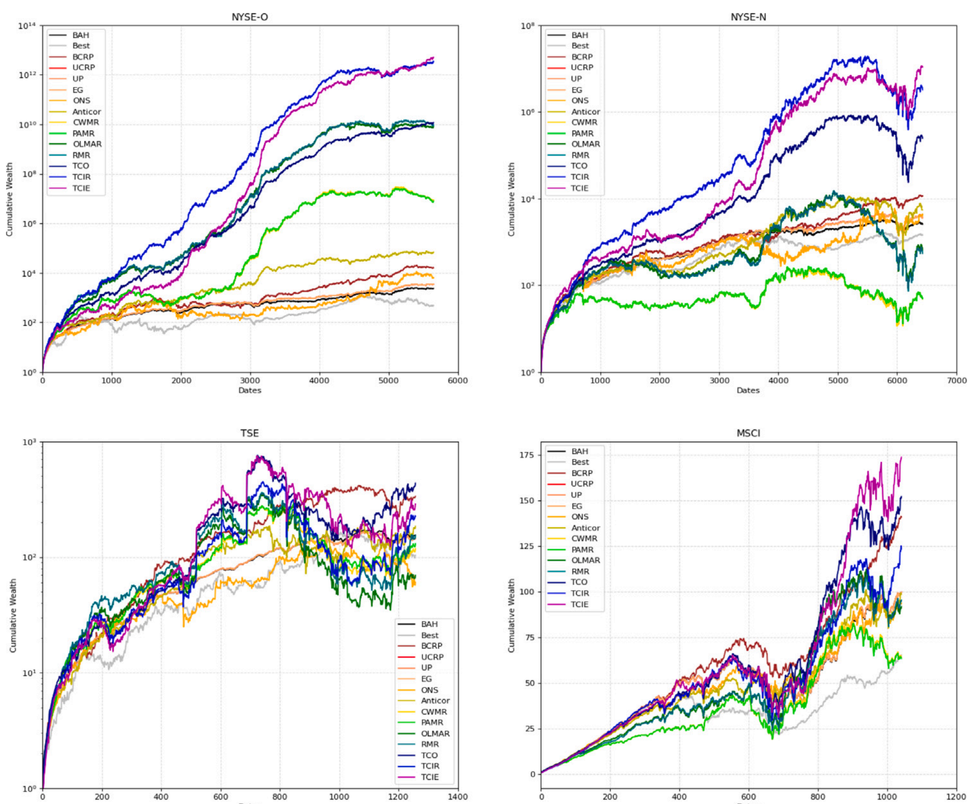

Fig. 6. Cumulative wealth on different data sets.

Here r_f represents the risk-free return, which is assumed to be 1, $\bar{R}$ denotes the average return of the portfolio and σ is the standard deviation of the portfolio. When comparing two portfolio selection algorithms, it is well recognized that the algorithm with a greater Sharpe ratio offers a superior return for the same level of risk.

In Table 5, we set the transaction cost rates to 0%, 0.25%, and 0.5% , respectively, and the constant cash inflow as 0.1. Table 5 shows the Sharpe ratio derived by all algorithms under certain transaction cost rates and constant cash inflows. When the transaction cost rate is zero, TCIE attains the largest Sharpe ratio on TSE and NYSE-O. TCIR achieves the largest Sharpe ratio on NYSE-N. The Sharpe ratios of TCIE on MSCI are slightly smaller than the largest and the second largest values achieved by CWMR and PAMR. When the transaction cost rate is not zero, TCIR and TCIE achieve the best performance on NYSE-O and NYSE-N. The Sharpe ratio of TCIE on TSE exhibits a slightly lower magnitude compared to the maximum value attained by TCO. The overall good performances of our algorithms on the Sharpe ratio imply that they can achieve good returns under controlled investment risk.

4.4.3. Information ratio

The Information ratio is another widely recognized metric for risk-adjusted returns. It measures the excess portfolio returns relative to the returns of a benchmark portfolio, taking into account the volatility of such returns. It is formally characterized as [42]:

$$IR = \frac{\bar{R} - \bar{R}^{\star}}{\sigma(R - R^{\star})}, \tag{13}$$

where $\bar{R}^{\star}$ is the benchmark portfolio's return and $\sigma(R - R^{\star})$ is the standard deviation of the excess return over the benchmark portfolio. Here, we use the UCRP algorithm as the benchmark. Given the amount

Table 5
Sharpe ratio achieved by various algorithms with transaction cost rates (0%, 0.25% and 0.5%) and constant cash inflow ($K = 0.1$).

Algorithms	NYSE-O			NYSE-N			TSE			MSCI		
	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%
BAH	0.0547	0.0547	0.0546	0.0458	0.0458	0.0457	0.0498	0.0495	0.0493	0.0012	0.001	0.0008
Best	0.0085	0.0022	−0.0040	0.0305	0.0223	0.0144	−0.0087	−0.0150	−0.0226	−0.0455	−0.0773	−0.1077
BCRP	0.0599	0.0570	0.0541	0.0555	0.0535	**0.0516**	0.0740	**0.0726**	**0.0713**	0.0378	**0.0372**	**0.0366**
UCRP	0.0729	0.0694	0.066	0.0507	0.0482	0.0457	0.0479	0.0434	0.0388	0.0029	0.0014	0.0000
UP	0.0727	0.0693	0.0659	0.0507	0.0483	0.0458	0.0478	0.0432	0.0388	0.0028	0.0014	0.0000
EG	0.0726	0.0692	0.0659	0.0507	0.0483	0.0459	0.0479	0.0436	0.0392	0.0028	0.0014	0.0000
ONS	0.0334	0.0332	0.0329	0.0325	0.0322	0.0319	−0.0089	−0.0097	−0.0105	0.0023	−0.0004	−0.0030
Anticor	0.1384	0.0607	−0.0201	0.0919	0.0302	−0.0330	0.0818	0.0313	−0.0205	0.0731	−0.0044	−0.0833
CWMR	0.2152	0.0846	−0.0525	0.0852	−0.0554	−0.2036	0.1132	0.0311	−0.0535	**0.1259**	−0.0700	−0.2770
PAMR	0.2149	0.0841	−0.0531	0.0865	−0.0529	−0.1996	0.1179	0.0338	−0.0531	0.1246	−0.0716	−0.2789
OLMAR	0.2106	0.1181	0.0197	0.1038	0.0162	−0.076	0.0823	0.0285	−0.0271	0.1157	−0.0205	−0.1654
RMR	0.2141	0.1188	0.0176	0.1056	0.0149	−0.0805	0.1018	0.0446	−0.0145	0.1179	−0.0231	−0.1730
TCO	0.1979	0.1405	0.0850	0.0988	0.0565	0.0148	0.0945	0.0673	0.0218	0.0859	0.0324	−0.0174
TCIR	0.2089	**0.1475**	**0.1015**	**0.1086**	0.0652	0.0466	0.1048	0.0468	0.0294	0.1042	0.0168	0.0117
TCIE	**0.2334**	0.1464	0.0768	0.1078	**0.0705**	0.0427	**0.1186**	0.0503	0.0586	0.1172	0.0343	−0.0009

Table 6
Information ratio achieved by various algorithms with transaction cost rates (0%, 0.25% and 0.5%) and constant cash inflow ($K = 0.1$).

	NYSE-O			NYSE-N			TSE			MSCI		
Algorithms	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%	0	0.25%	0.5%
BAH	−0.0354	−0.0356	−0.0357	−0.0253	−0.0254	−0.0255	0.006	0.005	0.0041	−0.0343	−0.0371	−0.0396
Best	−0.0141	−0.0208	−0.0273	−0.0092	−0.0184	−0.0267	−0.0208	−0.0273	−0.0351	−0.0584	−0.0959	−0.1307
BCRP	0.0338	0.0303	0.0268	0.021	0.0186	0.0161	0.0621	0.0606	**0.0591**	0.0323	0.0317	**0.0312**
UP	−0.0038	−0.2009	−0.441	−0.0108	−0.1388	−0.2575	−0.0274	−0.2545	−0.5139	−0.0058	−0.254	−0.2811
EG	0.0022	−0.1055	−0.2087	−0.0121	−0.1106	−0.2047	−0.0053	−0.0227	−0.3813	−0.0258	−0.003	−0.2773
ONS	0.0176	0.0173	0.017	0.0188	0.0185	0.0181	−0.0219	−0.0227	−0.0236	−0.0002	−0.0107	−0.0057
Anticor	0.1254	0.04	−0.0493	0.0818	0.01	−0.0637	0.0759	0.0225	−0.0323	0.1146	−0.1131	−0.1414
CWMR	0.2105	0.07	−0.0784	0.0744	−0.0876	−0.2608	0.1094	0.0234	−0.0655	**0.1898**	−0.1157	−0.4489
PAMR	0.2102	0.0695	−0.0791	0.0761	−0.0845	−0.256	0.1144	0.0263	−0.065	0.1877	−0.1157	−0.4519
OLMAR	0.2059	0.1072	0.0015	0.0965	−0.0014	−0.1057	0.0779	0.0225	−0.035	0.169	−0.0344	−0.2649
RMR	0.2097	0.1081	−0.0006	0.0987	−0.003	−0.1111	0.0980	0.0391	−0.0221	0.1726	−0.0386	−0.2776
TCO	0.1954	0.1317	0.0701	0.0915	0.0431	−0.0053	0.0900	**0.0691**	0.0147	0.1549	0.0522	−0.0363
TCIR	0.2042	**0.1390**	**0.0895**	**0.1017**	0.0536	**0.0330**	0.101	0.0413	0.0234	0.1507	0.022	0.0145
TCIE	**0.2303**	0.1374	0.0617	0.1009	**0.0595**	0.0282	**0.1151**	0.0449	0.0531	0.1713	**0.0500**	−0.005

of risk that was taken, a higher Information ratio represents a higher excess return of the algorithm.

Table 6 shows the Information ratio achieved by all algorithms under different transaction cost rates and constant cash inflow of 0.1. TCIR and TCIE perform well on the four public data sets. The overall performance of our algorithms on the Information ratio is very similar to the performance on the Sharpe ratio. Our algorithms are able to achieve good returns when compared to the UCRP algorithm.

4.4.4. Various cash inflows

To analyze the performance of OLPS algorithms under different cash inflows, we let the transaction cost rate be 0.25% and constant cash inflows ranging from 0 to 0.5. The cumulative wealth achieved by several algorithms is illustrated in Fig. 7. The evidence suggests that there is a positive correlation between the cumulative wealth and the constant cash inflow. Except in the case of zero cash inflow, cumulative wealth shows a linear growth trend across all data sets. In comparison to other algorithms, TCIE demonstrates superior performance in terms of cumulative wealth on MSCI, NYSE-O, and NYSE-N while ranking second in cumulative wealth on TSE. It demonstrates the effectiveness and practicality of our algorithm when incorporating transaction costs and constant cash inflows.

4.5. Parameter sensitivity

We then evaluate the sensitivity of the trade-off parameter λ. The transaction cost rate is set as 0.25% and constant cash inflow as 0.1. We derive the cumulative wealth of TCIR and TCIE with various λ/γ ranging from 0 to 25 with stepwise 5 (see Fig. 8). TCIR achieves the relatively maximum cumulative wealth on NYSE-N and MSCI when $\lambda/\gamma = 10$ and the relatively largest value on NYSE-O when $\lambda/\gamma = 5$. For TSE, the cumulative wealth achieved by TCIR increases as λ/γ becomes larger. In comparison, TCIE performs the relatively maximum cumulative wealth on MSCI, NYSE-O and NYSE-N when $\lambda/\gamma = 5$, and the relatively maximum on TSE when $\lambda/\gamma = 10$. This may explain why the performance of our algorithms on TSE is not as good as we expected when $\lambda/\gamma = 5$. Based on the analysis, we can see that the cumulative wealth achieved by our algorithms differs a lot under various values of λ/γ. Previously, we set $\lambda/\gamma = 5$ in uniform, which is not the best parameter selection. This validates that our proposed TCIR and TCIE still dominate previous online portfolio selection algorithms even when the parameter may not be the best.

By setting different values of λ/γ, we can certainly derive the relatively good or bad performance of our algorithms on each dataset. Therefore, we try to find the relatively best performance in hindsight, within the range of λ/γ in $[0.1, 25]$. This may give the investor some suggestions on the value of λ/γ for achieving a good performance on different datasets, as well as demonstrate the effectiveness and practicality of our algorithms. We split the range $[0.1, 25]$ into 250 different values with a stepwise 0.1. Then we test the cumulative wealth achieved under different values of λ/γ and derive the relatively best cumulative wealth achieved by our algorithms in Table 7. In Table 7, "TCIR(5.3)" means that algorithm TCIR achieves its relatively best cumulative wealth on NYSE-O when $\lambda/\gamma = 5.3$. For simplicity, we set the value of the parameter λ/γ in uniform in our previous experiments. Table 7 shows that our algorithms perform much better if we set different values of λ/γ on different datasets with different algorithms. It also gives the suggested value of λ/γ, which can achieve the relatively best cumulative wealth in hindsight. Overall, the results in Table 7 are better than our algorithms with $\lambda/\gamma = 5$ in uniform and much better than previous algorithms.

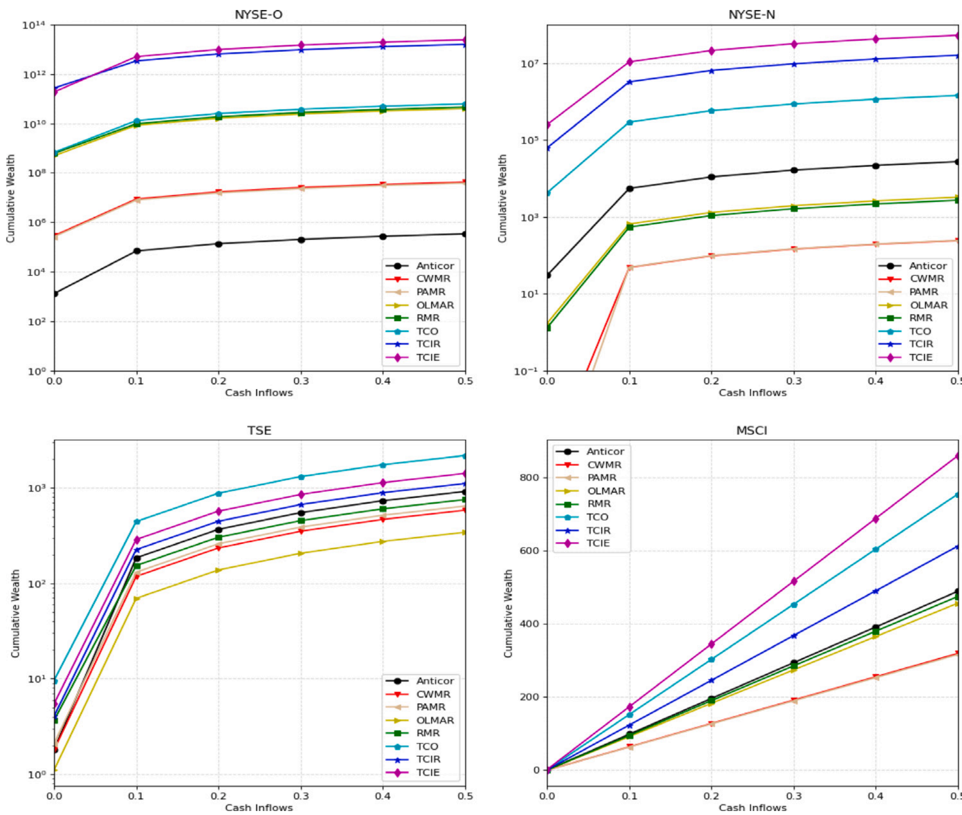

Fig. 7. Cumulative wealth with different cash inflows.

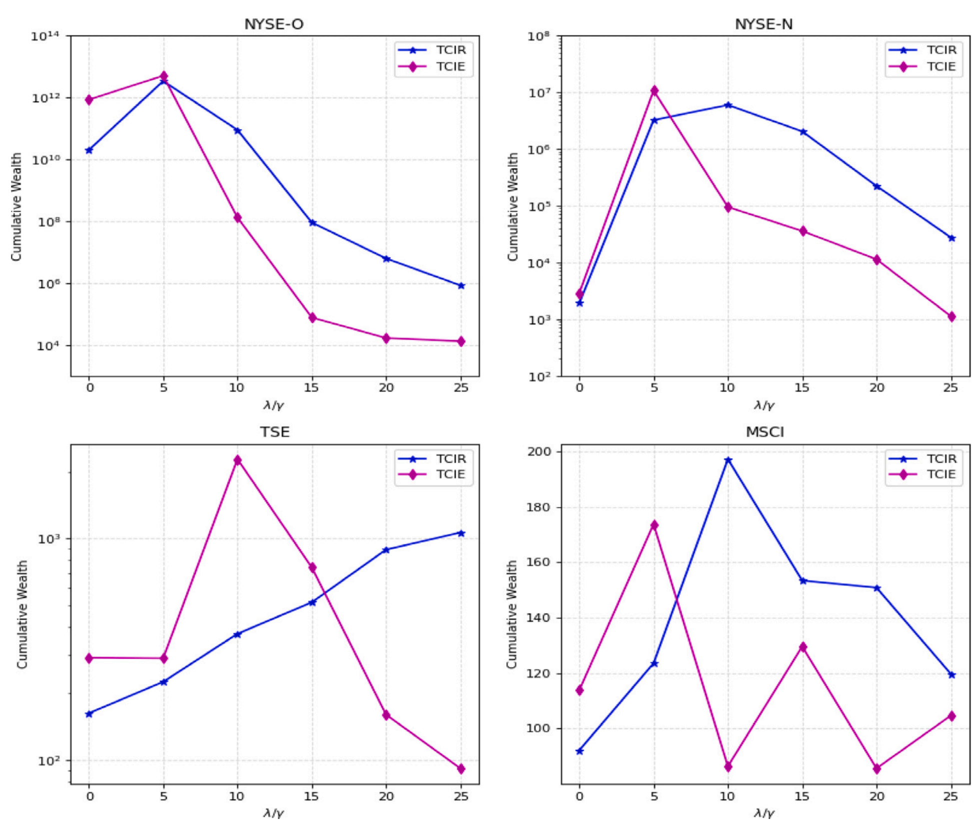

Fig. 8. Cumulative wealth with different trade-off parameters.

Table 7
Relatively best cumulative wealth achieved on the data sets with different λ/γ.

NYSE-O		NYSE-N		TSE		MSCI	
TCIR(5.3)	TCIE(3.1)	TCIR(7.7)	TCIE(4.1)	TCIR(24.8)	TCIE(10.1)	TCIR(10.3)	TCIE(7.6)
7.80E + 12	2.13E + 14	3.90E + 07	2.07E + 07	1267.27	2588.23	214.49	176.15

5. Conclusions

In this paper, the problem of online portfolio selection in the presence of transaction costs and constant cash inflows is well studied. We first propose a general framework for online portfolio selection, incorporating constant cash inflows and transaction costs. Through an investigation of the correlation between transaction costs and the transaction remainder factor, we present a novel approach to address transaction costs and calculate the accurate transaction remainder factor and portfolio vector simultaneously. And the L_1-penalty method is employed to control the loss by limiting trading volumes. An uncertainty set is also incorporated to minimize the impact of the estimation error on the optimal allocation of a portfolio. TCIR and TCIE are designed based on the proposed online portfolio selection model, and a series of numerical experiments are carried out to validate the efficiency of our algorithms. Compared with previous algorithms, TCIR and TCIE show superior performance in cumulative wealth and robustness even under different transaction cost rates and cash flows. In addition, TCIR and TCIE effectively reallocate constant cash inflows, demonstrating better performance across the return–risk measures such as Sharpe ratio and Information ratio. Parameter sensitivity analysis further validates the satisfactory cumulative wealth performance of our algorithms under different trade off parameters. Overall, the numerical experiments validate the proposed algorithms' ability of effectively handling transaction costs and constant cash inflows, which assists in improving the holistic investment performance.

There are still some downsides that should be carefully evaluated, despite the fact that TCIR and TCIE accomplish good performance in the investment. The first one is that cash inflows are not always constant in real investment behavior. The investors may adjust their cash inflows and outflows based on the performance of risky assets and other metrics. It is necessary to consider adaptive cash inflows and outflows in each trading period. The second drawback is that the proposed linear programming model sometimes results in an over-centralized investment strategy, where all the capital is allocated to one single risky asset. In the future, the diversification degree of the portfolio can be incorporated into the objective and appropriate linear programming solutions can be explored to ensure a diversified strategy. Finally, exploring more comprehensive techniques for return prediction, such as deep learning, can enhance the accuracy, which reduces the prediction error and narrows the uncertainty set in Eq. (7), leading to better-informed decision-making and thereby enhancing the robustness of the optimization model.

CRediT authorship contribution statement

Benmeng Lyu: Writing – original draft, Validation, Software, Methodology, Formal analysis. **Boqian Wu:** Software, Formal analysis, Data curation. **Sini Guo:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Jia-Wen Gu:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Wai-Ki Ching:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

The authors would like to thank the editor and the three anonymous referees for their helpful comments and suggestions. This research work was supported by National Natural Science Foundation of China Under Grant number 72201033 and 72002201, Hong Kong Research Grants Council under Grant Number 17309522, Hung Hing Ying Physical Science Research Grant, The University of Hong Kong, Hong Kong, Introduction Project of China Postdoctoral International Exchange Program under Grant Number YJ20220045, China Postdoctoral Science Foundation, China under Grant Number 2022M720426, and Beijing Institute of Technology Research Fund Program for Young Scholars, Guangdong Basic and Applied Basic Research Foundation 2023A1515030197, Shenzhen Humanities & Social Sciences Key Research Bases.

Appendix A

In order to assess the accuracy of the prediction techniques RMR and EMA, we select a dataset called TSE [7]. This dataset comprises the historical prices of 88 stocks listed on the Toronto Stock Exchange, spanning from January 4, 1994, to December 31, 1998.

The prediction absolute error for stock i at tth trading day can be calculated as follows:

$$e_{t,i} = |x_{t,i} - \tilde{x}_{t,i}|,$$

and the average absolute error is

$$\bar{e}_i = \frac{1}{n}\sum_{t=1}^{n}|x_{t,i} - \tilde{x}_{t,i}|.$$

The average absolute errors for the stocks in TSE are displayed in Fig. 9, from which we can observe that RMR and EMA both achieve a small average absolute error, and EMA performs relatively better.

Appendix B

The section gives the proof of Theorem 1.

Proof. For any feasible solution $\mathbf{b}_t$ of Problem (9), w_t can be determined by solving Eq. (2). And one corresponding feasible solution $(\mathbf{u}_t, \mathbf{v}_t)$ of Problem (10) can be found by solving Eq. (4): if $b_{t,i}w_{t-1} > \hat{b}_{t-1,i}$, $u_{t,i} = b_{t,i}w_{t-1} - \hat{b}_{t-1,i}$ and $v_{t,i} = 0$; otherwise, we have $u_{t,i} = 0$ and $v_{t,i} = \hat{b}_{t-1,i} - b_{t,i}w_{t-1}$, $i = 0, 1, \ldots, m$.

For any feasible solution $(\mathbf{u}_t, \mathbf{v}_t)$ of Problem (10), there are four cases:

(1) $u_{t+1,i} > 0, v_{t+1,i} = 0$;
(2) $u_{t+1,i} = 0, v_{t+1,i} > 0$;
(3) $u_{t+1,i} = 0, v_{t+1,i} = 0$;
(4) $u_{t+1,i} > 0, v_{t+1,i} > 0$.

For any feasible solution $\mathbf{b}_t$ of Problem (9), the corresponding feasible solution $(\mathbf{u}_t, \mathbf{v}_t)$ of the problem (10) falls into one of the first three cases. Similarly, for any feasible solution $(\mathbf{u}_t, \mathbf{v}_t)$ belonging to the first three cases, we can derive the corresponding feasible solution $\mathbf{b}_t$ of Problem (9):

if $u_{t+1,i} > 0, v_{t+1,i} = 0$, then $b_{t+1,i}w_t = u_{t+1,i} + \hat{b}_{t,i}$;
if $u_{t+1,i} = 0, v_{t+1,i} > 0$, then $b_{t+1,i}w_t = \hat{b}_{t,i} - v_{t+1}$;

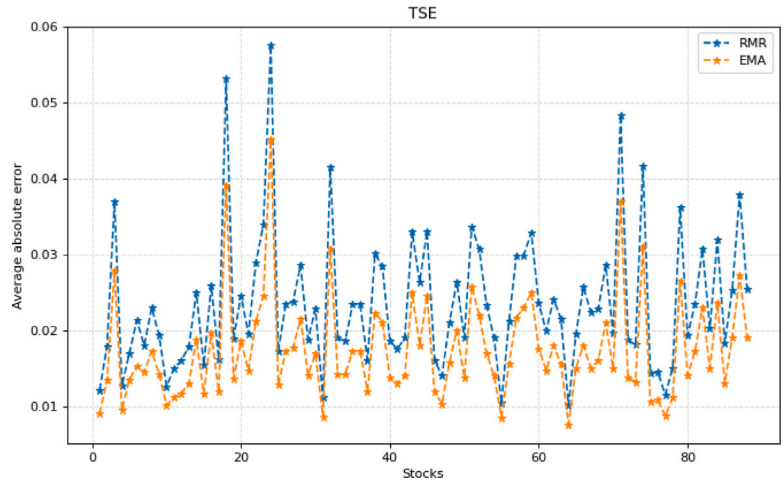

Fig. 9. Comparison of RMR and EMA.

if $u_{t+1,i} = 0, v_{t+1,i} = 0$, then $b_{t+1,i}w_t = \hat{b}_{t,i}$.

Using Eqs. (5)–(6), one can find the corresponding feasible solution $\mathbf{b}_t$. This implies that there is a one-to-one correspondence relationship between all the feasible solutions $\mathbf{b}_t$ of Problem (9) and the feasible solutions $(\mathbf{u}_t, \mathbf{v}_t)$ of Problem (10) which belong to the first three cases.

Then, it is sufficient to prove that the optimal solution of Problem (10) cannot fall into Case (4). Denote the optimal solution to Problem (10) as $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$. For $j = 0, 1, \dots, m$, if $u_{t,j}^\star > 0, v_{t,j}^\star > 0$, it is reasonable assume that $u_{t,j}^\star > v_{t,j}^\star > 0$. According to Problem (10), $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ meets the condition:

$$(\mathbf{u}_t^\star - \mathbf{v}_t^\star)\mathbf{1} + \gamma(\mathbf{u}_t^\star + \mathbf{v}_t^\star)\mathbf{1} - \gamma(u_{t,0}^\star + v_{t,0}^\star) = 0.$$

For $j = 0$, we can find another pair of $(\mathbf{u}_t', \mathbf{v}_t')$ to replace $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$, where $u_{t,i}' = u_{t,i}^\star, v_{t,i}' = v_{t,i}^\star$ for $i = 1, \dots, m$. And we set that $v_{t,0}' = 0$. In this case, $(\mathbf{u}_t', \mathbf{v}_t')$ meets the condition:

$$(\mathbf{u}_t' - \mathbf{v}_t')\mathbf{1} + \gamma(\mathbf{u}_t' + \mathbf{v}_t')\mathbf{1} - \gamma(u_{t,0}' + v_{t,0}') = 0.$$

Then

$$u_{t,0}' + \sum_{i\neq 0}(u_{t,i}' - v_{t,i}') + \gamma u_{t,0}' + \gamma \sum_{i\neq 0}(u_{t,i}' + v_{t,i}') - \gamma u_{t,0}' = 0.$$

$$u_{t,0}' + \sum_{i\neq 0}(u_{t,i}^\star - v_{t,i}^\star) + \gamma u_{t,0}' + \gamma \sum_{i\neq 0}(u_{t,i}^\star + v_{t,i}^\star) - \gamma u_{t,0}' = 0.$$

$$u_{t,0}' = -\sum_{i\neq 0}(u_{t,i}^\star - v_{t,i}^\star) - \gamma \sum_{i\neq 0}(u_{t,i}^\star + v_{t,i}^\star) = u_{t,0}^\star - v_{t,0}^\star.$$

It is straightforward to prove that the objective value of the objective function with $(\mathbf{u}_t', \mathbf{v}_t')$ is greater than the objective value of the objective function with $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ in Problem (10).

For $j = 1, \dots, m$, we can find another pair of $(\mathbf{u}_t', \mathbf{v}_t')$ to replace $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$, where $u_{t,i}' = u_{t,i}^\star, v_{t,i}' = v_{t,i}^\star$for $i = 0, 1, \dots, j-1, j+1, \dots, m$. And we also set $v_{t,j}' = 0$. Then we can derive

$$u_{t,j}' + \sum_{i\neq j}(u_{t,i}^\star - v_{t,i}^\star) + \gamma u_{t,j}' + \gamma \sum_{i\neq j}(u_{t,i}^\star + v_{t,i}^\star) - \gamma(u_{t,0}^\star + v_{t,0}^\star) = 0,$$

$$(1+\gamma)u_{t,j}' = -\sum_{i\neq j}(u_{t,i}^\star - v_{t,i}^\star) - \gamma \sum_{i\neq j}(u_{t,i}^\star + v_{t,i}^\star) + \gamma(u_{t,0}^\star + v_{t,0}^\star),$$

$$(1+\gamma)u_{t,j}' = (u_{t,j}^\star - v_{t,j}^\star) + \gamma(u_{t,j}^\star + v_{t,j}^\star),$$

$$u_{t,j}' = \frac{(u_{t,j}^\star - v_{t,j}^\star) + \gamma(u_{t,j}^\star + v_{t,j}^\star)}{1+\gamma}.$$

In the meantime, we have

$$(u_{t,j}' - v_{t,j}') - (u_{t,j}^\star - v_{t,j}^\star) = \frac{2\gamma}{1+\gamma} v_{t,j}^\star > 0$$

and

$$(u_{t,j}' + v_{t,j}') - (u_{t,j}^\star + v_{t,j}^\star) = \frac{-2}{1+\gamma} v_{t,j}^\star < 0.$$

One can see that the objective value of the objective function with $(\mathbf{u}_t', \mathbf{v}_t')$ is greater than the objective value of the objective function with $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$. Therefore, $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ with $u_{t,j}^\star > 0, v_{t,j}^\star > 0$ cannot be the optimal solution. Therefore, $\mathbf{b}_t^\star$ is the optimal solution of Problem (9) only when $(\mathbf{u}_t^\star, \mathbf{v}_t^\star)$ is the optimal solution of Problem (10). The proof is completed. □

References

[1] Markowitz H. Portfolio selection. J Finance 1952;7(1):77–91.
[2] Kelly JL. A new interpretation of information rate. Bell Syst Tech J 1956;35(4):917–26.
[3] Guan H, An Z. A local adaptive learning system for online portfolio selection. Knowl-Based Syst 2019;186:104958.
[4] Marques AC, Frej EA, de Almeida AT. Multicriteria decision support for project portfolio selection with the FITradeoff method. Omega 2022;111:102661.
[5] Fereydooni A, Barak S, Sajadi SMA. A novel online portfolio selection approach based on pattern matching and ESG factors. Omega 2024;123:102975.
[6] Li B, Hoi SC. Online portfolio selection: A survey. ACM Comput Surv 2014;46(3):1–36.
[7] Li B, Hoi SCH. Online portfolio selection: principles and algorithms. Boca Raton: CRC Press; 2018.
[8] Cover TM. Universal portfolios. Math Finance 1991;1(1):1–29.
[9] Helmbold DP, Schapire RE, Singer Y, Warmuth MK. On-line portfolio selection using multiplicative updates. Math Finance 1998;8(4):325–47.
[10] Agarwal A, Hazan E, Kale S, Schapire RE. Algorithms for portfolio management based on the newton method. In: Proceedings of the international conference on machine learning. 2006, p. 9–16.
[11] Borodin A, El-Yaniv R, Gogan V. Can we learn to beat the best stock. J Artificial Intelligence Res 2004;21(1):579–94.
[12] Li B, Hoi SC, Zhao P, Gopalkrishnan V. Confidence weighted mean reversion strategy for online portfolio selection. ACM Trans Knowl Discov Data 2013;7(1):1–38.
[13] Li B, Zhao P, Hoi SC, Gopalkrishnan V. PAMR: Passive aggressive mean reversion strategy for portfolio selection. Mach Learn 2012;87(2):221–58.
[14] Li B, Hoi SC. On-line portfolio selection with moving average reversion. In: Proceedings of the international conference on machine learning. 2012, p. 273–80.
[15] Huang D, Zhou J, Li B, Hoi SC, Zhou S. Robust median reversion strategy for online portfolio selection. IEEE Trans Knowl Data Eng 2016;28(9):2480–93.
[16] Das P, Banerjee A. Meta optimization and its application to portfolio selection. In: Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining. 2011, p. 1163–71.
[17] Zhang Y, Lin H, Yang X, Long W. Combining expert weights for online portfolio selection based on the gradient descent algorithm. Knowl-Based Syst 2021;234:107533.
[18] Blum A, Kalai A. Universal portfolios with and without transaction costs. Mach Learn 1999;35(3):193–205.
[19] Albeverio S, Lao L, Zhao X. On-line portfolio selection strategy with prediction in the presence of transaction costs. Math Methods Oper Res 2001;54(1):133–61.

[20] Das P, Johnson N, Banerjee A. Online lazy updates for portfolio selection with transaction costs. In: Proceedings of the AAAI conference on artificial intelligence. 2013, p. 202–8.
[21] Guo S, Gu J-W, Ching W-K. Adaptive online portfolio selection with transaction costs. European J Oper Res 2021;295(3):1074–86.
[22] Jiang Z, Xu D, Liang J. A deep reinforcement learning framework for the financial portfolio management problem. 2017, arXiv preprint arXiv:1706.10059.
[23] Li B, Wang J, Huang D, Hoi SC. Transaction cost optimization for online portfolio selection. Quant Finance 2018;18(8):1411–24.
[24] Guo S, Gu J-W, Fok CH, Ching W-K. Online portfolio selection with state-dependent price estimators and transaction costs. European J Oper Res 2023;311(1):333–53.
[25] Dimson E, Minio-Kozerski C. Closed-end funds: A survey. Financial Mark Inst Instrum 1999;8(2):1–41.
[26] Edelen RM. Investor flows and the assessed performance of open-end mutual funds. J Financ Econ 1999;53(3):439–66.
[27] Zhang Y, Zhao P, Wu Q, Li B, Huang J, Tan M. Cost-sensitive portfolio selection via deep reinforcement learning. IEEE Trans Knowl Data Eng 2022;34(1):236–48.
[28] Bauer E, Kohavi R. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Mach Learn 1999;36(1):105–39.
[29] Györfi L, Vajda I. Growth optimal investment with transaction costs. In: Proceedings of the international conference on algorithmic learning theory. 2008, p. 108–22.
[30] Guo S, Gu J-W, Ching W-K, Lyu B. Adaptive online mean-variance portfolio selection with transaction costs. Quant Finance 2024;24(1):59–82.
[31] Toloo M, Mensah EK, Salahi M. Robust optimization and its duality in data envelopment analysis. Omega 2022;108:102583.
[32] Georgantas A, Doumpos M, Zopounidis C. Robust optimization approaches for portfolio selection: a comparative analysis. Ann Oper Res 2021;301(1):1–17.
[33] Gorissen BL, Yanıkoğlu İ, den Hertog D. A practical guide to robust optimization. Omega 2015;53:124–37.
[34] Peng X, Xu D, Kong L, Chen D. L1-norm loss based twin support vector machine for data recognition. Inform Sci 2016;340:86–103.
[35] Györfi L, Lugosi G, Udina F. Nonparametric kernel-based sequential investment strategies. Math Finance 2006;16(2):337–57.
[36] Fabozzi FJ, Focardi SM, Kolm PN, Pachamanova DA. Robust portfolio optimization and management. New York: John Wiley & Sons; 2007.
[37] Tsagaris T, Jasra A, Adams N. Robust and adaptive algorithms for online portfolio selection. Quant Finance 2012;12(11):1651–62.
[38] Bodie Z, Kane A, Marcus A. Essentials of investments. New York: McGraw Hill; 2013.
[39] North DC, North DC. Transaction costs, institutions, and economic performance. San Francisco: ICS Press; 1992.
[40] Wu H, Li Z. Multi-period mean–variance portfolio selection with regime switching and a stochastic cash flow. Insurance Math Econom 2012;50(3):371–84.
[41] Sharpe WF. Mutual fund performance. J Bus 1966;39(1):119–38.
[42] Goodwin TH. The information ratio. Financ Anal J 1998;54(4):34–43.