# Adaptive Peak Price with Lazy Updates for Short-term Portfolio Optimization

Kailin Xie[0009−0002−9948−4596] and Ying Chu[0000−0002−5905−3989] (✉)

Department of Artificial Intelligence, Shenzhen University, P.R.China
xiekailin2021@email.szu.edu.cn, chuying@szu.edu.cn

**Abstract.** This paper introduces the novel Adaptive Peak Price with Lazy Updates (APPLU) approach for short-term portfolio optimization (SPO), a method that innovatively combines the Radial Basis Function (RBF) and a new lazy update approach to address the unique challenges of SPO. Our approach is tailored to balance the dual objectives of maximizing returns and minimizing transaction costs, which are critical concerns in dynamically allocating wealth among various assets over time. Unlike conventional methods that primarily rely directly on peak price information, APPLU introduces an adaptive peak price using RBFs to capture continuous depreciation information of assets, thereby alleviating the aggressive nature of traditional peak price strategies and avoiding frequent trades. Furthermore, we propose a new lazy update approach that employs unsquared $l^2$-norm regularization to represent portfolio changes. This approach contrasts with squared $l^2$-norm regularization, which disproportionately penalizes larger portfolio changes while being more lenient towards smaller changes. Thereby, our methodology offers a more balanced and effective approach to portfolio adjustment. Extensive experiments conducted on seven real datasets demonstrate that APPLU outperforms existing strategies in terms of cumulative return and risk-adjusted return, while effectively controlling transaction costs and maintaining a moderate wealth accumulation strategy.

**Keywords:** Short-term portfolio optimization · Lazy update · Radial basis function · Online learning.

## 1 Introduction

Short-term portfolio optimization (SPO) [13] focuses on the continuous and dynamic allocation of wealth across various assets over time within the field of machine learning. Its primary goal is to optimize the cumulative return of a portfolio, taking into account transaction costs and other practical constraints. A key challenge in SPO is the necessity of frequent allocation decisions at every time point, potentially leading to numerous asset trades.

The regularization technique is a primary approach for controlling transaction costs. Specifically, the $l^2$-norm squared regularization term is used to constrain portfolio changes, resulting in a lazy update approach. The Online Lazy Update (OLU) strategy [4] employs $l^2$-norm squared for portfolio change constraints and utilizes $l^1$-norm to induce sparsity in these changes. The Doubly Regularized Portfolio (DRP) [18] also employs a lazy update approach using $l^2$-norm squared but introduces sparsity directly to the portfolio itself. Extending this concept, Online Lazy Update with Group Sparsity [5] leverages sector information for group sparsity while implementing lazy updates to the portfolio. In addition to the lazy update approach, Transaction Cost Optimization (TCO) [17] considers the difference of portfolio between market close and open, using the $l^1$-norm to directly minimize the transaction costs. However, the use of $l^2$-norm squared in lazy update approach presents two issues due to square calculations. Firstly, combining $l^1$-norm with $l^2$-norm squared might lead to an awkward balance between sparsity and turnover, as evidenced by DRP's experimental results, which show a more significant effect on turnover than sparsity. Secondly, $l^2$-norm squared tends to encourage smaller portfolio changes while disproportionately penalizing larger adjustments.

Many advanced SPO strategies employ peak price for short-term price information. For instance, Short-term Sparse Portfolio Optimization [14], Short-term Portfolio Optimization with Loss Control [12] and Peak Price Tracking Approach [3] use the highest asset price within a certain time window to estimate expected asset prices, capturing investor irrationality and potentially achieving favorable incomes. However, peak price strategies cannot capture continuous asset depreciation as they omit price information other than the highest price. This limitation could lead to risky investments and substantial losses. Moreover, being inherently aggressive, peak price strategies might result in high turnover in pursuit of wealth.

Radial Basis Functions (RBF) have been increasingly acknowledged for their effectiveness across various applications, notably in portfolio optimization. Their applicability goes beyond traditional neural network domains, covering function approximation, clustering, classification, and solving complex nonlinear problems [9]. This versatility of RBFs also extends to financial modeling, where they have shown significant potential. For instance, the Adaptive Input and Composite Trend Representation System [11] utilizes RBFs to detect subtle trends and patterns in asset prices, embedding them into price prediction models to modulate the impact of different trends. This inspired us to develop a set of RBFs capable of effectively capturing continuous depreciation information, complementing our new lazy update approach.

To address the above-mentioned problems, we introduce the Adaptive Peak Price with Lazy Updates (APPLU) method for short-term portfolio optimization. APPLU effectively narrows the gap between maximizing returns and minimizing costs in SPO. Initially, we formulate an adaptive peak price using radial basis functions to represent the disparity between actual and peak prices, mitigating the innate aggressiveness of traditional peak price strategies and better aligning with the goal of reducing transaction costs. Furthermore, we incorporate an unsquared $l^2$-norm regularization for portfolio changes, providing a more transparent representation of these changes compared to the squared $l^2$-norm, which can lead to disproportionate penalties. Our contributions include:

- **Adaptive Peak Price with RBFs:** APPLU uniquely uses Radial Basis Functions (RBFs) to create an adaptive peak price model. This approach captures a comprehensive picture of asset prices, improving upon traditional peak price strategies that only consider the highest prices.
- **Unsquared $l^2$-Norm Regularization:** APPLU incorporates an unsquared $l^2$-norm regularization to manage portfolio changes, providing a consistent approach to portfolio changes. This method is a departure from the conventional squared $l^2$-norm regularization, ensuring a more balanced and practical response to market shifts.
- **Balanced Strategy:** Comprehensive experiments on seven real datasets demonstrate that APPLU successfully balances the maximization of returns with the minimization of transaction costs, ensuring a moderate strategy for wealth accumulation.

## 2 Problem Setting

In this paper, we use a standard and universal setting of portfolio selection in machine learning[12,3,17]. Consider a financial market with $m$ assets for $n$ period. At the end of the $t^{th}$ period, a non-negative $m$-dimensional vector $\mathbf{p}_t \in \mathbb{R}_+^m (t = 1, 2, \ldots, n)$ represents the close price of assets. A *relative price vector* [2] is introduced to see the change of asset prices as $\mathbf{x}_t \triangleq \frac{\mathbf{p}_t}{\mathbf{p}_{t-1}}$, where a division between two vectors represents *element-wise division* in this paper.

At the beginning of the $t^{th}$ period, an investment in the market is specified by a *portfolio vector* in $m$ dimensional simplex $\mathbf{b}_t \in \Delta_m := \left\{\mathbf{b} \in \mathbb{R}_+^m : \sum_{i=1}^m \mathbf{b}^{(i)} = 1\right\}$, where $\mathbf{b}_t^{(i)}$ denotes the proportion of total wealth invested in the $i$th asset. The non-negative constrain means no short is allowed and the equality constraint means that the portfolio is self-financed.

For the $t^{th}$ trading day, a portfolio $\mathbf{b}_t$ generated by the portfolio selection strategy results in a daily return $\mathbf{b}_t^\top \mathbf{x}_t$. When there is a transaction cost rate of $r$ for each trade in the portfolio re-balancing process, the cumulative wealth can be determined using the proportional transaction cost model [15] as:

$$S_n^r = S_0 \prod_{t=1}^{n} \left[ \left( \hat{\mathbf{b}}_t^\top \mathbf{x}_t \right) \times \left( 1 - \frac{r}{2} \sum_{i=1}^{m} \left| \hat{\mathbf{b}}_t^{(i)} - \tilde{\mathbf{b}}_{t-1}^{(i)} \right| \right) \right], \tag{1}$$

where $\tilde{\mathbf{b}}_{t-1}^{(i)} = \frac{\hat{\mathbf{b}}_{t-1}^{(i)} * \mathbf{x}_{t-1}^{(i)}}{\hat{\mathbf{b}}_{t-1}^\top \mathbf{x}_{t-1}}$ is the price adjusted portfolio of asset $i$ in the $(t-1)$th period. The term $(r/2) \sum_{i=1}^m \left| \hat{\mathbf{b}}_t^{(i)} - \tilde{\mathbf{b}}_{t-1}^{(i)} \right|$ represents the transaction cost incurred from the adjustment of portfolio $\tilde{\mathbf{b}}_{t-1}$ to $\mathbf{b}_t$ through re-balancing.

Finally, a portfolio learning algorithm learns sequentially a set of portfolio vectors $\{\mathbf{b}_t\}_{t=1}^n$ to maximize the final cumulative wealth as well as satisfy some risk management metrics.

## 3 Methodology

In this section, we propose the novel Adaptive Peak Price with Lazy Updates (APPLU) for short-term portfolio optimization, aiming to leverage the benefits of peak price in wealth generation and incorporate a lazy update strategy to reduce transaction costs.

### 3.1 Adaptive Peak Price

To capture the continuous depreciation information of assets and alleviate the aggressiveness of peak price, a radial basis function is introduced to adjust the peak price. We assume that there are $m$ prices of assets in a time window. The peak price is the maximum price of the asset on the most recent $w$ periods. The peak prices of different assets are calculated as $\hat{\mathbf{p}}_{t+1}^{(i)} = \max_{0 \leqslant k \leqslant w-1} \mathbf{p}_{t-k}^{(i)}$ $(i = 1, 2, \ldots, m)$ and the resulted price relative prediction is calculated as $\hat{\mathbf{x}}_{t+1}^{(i)} = \frac{\hat{\mathbf{p}}_{t+1}^{(i)}}{\mathbf{p}_t^{(i)}}$. We evaluate the gap between peak price and the actual price of single asset by proposed RBF as

$$\phi_i\left(\hat{\mathbf{x}}_{t-w+1}^{(i)}, .., \hat{\mathbf{x}}_t^{(i)}\right) = \exp\left(\frac{-\sum_{j=t-w+1}^{t}\left(\hat{\mathbf{x}}_j^{(i)} - \mathbf{x}_j^{(i)}\right)^2}{2\sigma_i^2}\right), \tag{2}$$

where the real relative price of $i$th asset $\left[\mathbf{x}_{t-w+1}^{(i)}, .., \mathbf{x}_t^{(i)}\right]$ is the center and $\sigma_i$ is the scale parameter. The RBF quantifies the similarity between the peak price and the actual price of the single asset in the recent time window. If relative peak price is closer to relative actual price, then the corresponding $\phi_i$ will be larger and peak price gains more influence in the following prediction. The center switches as time $t$ of real relative price goes on and keeps to newest asset status.

Then for asset $i$, the adaptive peak price is calculated as $\phi_i \max_{0 \leqslant k \leqslant w-1} \mathbf{p}_{t-k}^{(i)}$. An example is shown in Fig. 1, when the asset price depreciates continuously (time 3 to 13), RBF captures the depreciation and adjusts the peak price downward. When the continuous depreciation ends (time 13 to 18), APP gets closer and closer to the peak price. The relative distance from the current price vector adaptive peak price implies the increasing potential of the assets. Thus, we combine RBFs of different assets and form the expected return as follows:

$$\hat{\mathbf{X}}_{t+1} = [\phi_1 \hat{\mathbf{x}}_{t+1}^{(1)}, \phi_2 \hat{\mathbf{x}}_{t+1}^{(2)}, ..., \phi_m \hat{\mathbf{x}}_{t+1}^{(m)}]^\top. \tag{3}$$

We use $\hat{\mathbf{X}}_{t+1}$ as the price information input for the APPLU.

### 3.2 Portfolio Selection Model

To achieve higher cumulative wealth and reduce transaction costs, our portfolio selection model is designed with the following objectives: first, we should maximize $\mathbf{b}^\top \hat{\mathbf{X}}_{t+1}$, which is the increasing potential of the whole portfolio. Denote

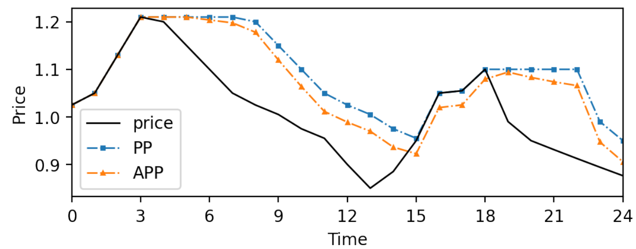



Fig. 1: An example of APP ($\sigma_i^2 = 3.5, w = 5$). "Price" denotes the actual asset price, "PP" denotes the peak price of the actual asset price.

$\boldsymbol{\varphi}_t = -\hat{\mathbf{X}}_{t+1}$, then we can change the maximization to a minimization. Second, drawing on [18,14], we adopt an $l^1$-regularization term and a self-financing constraint simultaneously to concentrate the portfolio on a few assets. Third, we introduce an $l^2$-regularization term to impose the consistency of portfolio weights before and after each rebalance. Then the proposed portfolio selection model is formed as

$$\mathbf{b}_{t+1} = \min_{\mathbf{b}} \mathbf{b}^\top \boldsymbol{\varphi}_t + \lambda_1 \|\mathbf{b}\|_1 + \lambda_2 \|\mathbf{b} - \mathbf{b}_t\|_2, \quad \text{s.t. } \mathbf{1}^\top \mathbf{b} = 1, \tag{4}$$

where the parameter $\lambda_1$ controls the sparsity of the portfolio, and $\lambda_2$ controls the change of the portfolio.

Different from previous studies [5,4,18] which use $l^2$-norm squared $\|\mathbf{b} - \mathbf{b}_t\|_2^2$, we adopt $l^2$-norm itself $\|\mathbf{b} - \mathbf{b}_t\|_2$ to reduce turnover rates and transaction costs. The $l^2$-norm allows the portfolio to be more responsive to the new predictions encapsulated in $\boldsymbol{\varphi}_t$. This means that the portfolio can adjust more fluidly to the new information rather than being overly anchored to the past weights $\mathbf{b}_t$. In contrast, due to the presence of square calculations, the $l^2$-norm squared encourages small portfolio changes but excessively penalizes large portfolio changes. This characteristic, often overlooked, can significantly impact the long-term performance of an investment portfolio.

Note that the above regularization term in Eq. (4) can be linked to existing learning frameworks in both portfolio selection and machine learning. For instance, if we omit $\mathbf{b}_t$, the above regularization term becomes the so called $l^{1,2}$-norm ($\lambda_1 \|\mathbf{b}\|_1 + \lambda_2 \|\mathbf{b}\|_2$) [20] to prevents extreme positions. A similar regularization framework has also been combined with linear regression to derive sparse-group lasso.[19].

### 3.3 Computation

We have developed a comprehensive computational approach for APPLU. The APPLU algorithm, detailed in Algorithm 1, outlines the procedure followed each trading period to update portfolio weights in response to fresh market data.

**Algorithm 1** APPLU Algorithm

**Input:** Given parameters $w$, $\lambda_1$, $\lambda_2$, $\{\sigma_i^2\}_{i=1}^m$, the actual price relatives $\{\mathbf{x}_{t-k}\}_{k=0}^{w-1}$ and the peak price relatives $\{\hat{\mathbf{x}}_{t-k}\}_{k=-1}^{w-1}$ in recent time window, the current portfolio $\mathbf{b}_t$.

1: Calculate the RBFs $\{\phi_i\}_{i=1}^m$ by Eq. (2).

2: Calculate the APP vector $\hat{\mathbf{X}}_{t+1}$ by Eq. (3) and adjust it to fit the minimization $\boldsymbol{\varphi}_t = -\hat{\mathbf{X}}_{t+1}$.

3: Solve Problem (4) and get solution $\mathbf{b}_{t+1}$.

4: Normalize: $\hat{\mathbf{b}}_{t+1} = \arg\min_{\mathbf{b}\in\Delta_d} \|\mathbf{b} - \mathbf{b}_{t+1}\|^2$

**Output:** The next portfolio $\hat{\mathbf{b}}_{t+1}$.

In step 2 of the algorithm, formulation (4) is identified as a convex optimization problem, primarily because the objective function comprises a sum of convex terms, and the constraint itself is convex. This allows for efficient resolution using CVXPY [6] and the open-source solver ECOS [7]. In step 4, the portfolio weights $\mathbf{b}_{t+1}$ are normalized [8] to ensure they represent a feasible portfolio for the upcoming trading period. Our backtest computations, conducted on a CPU AMD-3500x, demonstrate that each round of computation is completed in under 0.02 seconds, underscoring the model's suitability for real-time trading scenarios.

## 4 Experimental Results

In this section, we focus on the comparison studies. First, we introduce testing datasets, competing portfolio strategies and criteria of evaluation. Then, we experimentally tune the hyper parameters of APPLU and conduct ablation experiments. At last, we report and analyze the results of comparison studies, and also discuss about turnover and transaction costs.

### 4.1 Dataset

To ensure the reproducibility of our results and the fairness of our comparisons to existing algorithms, we conduct extensive experiments on seven public benchmark datasets: DJIA [1], TSE [1], NYSE(O) [2], NYSE(N) [16], MSCI [15], SP500 [1] and ETF23. They contain real-world daily *close price relatives* from diverse stock and index market, including the New York Stock Exchange (NYSE), the Toronto Stock Exchange (TSE), the MSCI World Index (MSCI), the Dow Jones Industrial Average (DJIA), the Standard and Poor's 500 (SP500) and the Chinese ETF market (ETF23). The detailed information about these datasets are listed in Table 1.

### 4.2 Competing Portfolio Strategies

For the purpose of comparison with APPLU, five state-of-the-art portfolio selection strategies and a benchmark algorithm have been selected. The benchmark

Table 1: Detailed Information of seven datasets

| Data set | Region | Time | Days | Assets |
|---|---|---|---|---|
| DJIA | US | 14/1/2001 - 14/1/2003 | 507 | 30 |
| TSE | CA | 4/1/1994 - 31/12/1998 | 1259 | 88 |
| NYSE(O) | US | 3/7/1962 - 31/12/1984 | 5651 | 36 |
| NYSE(N) | US | 1/1/1985 - 30/6/2010 | 6431 | 23 |
| SP500 | US | 2/1/1998 - 31/1/2003 | 1276 | 25 |
| MSCI | Global | 1/4/2006 - 31/3/2010 | 1043 | 24 |
| ETF23 | CN | 1/2/2021 - 1/10/2023 | 647 | 23 |

algorithm, namely uniform buy-and-hold (BAH), has been traditionally used as performance benchmark in the field of investment. The strategy invests equally in $m$ assets at the onset and maintains this allocation throughout, often considered as a market strategy leading to the production of a market index [15]. We use the recommended parameters in the original papers for the competing portfolio strategies as follows:

1. *OLMAR.* [15] It takes the moving average to predict the future price. The parameters are set as: $\epsilon = 10$.
2. *TCO1.* [17] It proposes a $l^1$-norm for transaction cost. The parameters are set as: $\lambda = 10\gamma$, $\eta = 10$.
3. *TCO2.* [17] The parameters are set as: $\lambda = 10\gamma$, $w = 4$, $\eta = 10$.
4. *SPOLC.* [12] The short-term portfolio optimization with loss control strategy with the window size $w = 5$ and the mixing parameter $\gamma = 0.025$.
5. *TCR.* [21] It improve price information and solving algorithm in TCO. TCR represents the TCR2 strategy in original paper since it has better performance. The parameters are set as: $\lambda = 10\gamma$, $\rho = 0.618$.

The parameters for APPLU are set as: $\lambda_1 = 1, \lambda_2 = 0.04$, and $\sigma_i^2 = 3.5$. In order to utilize RBFs, the APPLU requires at least $w$ days of history data. To ensure fairness, all strategies begin adopting their respective algorithm outputs for investment from the sixth day under the transaction cost rate of 0.5%.

### 4.3 Evaluation Criteria

We use four metrics to evaluate the investing performance of APPLU. Cumulative wealth and annualized return measure the total return of an investment strategy. Sharpe ratio and calmar ratio measure risk-adjusted returns.

*Cumulative Wealth* (CW) is utilized as the key evaluation metric for the investment performance of each portfolio selection algorithm, which is computed by Eq. (1).

*Annualized Percentage Yield* (APY) is a widely used metric for evaluating investment returns. It represents the average return of a strategy over the course of a year. APY is computed as $\mathrm{APY} = \mathrm{CW}^{1/y} - 1$, where $y$ represents the number of years according to $n$ trading days. In this study, all datasets consist of daily prices. Therefore, $y$ is calculated as $n$ divided by 252, which is the average number of annual trading days.

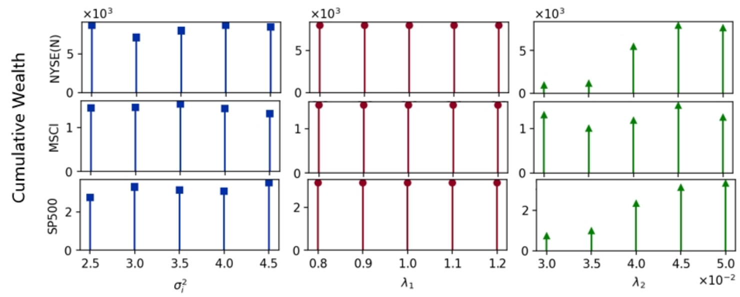


Fig. 2: Cumulative wealth with respect to different $\sigma_i^2$ (fix $\lambda_1 = 1, \lambda_2 = 0.04$), $\lambda_1$ (fix $\sigma_i^2 = 3.5, \lambda_2 = 0.04$) and $\lambda_2$ (fix $\sigma_i^2 = 3.5, \lambda_1 = 1$) on three datasets. Three columns share the same x-axis.

*Sharpe Ratio* (SR) serves as a widely utilized metric for evaluating risk-adjusted returns and is defined as SR $= \frac{\bar{r}_s - r_f}{\sigma(r_s)}$, where $r_f$ is the return of a risk-free asset and is set to 0 in this paper since we do not consider a risk-free asset, $\sigma(r_s)$ is the standard deviation of return $r_s$ estimated by the samples $r_{s,t}$ in $n$ trading period.

*Calmar Ratio* (CR) is a comparison of the average annual compound return and the maximum drawdown (MDD) risk, which is widely adopted in fund management. The calculation formula is CR=APY/MDD, where MDD $= \max_{t \in [1,T]} \frac{M_t - S_t}{M_t}$, $M_t = \max_{k \in [1,t]} S_k$.

*Turnover* is a measure of the cumulative change in wealth proportion vectors during the trading periods, and it is defined as Turnover$= \sum_{t=2}^{T} ||\mathbf{b}_t - \mathbf{b}_{t-1}||_1$. A high turnover generates more commissions on trades placed by a broker.

## 4.4 Parameter Setting

We conduct a comprehensive analysis of the parameter setting of APPLU through experiments using benchmark datasets. Similar to previous studies [15,12,10], we adopt an empirical method to determine the parameters based on their CWs computed by Eq. (1). The value of $w$ is set to 5, which aligns with previous research [15,12,10] and is a commonly used time window size in stock and futures investment as it reflects the recent financial environment. With regards to the parameters $\lambda_1$, $\lambda_2$ and $\sigma_i^2$, an initial approximation is made followed by fine-tuning in incremental steps.

We first fix $\lambda_1 = 1, \lambda_2 = 0.04$ and change $\sigma_i^2$ between 2.5 to 4.5 and see the cumulative wealth in different datasets. The first column in Fig. 2 show that APPLU is stable at $\sigma_i^2 = 3.5$. Then we keep fine tune the $\lambda_1$ and $\lambda_2$ by fixing the other two parameters. Therefore, by fixing two parameters and tuning the remaining one, the parameters $\lambda_1 = 1, \lambda_2 = 0.04$, and $\sigma_i^2 = 3.5$ are determined as the optimal values for APPLU.

Table 2: Performance of algorithms under the transaction cost rate of 0.5%.

| Metric | Algorithm | DJIA | TSE | NYSE(O) | NYSE(N) | SP500 | MSCI | ETF23 |
|---|---|---|---|---|---|---|---|---|
| CW | BAH | 0.78 | 1.56 | 13.8 | 18.25 | 1.39 | 0.89 | 0.87 |
| | OLMAR | 0.44 | 1.08 | 3.89e+08 | 0.82 | 0.28 | 0.44 | 0.16 |
| | TCO1 | 0.42 | 1.95 | 8.45e+07 | 32.11 | 0.22 | 1.46 | 0.28 |
| | TCO2 | 0.95 | 9.78 | 5.35e+08 | 3.63e+03 | 1.69 | 1.48 | 0.44 |
| | SPOLC | 0.72 | 5.63 | 4.05e+05 | 0.212 | 1.53 | 0.37 | 0.14 |
| | TCR | 1.07 | 3.61 | 1.18e+10 | 4.52e+03 | 2.17 | 1.51 | 0.59 |
| | APPLU | **1.70** | **15.70** | **3.28e+11** | **7.98e+03** | **3.14** | **1.53** | **1.07** |
| APY | BAH | -0.115 | 0.093 | 0.124 | 0.121 | 0.068 | -0.027 | -0.054 |
| | OLMAR | -0.332 | 0.015 | 1.416 | -0.008 | -0.223 | -0.178 | -0.516 |
| | TCO1 | -0.353 | 0.143 | 1.257 | 0.146 | -0.261 | 0.097 | -0.391 |
| | TCO2 | -0.027 | 0.579 | 1.45 | 0.379 | 0.109 | 0.1 | -0.276 |
| | SPOLC | -0.150 | 0.413 | 0.778 | -0.059 | 0.087 | -0.215 | -0.542 |
| | TCR | 0.036 | 0.293 | 1.813 | 0.394 | 0.165 | 0.103 | -0.185 |
| | APPLU | **0.301** | **0.735** | **2.263** | **0.422** | **0.253** | **0.109** | **0.026** |
| SR | BAH | -0.024 | 0.047 | 0.054 | 0.046 | 0.025 | 0.001 | -0.013 |
| | OLMAR | -0.036 | 0.028 | 0.118 | 0.016 | -0.014 | -0.02 | -0.126 |
| | TCO1 | -0.06 | 0.033 | 0.136 | 0.033 | -0.034 | 0.028 | -0.107 |
| | TCO2 | 0.009 | 0.059 | 0.139 | 0.056 | 0.028 | 0.028 | -0.058 |
| | SPOLC | -0.008 | 0.0504 | 0.091 | 0.0053 | 0.026 | -0.029 | -0.153 |
| | TCR | 0.021 | 0.045 | 0.132 | 0.056 | 0.0346 | 0.028 | -0.023 |
| | APPLU | **0.049** | **0.065** | **0.148** | **0.057** | **0.043** | **0.029** | **0.016** |
| CR | BAH | -0.299 | 0.309 | 0.298 | 0.225 | 0.148 | -0.042 | -0.189 |
| | OLMAR | -0.432 | 0.016 | 2.36 | -0.008 | -0.293 | -0.225 | -0.604 |
| | TCO1 | -0.5 | 0.169 | 2.796 | 0.149 | -0.323 | 0.17 | -0.533 |
| | TCO2 | -0.061 | 0.7 | 3.453 | 0.39 | 0.252 | 0.167 | -0.463 |
| | SPOLC | -0.218 | 0.536 | 1.250 | -0.060 | 0.164 | -0.267 | -0.621 |
| | TCR | 0.077 | 0.322 | 3.344 | 0.412 | 0.295 | 0.190 | -0.340 |
| | APPLU | **0.686** | **0.919** | **4.756** | **0.430** | **0.465** | **0.192** | **0.068** |

Table 3: Ablation experiment results. CWs of algorithms under the transaction cost rate of 0.5%.

| Algorithm | DJIA | TSE | NYSE(O) | NYSE(N) | SP500 | MSCI | ETF23 |
|---|---|---|---|---|---|---|---|
| APPLU-PP | 1.45 | 11.1 | 5.36e+10 | 2.14e+03 | 2.61 | 1.16 | 0.99 |
| APPLU-squared | 0.99 | 14.8 | 5.13e+09 | 6.23e+02 | 1.78 | 0.90 | 0.46 |
| APPLU | **1.70** | **15.70** | **3.28e+11** | **7.98e+03** | **3.14** | **1.53** | **1.07** |

### 4.5 Ablation Experiment

To demonstrate the effectiveness of our method, we conducted two ablation experiments to observe the cumulative wealth performance of the model across various datasets. Firstly, to verify the improvement effect of RBFs on peak prices, we omitted the RBFs from the Eq. (3), resulting in a version named APPLU-PP. Furthermore, to validate the superiority of unsquared $l^2$-norm regularization over squared $l^2$-norm regularization, we replaced the $\|\mathbf{b} - \mathbf{b}_t\|_2$ term in the Eq. (4) with squared $l^2$-norm $\|\mathbf{b} - \mathbf{b}_t\|_2^2$, yielding a version named APPLU-squared.

The ablation study results in Table 3 reveal the APPLU significantly outperforms its variants across various datasets. APPLU-PP and APPLU-squared demonstrate inferior performance, underscoring the critical contributions of RBFs for capturing market dynamics and unsquared $l^2$-norm for flexible portfolio adjustments. The superior cumulative wealths achieved by APPLU across all

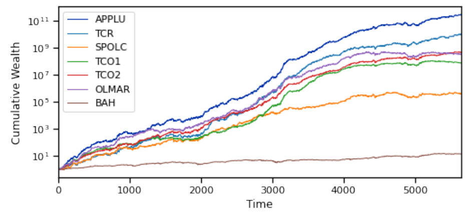



Fig. 3: CW of different algorithms during the entire investments in NYSE(O).

datasets confirm the effectiveness of combining these techniques for short-term portfolio optimization under transaction costs.

### 4.6 Comparison Studies

Table 2 provides a comprehensive comparison of our APPLU method against various established portfolio strategies across multiple real-world datasets, under a transaction cost scenario of 0.5%.

APPLU significantly outperformed other strategies in terms of CW, achieving the highest values in all datasets. Cumulative wealth plots on NYSE(O) are presented in Fig. 3 to provide a visual representation. This showcases APPLU's robustness and its capability to generate superior wealth returns, a critical factor in SPO. In the APY metric, APPLU's performance was exemplary, leading in all datasets. This highlights its consistent ability to generate effective returns across different market conditions, a testament to its adaptability.

The SR results further underscore APPLU's superior risk-adjusted return capabilities. Leading in this metric across all datasets, APPLU demonstrates not only its profitability but also its efficiency in managing investment risks, an essential aspect of portfolio optimization. APPLU also exhibited strong performance in terms of CR, again outperforming other strategies in most datasets. This indicates APPLU's proficient risk management in relation to the returns it generates, balancing return with maximum drawdown risks effectively.

Overall, the empirical evidence suggests that APPLU is a robust and effective strategy for SPO, capable of outperforming established strategies across key performance indicators.

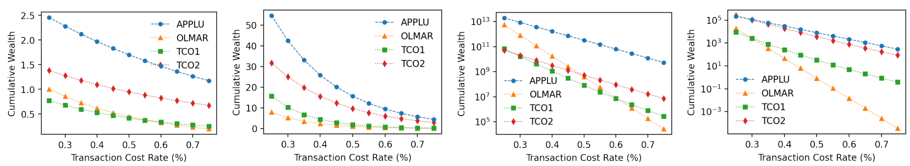
### 4.7 Turnover and Transaction Costs

As depicted in Table 4, APPLU demonstrates superior performance across the majority of datasets in terms of turnover rates. Notably, APPLU records the lowest turnover in all datasets except TSE, which ranks the second. It demonstrates the effectiveness of our proposed model in the turnover control and leads to lower transaction costs.

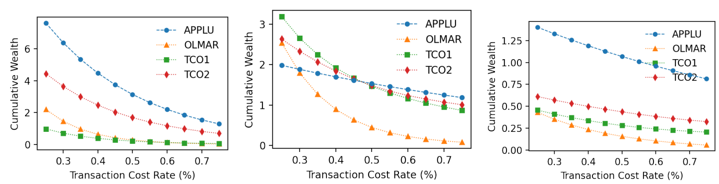
To assess the practicality of the portfolio selection algorithms, we perform experiments on cumulative wealth while changing the transaction cost rate $r$ between 0.25% and 0.75%. The findings, displayed in Fig. 4, indicate that APPLU can strongly withstand a range of reasonable transaction costs. Additionally, APPLU is comparable to other state-of-the-art algorithms in the most datasets. This demonstrates that APPLU is a capable algorithm for managing transaction costs, making it suitable for real-world financial environments.

Table 4: Turnover of algorithms under the transaction cost rate of 0.5%.

| Metric | Algorithm | DJIA | TSE | NYSE(O) | NYSE(N) | SP500 | MSCI | ETF23 |
|---|---|---|---|---|---|---|---|---|
| | OLMAR | 322.6 | 318 | 339.4 | 313.6 | 325.1 | 336.2 | 312.8 |
| | TCO1 | 260.4 | 345.4 | 263.4 | 205.3 | 248.8 | 155.6 | 194.8 |
| | TCO2 | 154.6 | 189.9 | 158.1 | 131.2 | 150.5 | 98.1 | 120.5 |
| Turnover | SPOLC | 281.5 | 274.1 | 289.8 | 266.5 | 277.2 | 296 | 285.3 |
| | TCR | 150.9 | **153.1** | 149.1 | 105.5 | 141.3 | 60.7 | 100.9 |
| | APPLU | **145.3** | 198.8 | **144.9** | **103.2** | **139.3** | **49.5** | **84.2** |




(a) DJIA (b) TSE (c) NYSE(O) (d) NYSE(N)




(e) SP500 (f) MSCI (g) ETF23

Fig. 4: Scalability of transaction cost in terms of cumulative wealth.

## 5 Conclusion

In this paper, we introduced Adaptive Peak Price with Lazy Updates (APPLU), a novel strategy for short-term portfolio optimization. APPLU stands out by dynamically adjusting asset allocations using radial basis functions and new lazy update approach. Our extensive experiments across diverse real-world datasets demonstrated that APPLU surpasses existing short-term portfolio optimization systems in key performance metrics, including cumulative wealth and sharpe ratio. APPLU's low turnover rates significantly reduce transaction costs, making it suitable for practical financial scenarios.

**Acknowledgements.** This work was funded by the Stabilization Support Plan for Shenzhen Higher Education Institutions, grant number 20200812165210001.

## References

1. Borodin, A., El-Yaniv, R., Gogan, V.: Can we learn to beat the best stock. Advances in Neural Information Processing Systems **16** (2003)

2. Cover, T.M.: Universal portfolios. Mathematical finance **1**(1), 1–29 (1991)
3. Dai, H.L., Huang, C.Y., Dai, H.M., Lai, F.T., Lv, X.T., Adnan, R.M.: A novel adjusted learning algorithm for online portfolio selection using peak price tracking approach. Decision Analytics Journal p. 100256 (2023)
4. Das, P., Johnson, N., Banerjee, A.: Online lazy updates for portfolio selection with transaction costs. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 27, pp. 202–208 (2013)
5. Das, P., Johnson, N., Banerjee, A.: Online portfolio selection with group sparsity. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 28 (2014)
6. Diamond, S., Boyd, S.: Cvxpy: A python-embedded modeling language for convex optimization. JOURNAL OF MACHINE LEARNING RESEARCH **17** (2016)
7. Domahidi, A., Chu, E., Boyd, S.: Ecos: An socp solver for embedded systems. In: 2013 European control conference (ECC). pp. 3071–3076. IEEE (2013)
8. Duchi, J., Shalev-Shwartz, S., Singer, Y., Chandra, T.: Efficient projections onto the l 1-ball for learning in high dimensions. In: Proceedings of the 25th international conference on Machine learning. pp. 272–279 (2008)
9. Glaubitz, J., Nordström, J., Öffner, P.: Energy-stable global radial basis function methods on summation-by-parts form. Journal of Scientific Computing **98**(1), 30 (2024)
10. Huang, D., Zhou, J., Li, B., Hoi, S.C., Zhou, S.: Robust median reversion strategy for online portfolio selection. IEEE Transactions on Knowledge and Data Engineering **28**(9), 2480–2493 (2016)
11. Lai, Z.R., Dai, D.Q., Ren, C.X., Huang, K.K.: Radial basis functions with adaptive input and composite trend representation for portfolio selection. IEEE Transactions on Neural Networks and Learning Systems **29**(12), 6214–6226 (2018)
12. Lai, Z.R., Tan, L., Wu, X., Fang, L.: Loss control with rank-one covariance estimate for short-term portfolio optimization. The Journal of Machine Learning Research **21**(1), 3815–3851 (2020)
13. Lai, Z.R., Yang, H.: A survey on gaps between mean-variance approach and exponential growth rate approach for portfolio optimization. ACM Computing Surveys (CSUR) **55**(2), 1–36 (2022)
14. Lai, Z.R., Yang, P.Y., Fang, L., Wu, X.: Short-term sparse portfolio optimization based on alternating direction method of multipliers. The Journal of Machine Learning Research **19**(1), 2547–2574 (2018)
15. Li, B., Hoi, S.C., Sahoo, D., Liu, Z.Y.: Moving average reversion strategy for on-line portfolio selection. Artificial Intelligence **222**, 104–123 (2015)
16. Li, B., Hoi, S.C., Zhao, P., Gopalkrishnan, V.: Confidence weighted mean reversion strategy for online portfolio selection. ACM Transactions on Knowledge Discovery from Data (TKDD) **7**(1), 1–38 (2013)
17. Li, B., Wang, J., Huang, D., Hoi, S.C.: Transaction cost optimization for online portfolio selection. Quantitative Finance **18**(8), 1411–1424 (2018)
18. Shen, W., Wang, J., Ma, S.: Doubly regularized portfolio with risk minimization. In: proceedings of the AAAI Conference on Artificial Intelligence. vol. 28 (2014)
19. Simon, N., Friedman, J., Hastie, T., Tibshirani, R.: A sparse-group lasso. Journal of computational and graphical statistics **22**(2), 231–245 (2013)
20. Zhao, H., Kong, L., Qi, H.D.: Optimal portfolio selections via $l^{1,2}$-norm regularization. Computational Optimization and Applications **80**(3), 853–881 (2021)
21. Zhou, H., Yao, X., Cai, S., Zhang, N.: Transaction cost regularization for online portfolio selection. In: 2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA). pp. 288–295. IEEE (2022)