Journal of Industrial and Management Optimization

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

JIMO, 22(6): 2755–2783.
DOI:10.3934/jimo.2026101
Received: 23 February 2026
Revised: 26 April 2026
Accepted: 08 May 2026
Published: 20 May 2026

Research article

Cardinality-constrained maximal predictability portfolios with an ℓ_2 regularization

Katsuhiro Tanaka* **and Rei Yamamoto**

Department of Industrial and Systems Engineering, Keio University, 3-14-1, Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa 223-8522, Japan

* **Correspondence:** Email: katsuhiro-tanaka@keio.jp; Tel: +81-45-566-1774.

Abstract: The maximal predictability portfolio (MPP) is a portfolio optimization framework that explicitly incorporates return predictability into the objective function. While MPP has been shown to achieve favorable empirical performance, it is prone to overfitting when using rich sets of candidate assets and factors. To address this issue, we study a cardinality-constrained MPP formulation with an ℓ_2 regularization term and investigate how regularization reshapes the structure and out-of-sample behavior of MPP. The resulting formulation leads to a challenging optimization problem involving binary variables and nonconvex constraints. We show that, through an appropriate bilevel reformulation, the problem can be handled within the cutting-plane algorithm (CPA) framework. In particular, by exploiting a globally optimal solution of the primal lower-level problem, we derive an equivalent convex formulation that preserves the finite termination and ε-optimality guarantees of CPA. We also derive a subgradient expression tailored to the MPP structure, which enables the construction of valid cutting planes despite the nonconvexity of the original formulation. For practical large-scale computation, we further incorporate normalized linearization as a heuristic accelerator for the lower-level problem, which improves tractability. However, the exact finite termination and ε-optimality guarantees of CPA no longer directly apply. Numerical experiments demonstrate that the proposed approach, combined with normalized linearization, can efficiently compute high-quality solutions for large-scale instances where exact methods become computationally impractical. The results further illustrate that moderate ℓ_2 regularization improves out-of-sample predictability, whereas excessive regularization diminishes the predictive structure of MPP. Overall, our findings highlight the role of ℓ_2 regularization in balancing predictability, stability, and practical feasibility within the MPP framework.

Keywords: portfolio optimization; maximal predictability portfolio; ℓ_2 regularization; cardinality constraint; nonconvex mixed-integer quadratically constrained quadratic programming
Mathematics Subject Classification: 90C11, 90C26, 90C06

Abbreviations: MPP: Maximal predictability portfolio; CPA: Cutting-plane algorithm; NL: Normalized linearization; TS: Two-step approach

1. Introduction

The maximal predictability portfolio (MPP) was proposed by [1] as a framework that constructs a portfolio by maximizing the coefficient of determination of returns obtained from factor-based regression models. The objective of MPP is to enhance return predictability, namely, to increase the probability that portfolio returns exceed a prespecified level. Rather than evaluating portfolios solely through distributional moments of returns, MPP explicitly incorporates a measure of return predictability into the objective function.

An important feature of MPP is its ability to utilize conditional return models, making it applicable under asymmetric return distributions with skewness and fat tails [2]. Empirical studies have shown that improvements in return predictability achieved through conditional return models are often associated with favorable portfolio performance [3]. In addition, the effectiveness of MPP has been documented across a variety of empirical settings [2, 4–6]. Accordingly, MPP has been recognized as a portfolio optimization framework that explicitly emphasizes predictability.

To enhance predictability, MPP often relies on rich sets of candidate assets and factors, which can improve in-sample fit but may deteriorate out-of-sample performance [7, 8]. The MPP formulation is designed to enhance return predictability and does not directly control risk measures representing portfolio uncertainty, such as variance. Therefore, portfolio weights may become concentrated in a small subset of assets that exhibit high in-sample predictability. Consequently, although the resulting portfolio may show strong in-sample predictive performance, it may also suffer from overfitting, which can deteriorate out-of-sample investment performance.

A standard remedy for overfitting is ℓ_2 regularization [9]. In portfolio optimization, ℓ_2 regularization is known to stabilize out-of-sample performance by shrinking portfolio weights toward more balanced allocations [10–12]. Therefore, we introduce an ℓ_2 regularization term to suppress extreme portfolio weights and improve portfolio stability.

However, strong ℓ_2 regularization may induce excessive diversification, leading to large portfolios with high monitoring and transaction costs. To address this practical issue, a cardinality constraint was introduced to explicitly limit the number of invested assets [13]. The cardinality constraint and the ℓ_2 regularization term are not intended to serve the same purpose. The cardinality constraint controls the number of invested assets, whereas the ℓ_2 regularization term controls the concentration of weights within the selected asset set. This distinction is particularly important in the MPP setting, because the cardinality constraint alone allows highly uneven allocations and may even intensify concentration on assets with strong in-sample predictability. Therefore, the ℓ_2 regularization term remains necessary as a complementary device for suppressing extreme weights and improving portfolio stability.

In this study, we extend the MPP formulation by incorporating an ℓ_2 regularization term and a cardinality constraint to mitigate overfitting and improve predictive stability. Our objective is to investigate how ℓ_2 regularization reshapes the structure and out-of-sample behavior of MPP, as opposed to conducting a performance comparison with alternative portfolio optimization models. The resulting formulation balances diversification and sparsity, but leads to a challenging optimization problem involving a nonconvex constraint and binary variables. While heuristic algorithms tailored to MPP, such as those proposed by [6], have been shown to efficiently produce high-quality solutions, they do not guarantee solution optimality. As the objective of this study is to examine how ℓ_2 regularization affects the structure and predictive behavior of MPP, an optimization framework with explicit ε-optimality guarantees is required.

This consideration motivates the use of the cutting-plane algorithm (CPA) proposed by [14], which provides theoretical guarantees of ε-optimality for cardinality-constrained portfolio optimization problems with ℓ_2 regularization. Its scalability has been demonstrated by solving large-scale regularized portfolio problems with over 1000 assets. Subsequent studies further extended the CPA framework to other convex optimization models, highlighting its broad applicability [15–17]. Moreover, CPA has also been successfully applied beyond portfolio optimization problems [18, 19], motivating its consideration in the context of MPP.

The CPA framework alternates between an upper-level problem with binary variables and a lower-level problem with continuous variables. In contrast, MPP involves a nonconvex lower-level problem, rendering the conventional CPA inapplicable. To address this challenge, we decompose the computation of the lower-level problem into (i) a nonconvex primal quadratic program and (ii) an equivalent convex reformulation that exploits the primal solution, both of which together allow the nonconvex structure to be handled within a cutting-plane framework while retaining explicit ε-optimality and finite convergence guarantees.

To efficiently solve the nonconvex primal problem, we adopt the normalized linearization (NL) algorithm proposed by [20], which has been shown to exhibit good scalability for large-scale MPP problems while producing high-quality approximate solutions. By integrating NL into the computation of the lower-level problem, the proposed approach improves computational tractability while sacrificing theoretical optimality guarantees. Nevertheless, consistent with prior studies, NL yields high-quality approximate solutions, and the resulting CPA solutions are empirically close to the optimal solutions.

The main contributions of this study are summarized as follows:

- We extend the MPP formulation by incorporating an ℓ_2 regularization term and a cardinality constraint, resulting in a nonconvex mixed-integer quadratically constrained quadratic programming (MIQCQP) problem that balances predictability and sparsity.
- To enable the use of the existing CPA framework, we reformulate the nonconvex lower-level problem into an equivalent convex formulation. This reformulation allows MPP to be handled within CPA without altering its fundamental framework.
- We derive a dual formulation and a subgradient expression tailored to the MPP structure, which makes it possible to construct valid cutting planes. Computational efficiency is further enhanced by integrating NL [20].
- Empirical results show that moderate ℓ_2 regularization improves out-of-sample return predictability and investment performance, whereas excessive regularization weakens the predictive advantage of MPP.

The remainder of this article is organized as follows. Section 2 formulates the MPP problem with an ℓ_2 regularization term and a cardinality constraint, and presents a dual reformulation required to apply the CPA framework. Section 3 describes the proposed CPA for nonconvex MPP problems and introduces the NL algorithm for accelerating the lower-level computations. Section 4 outlines the benchmark heuristic algorithm and the computational settings used in the experiments. Section 5 reports the numerical results, including algorithmic performance and portfolio investment outcomes. Finally, Section 6 concludes the paper and discusses directions for future research.

2. Problem formulation

This section introduces the basic notation and formulates the problem.

2.1. MPP

We present the MPP problem following [20]. Let N, T, and K be the number of candidate stocks, periods, and factors, respectively, and let $\mathcal{N} := \{1, \cdots, N\}$, $\mathcal{T} := \{1, \cdots, T\}$, $\mathcal{K} := \{1, \cdots, K\}$. The return of asset n in period t, $R_{n,t}$ $(n \in \mathcal{N},\ t \in \mathcal{T})$, can be expressed as a multifactor regression model as follows:

$$R_{n,t} \ = \ \beta_{n,0} + \sum_{k=1}^{K} \beta_{n,k} f_{k,t-1} \ + \ \varepsilon_{n,t} \ (n \in \mathcal{N}, t \in \mathcal{T}), \tag{2.1}$$

where $f_{k,t-1}$ $(k \in \mathcal{K}, t \in \mathcal{T})$ is factor k in period $t-1$, $\beta_{n,0}$ $(n \in \mathcal{N})$ is an intercept of asset n, $\beta_{n,k}$ $(n \in \mathcal{N},\ k \in \mathcal{K})$ represents the coefficient on factor k for asset n, and $\varepsilon_{n,t}$ $(n \in \mathcal{N},\ t \in \mathcal{T})$ is the residual of the asset n in period t*. Throughout the paper, we condition on realized factor values and use this model as a regression-based representation of observed returns.

We assume that $f_{k,t-1}$ and $\varepsilon_{n,t}$ are independent and that $E[\boldsymbol{\varepsilon}_n] = \mathbf{0}$ $(n \in \mathcal{N})$, where $\mathbf{0}$ denotes the all-zeros vector of appropriate dimension and $\boldsymbol{\varepsilon}_n := (\varepsilon_{n,1}, \ldots, \varepsilon_{n,T})^\top$ $(n \in \mathcal{N})$. $\beta_{n,0}$ $(n \in \mathcal{N})$ and $\beta_{n,k}$ $(n \in \mathcal{N}, k \in \mathcal{K})$ can be estimated using an ordinary least squares method that minimizes the sum of the squares of residuals. $\tilde{r}_n$ denotes the expected return of asset n as in (2.1) and can be expressed as $\tilde{r}_n \ = \ \beta_{n,0} \ + \ \sum_{k=1}^{K} \beta_{n,k} f_{k,T}$ $(n \in \mathcal{N})$. Additionally, $\tilde{\boldsymbol{r}}$ denotes a vector of $\tilde{r}_n$ $(n \in \mathcal{N})$.

Next, we define $\boldsymbol{P} \in \mathbb{R}^{N \times N}$ for a variance–covariance matrix of the return vector as follows:

$$\boldsymbol{P} := \ \frac{\boldsymbol{R}^\top \boldsymbol{R}}{T}, \ \boldsymbol{R}^\top : = \begin{pmatrix} R_{1,1} - r_1 & \cdots & R_{1,T} - r_1 \\ \vdots & & \vdots \\ R_{N,1} - r_N & \cdots & R_{N,T} - r_N \end{pmatrix}, \ r_n := \ \frac{\sum_{t=1}^{T} R_{n,t}}{T} \ (n \in \mathcal{N}). \tag{2.2}$$

In addition, we assume that $\boldsymbol{P}$ is a positive definite matrix.

We define $\boldsymbol{Q} \in \mathbb{R}^{N \times N}$ for a variance–covariance matrix of the residual vector as follows:

$$\boldsymbol{Q} := \ \frac{\boldsymbol{E}^\top \boldsymbol{E}}{T}, \ \boldsymbol{E}^\top := \begin{pmatrix} \varepsilon_{1,1} & \cdots & \varepsilon_{1,T} \\ \vdots & & \vdots \\ \varepsilon_{N,1} & \cdots & \varepsilon_{N,T} \end{pmatrix}. \tag{2.3}$$

Given the weight of each asset in a portfolio as $\boldsymbol{x} \in \mathbb{R}^N$, we define $R^2(\boldsymbol{x})$, the coefficient of determination as follows:

$$R^2(\boldsymbol{x}) := \ 1 - \frac{\boldsymbol{x}^\top \boldsymbol{Q} \boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x}}. \tag{2.4}$$

*The lagged factors $f_{k,t-1}$ are used to predict the next-period return $R_{n,t}$, reflecting the factor at time $t-1$.

We define the feasible investment set $\mathcal{X}$, as follows:

$$\mathcal{X} := \left\{ \boldsymbol{x} \in \mathbb{R}^N \mid \sum_{n=1}^{N} \tilde{r}_n x_n \geq \rho, \sum_{n=1}^{N} x_n = 1, 0 \leq x_n \leq 1 \ (n \in \mathcal{N}) \right\}, \tag{2.5}$$

where ρ denotes a user-defined parameter representing the target portfolio return.

The MPP problem is written as follows:

$$\left| \begin{array}{ll} \min\limits_{\boldsymbol{x}} & \dfrac{\boldsymbol{x}^\top \boldsymbol{Q} \boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x}} \\ \text{s.t.} & \boldsymbol{x} \in \mathcal{X}. \end{array} \right. \tag{2.6}$$

2.2. *Nonconvex QCQP reformulation*

This subsection reformulates Problem (2.6) as a nonconvex QCQP.

The objective function of Problem (2.6) can be rewritten by applying (2.2) and (2.3). Let $\eta := 1/\sqrt{\boldsymbol{x}^\top \boldsymbol{R}^\top \boldsymbol{R} \boldsymbol{x}}$, which is positive. Then, we define $\boldsymbol{y} := \boldsymbol{x}\eta$ and introduce the following set, $\mathcal{Y}$:

$$\mathcal{Y} := \left\{ \boldsymbol{y} \mid \sum_{n=1}^{N} \tilde{r}_n y_n \geq \rho \sum_{n=1}^{N} y_n, \ M \geq y_n \geq 0 \ (n \in \mathcal{N}) \right\}, \tag{2.7}$$

where $\sum_{n=1}^{N} y_n = \eta$ and M is sufficiently large so that the upper-bound constraints are inactive at an optimal solution. We reformulate Problem (2.6) as follows:

$$\left| \begin{array}{ll} \min\limits_{\boldsymbol{y}} & \boldsymbol{y}^\top \boldsymbol{E}^\top \boldsymbol{E} \boldsymbol{y} \\ \text{s.t.} & \boldsymbol{y}^\top \boldsymbol{R}^\top \boldsymbol{R} \boldsymbol{y} = 1, \\ & \boldsymbol{y} \in \mathcal{Y}. \end{array} \right. \tag{2.8}$$

Next, we incorporate vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^T$. Referring to Section 3 of [20], we reformulate Problem (2.8) as follows:

$$\left| \begin{array}{ll} \min\limits_{\boldsymbol{y},\boldsymbol{u},\boldsymbol{v}} & \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y}, \ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y}, \ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\ & \boldsymbol{y} \in \mathcal{Y}. \end{array} \right. \tag{2.9}$$

Thus, a nonconvex QCQP with the constraint $\boldsymbol{u}^\top \boldsymbol{u} = 1$ is obtained in Problem (2.9).

2.3. *ℓ_2 regularization term*

This subsection introduces an ℓ_2 regularization term to Problem (2.9). We impose an ℓ_2-norm constraint $\|\boldsymbol{x}\|_2 := \sqrt{\boldsymbol{x}^\top \boldsymbol{x}}$ in Problem (2.9) as follows:

$$\left| \begin{array}{ll} \min\limits_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{u},\boldsymbol{v}} & \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y}, \ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y}, \ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\ & \boldsymbol{y} \in \mathcal{Y}, \\ & \boldsymbol{y} = \boldsymbol{x}\eta, \|\boldsymbol{x}\|_2 \leq C_1, \end{array} \right. \tag{2.10}$$

where C_1 is a positive user-defined parameter. According to Proposition 1 in [10], the norm constraint is equivalent to a robust inequality independent of the objective function or other constraints.

Moreover, under the long-only setting ($\boldsymbol{x} \geq 0$) with an equality budget constraint ($\sum_{n=1}^{N} x_n = 1$), imposing an ℓ_1 regularization does not alter the optimal solution, as it reduces to a constant. Therefore, we focus on ℓ_2 regularization in this study.

Decreasing C_1 yields a portfolio closer to an equally weighted portfolio, as indicated by [11, 12]. Without the ℓ_2-norm constraint, the portfolio tends to concentrate on a small number of assets†. Adding the ℓ_2-norm constraint is one way to avoid an unbalanced portfolio weight. This is because the ℓ_2-norm constraint promotes investment stock diversification, as reported by [13]‡. Diversification leads to the construction of a robust portfolio that is more likely to satisfy the expected-return constraint ($\sum_{n=1}^{N} \tilde{r}_n x_n \geq \rho$), although $R^2(\boldsymbol{x})$ decreases. However, when N is large, decreasing C_1 tends to induce excessive diversification, so that the portfolio ends up including almost all stocks.

In this study, we propose MPP with an ℓ_2 regularization term in the objective function from Problem (2.10), similar to a general ℓ_2 regularization approach to convex programming:

$$
\left|\begin{array}{ll}
\min\limits_{\boldsymbol{y},\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{x}^\top \boldsymbol{x} + \boldsymbol{v}^\top \boldsymbol{v} \\
\text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y},\ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y},\ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\
& \boldsymbol{y} \in \mathcal{Y}, \\
& \boldsymbol{y} = \boldsymbol{x}\eta,
\end{array}\right. \tag{2.11}
$$

where $\gamma(> 0)$ is a user-defined hyperparameter. It is difficult to search for the optimal solutions to Problem (2.11) because Problem (2.11) includes strong nonconvexity with the nonconvex constraint $\boldsymbol{u}^\top \boldsymbol{u} = 1$ as well as an additional nonconvex constraint $\boldsymbol{y} = \boldsymbol{x}\eta$. Therefore, we consider the following surrogate MPP problem by replacing the ℓ_2 regularization term from $\boldsymbol{x}$ to $\boldsymbol{y}$:

$$
\left|\begin{array}{ll}
\min\limits_{\boldsymbol{y},\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}^\top \boldsymbol{y} + \boldsymbol{v}^\top \boldsymbol{v} \\
\text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y},\ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y},\ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\
& \boldsymbol{y} \in \mathcal{Y}.
\end{array}\right. \tag{2.12}
$$

As Problem (2.12) can exclude the nonconvex constraint $\boldsymbol{y} = \boldsymbol{x}\eta$, we can obtain the solution in less time than Problem (2.11).

Then, Problem (2.12) is equal to the following problem derived from Problem (2.6):

$$
\left|\begin{array}{ll}
\min\limits_{\boldsymbol{x}} & \dfrac{\frac{1}{2\gamma}\boldsymbol{x}^\top \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x}} \\
\text{s.t.} & \boldsymbol{x} \in \mathcal{X}.
\end{array}\right. \tag{2.13}
$$

The numerator of the objective function in Problem (2.13) is equal to $\boldsymbol{x}^\top(\frac{1}{2\gamma}\boldsymbol{I} + \boldsymbol{Q})\boldsymbol{x}$, where $\boldsymbol{I}$ is an identity matrix. Then, referring to [11], the ℓ_2 regularization term contributes to regularizing the matrix, $\boldsymbol{Q}$. Moreover, this approach is close to the Bayesian shrinkage for stabilizing the matrix, as suggested by [21]. Therefore, this reformulation is theoretically well-motivated.

†Ex. In $N = 5$, $\boldsymbol{x}^{*\top} = \{0.6, 0.3, 0.1, 0, 0\}$, $\|\boldsymbol{x}^*\|_2 \fallingdotseq 0.6782$. The top two stocks make up 90% of the portfolio.

‡Ex. In $N = 5$, $\boldsymbol{x}^{*\top} = \{0.4, 0.3, 0.2, 0.05, 0.05\}$, $\|\boldsymbol{x}^*\|_2 \fallingdotseq 0.5431$. Each stock is distributed with a balance.

Besides, if $\boldsymbol{P} = k\boldsymbol{I}$ for some $k > 0$, the ℓ_2 regularization term in Problem (2.13) reduces to a constant and therefore does not affect the diversification. This corresponds to a highly restrictive case in which all assets are uncorrelated and have identical variances. In general, the covariance matrix $\boldsymbol{P}$ is not proportional to the identity, and the regularization term induces shrinkage of portfolio weights, with its effect shaped by the covariance structure among assets.

Notably, we do not claim that the surrogate Problem (2.12), which regularizes $\boldsymbol{y}$ instead of $\boldsymbol{x}$, preserves the exact interpretation of direct ℓ_2 regularization on $\boldsymbol{x}$. What is lost is the direct regularization of the portfolio weight vector itself. What is preserved is a diversification-inducing effect: the regularization on $\boldsymbol{y}$ discourages highly concentrated allocations while maintaining computational tractability. However, it no longer guarantees an exactly equally weighted portfolio.

Remark 1. *We suppose the following problem for the regularization term case:*

$$\tilde{\boldsymbol{x}} := \arg\min_{\boldsymbol{x} \in \mathcal{X}} \{\, \boldsymbol{x}^\top \boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{X} \}, \tag{2.14}$$

where $\tilde{\boldsymbol{x}}$ is an optimal solution. Problem (2.14) regularizes $\boldsymbol{x}$. Moreover, we assume the following problem:

$$\max_{\boldsymbol{x}} \{\, \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{X} \}. \tag{2.15}$$

Problem (2.15) focuses on a few elements of $\boldsymbol{x}$ because $\boldsymbol{P}$ is assumed to be a positive definite matrix[§]*. Thus, Problems (2.14) and (2.15) have opposite effects on $\boldsymbol{x}$. Then, we construct the following problem using Problems (2.14) and (2.15):*

$$\hat{\boldsymbol{x}} := \arg\min_{\boldsymbol{x} \in \mathcal{X}} \left\{ \frac{\boldsymbol{x}^\top \boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x}} \mid \boldsymbol{x} \in \mathcal{X} \right\}, \tag{2.16}$$

where $\hat{\boldsymbol{x}}$ is the optimal solution. The objective function of Problem (2.16) is the first term of the objective function in Problem (2.13). Here, let us define $C_2 := \hat{\boldsymbol{x}}^\top \boldsymbol{P} \hat{\boldsymbol{x}}$, and then Problem (2.16) is equivalent to the following problem:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x} \in \mathcal{X}} \{\, \boldsymbol{x}^\top \boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{X}, \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x} = C_2 \}. \tag{2.17}$$

$\tilde{\boldsymbol{x}}^\top \tilde{\boldsymbol{x}} \leq \hat{\boldsymbol{x}}^\top \hat{\boldsymbol{x}}$ holds because of the interference of $\boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x} = C_2$. However, C_2 obtained from Problem (2.16) does not exceed the value obtained from Problem (2.15). This is because, in Problem (2.16), the regularization of Problem (2.14) is likely to neutralize the concentration to a specified element of $\boldsymbol{x}$ in Problem (2.15).

Accordingly, the reformulation in Problem (2.16) provides a weaker regularization effect on $\boldsymbol{x}$ compared with Problem (2.14). Furthermore, Problem (2.12) encourages diversification more weakly than Problem (2.11).

Additionally, we can represent Problem (2.12) in the dual formulation.

[§]In an easy constraint case, $\max_{\boldsymbol{x}}\{\, \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x} \mid \sum_{n=1}^{N} x_n = 1, 0 \leq x_n (n \in \mathcal{N})\} = \max\{P_{n,n} (n \in \mathcal{N})\}$, where $P_{n,n}$ is each diagonal element in $\boldsymbol{P}$. Accordingly, regarding $\boldsymbol{x}$, only one element is one and the other elements are zero.

Theorem 1. *By directly applying the dual reformulation for convex optimization problems [14], the dual formulation of Problem (2.12) is as follows:*

$$\left| \begin{array}{ll} \max\limits_{\boldsymbol{w},\boldsymbol{\phi},s,b} \min\limits_{\boldsymbol{u}} & -\dfrac{\gamma}{2}\boldsymbol{w}^\top \boldsymbol{w} - \dfrac{\boldsymbol{\phi}^\top \boldsymbol{\phi}}{4} + s \\ s.t. & \boldsymbol{w} \geq -\boldsymbol{E}^\top \boldsymbol{\phi} + 2s\boldsymbol{R}^\top \boldsymbol{u} + b(\tilde{\boldsymbol{r}} - \rho \mathbf{1}), \\ & \boldsymbol{w} \geq \mathbf{0},\ b \geq 0, \end{array} \right. \tag{2.18}$$

where $\boldsymbol{w} \in \mathbb{R}^N,\ b \in \mathbb{R},\ s \in \mathbb{R},\ \boldsymbol{\phi} \in \mathbb{R}^T$ are variables and $\mathbf{1}$ denotes the all-ones vector of appropriate dimension.

Proof. See Appendix A. □

Observation 1. *Because the constraint $\boldsymbol{u}^\top \boldsymbol{u} = 1$ is nonconvex, strong duality does not hold in Theorem 1. Therefore, different from the cases of convex optimization problems, the dual Problem (2.18) is not equivalent to the primal Problem (2.12).*

2.4. Cardinality constraint

This subsection adds a cardinality constraint to Problem (2.12) to limit the number of invested stocks. Let S be a user-defined parameter indicating an upper bound of the number of invested assets. We impose the following cardinality constraint:

$$\|\boldsymbol{y}\|_0 \leq S, \tag{2.19}$$

where $\|\cdot\|_0$ is an ℓ_0-pseudo-norm (i.e., the number of nonzero elements).

We define the following set, $\mathcal{Z}$, by introducing a vector of binary variables, $\boldsymbol{z} \in \{0,1\}^N$:

$$\mathcal{Z} := \left\{ \boldsymbol{z} \in \{0,1\}^N \mid \sum_{n=1}^{N} z_n \leq S \right\}. \tag{2.20}$$

Moreover, we define the following constraint,

$$0 \leq y_n \leq M z_n \ (n \in \mathcal{N}). \tag{2.21}$$

Equivalently, $0 \leq y_n \leq M z_n$ $(n \in \mathcal{N})$ can be described by the following logical implication:

$$z_n = 0 \Rightarrow y_n = 0 \ (\forall n \in \mathcal{N}). \tag{2.22}$$

We attach (2.20) and (2.21) to Problem (2.12) as follows:

$$\left| \begin{array}{ll} \min\limits_{\boldsymbol{y},\boldsymbol{z},\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}^\top \boldsymbol{y} + \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y},\ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y},\ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\ & \boldsymbol{y} \in \mathcal{Y},\ \boldsymbol{z} \in \mathcal{Z},\ 0 \leq y_n \leq M z_n \ (n \in \mathcal{N}). \end{array} \right. \tag{2.23}$$

Problem (2.23) is a nonconvex MIQCQP problem.

Remark 2. *Consider the equivalent x-formulation in Problem (2.13). As $\boldsymbol{y} = \boldsymbol{x}\eta$ with $\eta > 0$, $\boldsymbol{x}$ and $\boldsymbol{y}$ have the same set of nonzero components (support), and hence the cardinality constraint $\|\boldsymbol{y}\|_0 \leq S$ is equivalently interpreted as $\|\boldsymbol{x}\|_0 \leq S$.*

Let $m := \|\boldsymbol{x}\|_0$, that is, the number of selected assets. Under the long-only and budget constraints in $\mathcal{X}$, namely, $x_n \geq 0$ $(n \in \mathcal{N})$ and $\sum_{n=1}^{N} x_n = 1$, we have

$$1 = \left(\sum_{n \in I(\boldsymbol{x})} x_n \right)^2 \leq m \sum_{n \in I(\boldsymbol{x})} x_n^2 = m\|\boldsymbol{x}\|_2^2, \tag{2.24}$$

by the Cauchy–Schwarz inequality, where $I(\boldsymbol{x}) := \{n \in N \mid x_n > 0\}$. It follows that

$$\|\boldsymbol{x}\|_2^2 \geq \frac{1}{m} \geq \frac{1}{S}. \tag{2.25}$$

Moreover, the equality $\|\boldsymbol{x}\|_2^2 = 1/m$ holds if and only if the nonzero weights are equally allocated. This lower bound explicitly quantifies the extent to which the cardinality constraint limits diversification. Therefore, when the number of selected assets is fixed at m, the ℓ_2 regularization term is minimized by the equally weighted allocation over those m selected assets.

This inequality provides a direct characterization of the extreme-value behavior of the ℓ_2 regularization term under the cardinality constraint. This observation clarifies that the cardinality constraint and the ℓ_2 regularization term play fundamentally different roles. The cardinality constraint controls the number of selected assets, whereas the ℓ_2 regularization term controls the dispersion of weights within the selected asset set. In particular, the cardinality constraint allows highly uneven allocations within the selected set. By contrast, the ℓ_2 regularization term promotes a more even distribution of weights within the selected asset set. Thus, the two components are complementary rather than redundant, as they act on different structural aspects of the portfolio: sparsity versus weight dispersion.

For a fixed γ, S limits the degree of diversification induced by the ℓ_2 regularization term, because $\|\boldsymbol{x}\|_2^2$ cannot be smaller than $1/S$. For example, when $S = 1$, we necessarily have $\|\boldsymbol{x}\|_2^2 = 1$, and no diversification is possible regardless of the value of γ. As S increases, the lower bound $1/S$ decreases, and the ℓ_2 regularization term becomes more effective in equalizing the selected weights.

For a fixed S, γ directly controls the strength of the trade-off between concentration and equalization. As $\gamma \to \infty$, the contribution of the ℓ_2 regularization term vanishes, and the model reduces to the cardinality-constrained MPP. In contrast, as γ decreases, the ℓ_2 regularization term increasingly penalizes concentrated allocations through $\|\boldsymbol{x}\|_2^2$, while the cardinality constraint continues to limit the attainable level of diversification.

Therefore, the resulting portfolio is determined by the interaction between the combinatorial sparsity imposed by S and the continuous regularization effect controlled by γ.

2.5. Bilevel optimization reformulation

We formulate Problem (2.23) as a bilevel optimization problem following [14, 22, 23] in order to apply CPA.

2.5.1. Lower-level problem formulation of the nonconvex problem

We obtain the continuous lower-level problem by fixing the binary vector $\boldsymbol{z}$. Although this lower-level problem remains nonconvex, it can be embedded into the CPA framework by constructing an equivalent convex problem based on its global optimal solution.

First, we formulate the upper-level problem involving only binary variables:

$$\min_{\boldsymbol{z}} \; f(\boldsymbol{z}) \quad \text{s.t.} \;\; \boldsymbol{z} \in \mathcal{Z}. \tag{2.26}$$

Next, we formulate a lower-level problem. For the preparation, we define a subvector and the corresponding submatrices as follows:

$$\begin{aligned} \boldsymbol{y}_z &= (y_n \mid z_n = 1) \in \mathbb{R}^{|\mathcal{N}(z)|}, \\ \tilde{\boldsymbol{r}}_z &= (\tilde{r}_n \mid z_n = 1) \in \mathbb{R}^{|\mathcal{N}(z)|}, \\ \boldsymbol{E}_z &= (\boldsymbol{E}_{(:,n)} \mid z_n = 1) \in \mathbb{R}^{T\times|\mathcal{N}(z)|}, \\ \boldsymbol{R}_z &= (\boldsymbol{R}_{(:,n)} \mid z_n = 1) \in \mathbb{R}^{T\times|\mathcal{N}(z)|}, \end{aligned} \tag{2.27}$$

where $\mathcal{N}(z) := \{n \in \mathcal{N} \mid z_n = 1\}$. In addition, we define a subset as follows:

$$\mathcal{Y}_z := \left\{ \boldsymbol{y}_z \mid \sum_{n\in\mathcal{N}(z)} \tilde{r}_n y_n \ge \rho \sum_{n\in\mathcal{N}(z)} y_n, \; 0 \le y_n \le M \; (n \in \mathcal{N}(z)) \right\}. \tag{2.28}$$

Then, we can reformulate the following lower-level problem by eliminating $z \in \mathcal{Z}$ in Problem (2.23) and replacing (2.27) and (2.28).

$$\begin{aligned} f(\boldsymbol{z}) = \min_{\boldsymbol{y}_z,\boldsymbol{u},\boldsymbol{v}} \quad & \frac{1}{2\gamma}\boldsymbol{y}_z^\top \boldsymbol{y}_z + \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} \quad & \boldsymbol{v} = \boldsymbol{E}_z\boldsymbol{y}_z, \; \boldsymbol{u} = \boldsymbol{R}_z\boldsymbol{y}_z, \; \boldsymbol{u}^\top\boldsymbol{u} = 1, \\ & \boldsymbol{y}_z \in \mathcal{Y}_z. \end{aligned} \tag{2.29}$$

Problem (2.29) is a nonconvex QCQP.

Lemma 1. *For any fixed $z \in Z$ such that Problem (2.29) is feasible, Problem (2.29) admits a global optimal solution.*

Proof. For any fixed $\boldsymbol{z}$ such that Problem (2.29) is feasible, the set $\mathcal{Y}_z$ is compact. Indeed, it is closed because it is defined by linear inequalities, and bounded due to the constraints $0 \le y_n \le M$ for each $n \in \mathcal{N}(z)$. In addition, for each $\boldsymbol{y}_z \in \mathcal{Y}_z$, the vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ are uniquely determined once $\boldsymbol{y}_z$ is fixed through the linear constraints $\boldsymbol{u} = \boldsymbol{R}_z\boldsymbol{y}_z$ and $\boldsymbol{v} = \boldsymbol{E}_z\boldsymbol{y}_z$. Hence, the feasible set of Problem (2.29) is the subset of $\mathcal{Y}_z$ satisfying $\boldsymbol{y}_z^\top \boldsymbol{R}_z^\top \boldsymbol{R}_z \boldsymbol{y}_z = 1$, which is a closed condition since it is a continuous quadratic function. Therefore, the feasible set of Problem (2.29) is compact. As the objective function is continuous, Problem (2.29) attains a global optimal solution whenever it is feasible. □

As shown in Observation 1, we cannot obtain a dual problem equivalent to Problem (2.29). However, after we calculate a global optimal solution $\boldsymbol{u}^*(z)$ by solving Problem (2.29) using a commercial solver such as Gurobi[¶], we can formulate the following convex problem,

$$\begin{aligned} f(z) = \min_{\boldsymbol{y}_z,\boldsymbol{u},\boldsymbol{v}} \quad & \frac{1}{2\gamma}\boldsymbol{y}_z^\top \boldsymbol{y}_z + \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} \quad & \boldsymbol{v} = \boldsymbol{E}_z \boldsymbol{y}_z,\ \boldsymbol{u} = \boldsymbol{R}_z \boldsymbol{y}_z,\ \boldsymbol{u}^\top(\boldsymbol{u}^*(z)) = 1, \\ & \boldsymbol{y}_z \in \mathcal{Y}_z. \end{aligned} \tag{2.30}$$

For a fixed z, we first solve the nonconvex Problem (2.29) to obtain a global optimal solution by using a commercial solver. Let $(\boldsymbol{y}_z^*, \boldsymbol{u}^*(z), \boldsymbol{v}^*)$ be a global optimal solution of Problem (2.29). Based on this solution, we construct Problem (2.30) by replacing the nonconvex constraint $\boldsymbol{u}^\top\boldsymbol{u} = 1$ in Problem (2.29) with the linear constraint $\boldsymbol{u}^\top(\boldsymbol{u}^*(z)) = 1$.

In practice, a global solution of Problem (2.29) can be obtained using a commercial solver that supports nonconvex QCQPs, such as Gurobi. In the following, we assume the use of such a global solver.

Lemma 2. *Fix a binary vector $z \in \{0,1\}^N$. Let $(\boldsymbol{y}_z^*, \boldsymbol{u}^*(z), \boldsymbol{v}^*)$ be a global optimal solution of Problem (2.29). Then Problem (2.29) and Problem (2.30), constructed with $\boldsymbol{u}^*(z)$, have the same optimal value.*

Proof. The optimal solution $(\boldsymbol{y}_z^*, \boldsymbol{u}^*(z), \boldsymbol{v}^*)$ of Problem (2.29), which satisfies $(\boldsymbol{u}^*(z))^\top(\boldsymbol{u}^*(z)) = 1$, is feasible for Problem (2.30). This implies that the optimal value of Problem (2.30) is less than or equal to that of Problem (2.29).

Conversely, for any feasible solution $(\hat{\boldsymbol{y}}_z, \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}})$ of Problem (2.30), the Cauchy–Schwarz inequality implies $\|\hat{\boldsymbol{u}}\|_2^2 \geq 1$ by $1 = (\boldsymbol{u}^\top\boldsymbol{u}^*(z))^2 \leq \|\boldsymbol{u}\|_2^2\,\|\boldsymbol{u}^*(z)\|_2^2 = \|\boldsymbol{u}\|_2^2$. Therefore, $(\hat{\boldsymbol{y}}_z, \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}})/\sqrt{\hat{\boldsymbol{u}}^\top\hat{\boldsymbol{u}}}$ is feasible for Problem (2.29). This implies that the optimal value of Problem (2.29) is less than or equal to that of Problem (2.30).

Hence, both problems attain the same optimal value. □

2.5.2. Dual formulation in the CPA framework

Having established an equivalent convex formulation for each fixed z (Lemma 2), we proceed to derive the dual formulation required for implementing CPA.

Theorem 2. *Fix a binary vector $z \in \{0,1\}^N$, and let $\boldsymbol{u}^*(z)$ be a global optimal solution of Problem (2.29). Then, Problem (2.29) can be equivalently handled as a convex optimization problem; it attains the same optimal solution as Problem (2.30), and admits a dual formulation that can be used within the CPA framework (2.31).*

$$f(z) = \max_{\boldsymbol{w}_z,\boldsymbol{\phi},s,b} -\frac{\gamma}{2}(\boldsymbol{w}_z^\top\boldsymbol{w}_z) - \frac{\boldsymbol{\phi}^\top\boldsymbol{\phi}}{4} + s \tag{2.31a}$$

$$s.t.\ \ \boldsymbol{w}_z \geq -\boldsymbol{E}_z^\top\boldsymbol{\phi} + 2s\boldsymbol{R}_z^\top(\boldsymbol{u}^*(z)) + b(\tilde{\boldsymbol{r}}_z - \rho\boldsymbol{1}), \tag{2.31b}$$

$$\boldsymbol{w}_z \geq \boldsymbol{0},\ b \geq 0, \tag{2.31c}$$

where $\boldsymbol{w}_z = (w_n \mid z_n = 1) \in \mathbb{R}^{|\mathcal{N}(z)|}$, $b \in \mathbb{R}$, $s \in \mathbb{R}$, $\boldsymbol{\phi} \in \mathbb{R}^T$ are dual variables.

¶ https://www.gurobi.com

Proof. See Appendix B. □

Notably, this result establishes a valid dual formulation for the lower-level problem regardless of whether the original formulation is convex or nonconvex, which is essential for embedding the MPP problem into the existing CPA calculation step.

$(\boldsymbol{\phi}^*, s^*, b^*)$ is an optimal solution of $(\boldsymbol{\phi}, s, b)$ in Problem (2.31). When $z_n = 0$ $(n \in \mathcal{N})$, according to Remark 2 by [15], we estimate $\boldsymbol{w}^*$ satisfying $\boldsymbol{w} \geq \boldsymbol{0}$ as

$$w_n = \max\left[-(\boldsymbol{E}_{(:,n)})^\top \boldsymbol{\phi}^* + 2s^*(\boldsymbol{R}_{(:,n)})^\top \boldsymbol{u}^*(\boldsymbol{z}) + b^*(\tilde{r}_n - \rho),\, 0\right] (n \in \mathcal{N} \text{ with } z_n = 0). \tag{2.32}$$

This dual solution is optimal because of its coefficient at $z_n = 0$ in the objective function (2.31a) and then $-\frac{\gamma}{2}(\boldsymbol{w}_z^\top \boldsymbol{w}_z) = -\frac{\gamma}{2}\sum_{n\in\mathcal{N}} z_n w_n w_n$. Additionally, similar to [14], we can observe the relationship between the primal variable $\boldsymbol{y} \in \mathbb{R}^N$ and the dual variable $\boldsymbol{w} \in \mathbb{R}^N$:

$$\boldsymbol{y} = \gamma \mathrm{Diag}(\boldsymbol{z})\boldsymbol{w}, \tag{2.33}$$

where $\mathrm{Diag}(\boldsymbol{z})$ is the diagonal matrix whose diagonal elements are given by the components of $\boldsymbol{z}$. Thus, we can obtain a subgradient of $f(\boldsymbol{z})$ for $\boldsymbol{z} \in \{0, 1\}^N$ as follows:

$$g(\boldsymbol{z}) := -\frac{\gamma}{2}(\boldsymbol{w}(\boldsymbol{z})^* \circ \boldsymbol{w}(\boldsymbol{z})^*) \in \partial f(\boldsymbol{z}), \tag{2.34}$$

where $\boldsymbol{w}(\boldsymbol{z})$ is given by (2.32) and $\boldsymbol{w}(\boldsymbol{z}) \circ \boldsymbol{w}(\boldsymbol{z})$ denotes the Hadamard product of $\boldsymbol{w}(\boldsymbol{z})$ with itself. Although $\boldsymbol{w}(\boldsymbol{z})$ depends on $\boldsymbol{z}$ through the optimal solution of Problem (2.31), the implicit dependence does not contribute to the subgradient expression of the objective function.

Remark 3. *The subgradient in Equation (2.34) is derived following the same procedure as in the existing CPA for convex portfolio optimization models [14]. In all these settings, the subgradient is evaluated at a fixed $\boldsymbol{z}$ using the optimal solution of the corresponding lower-level problem, and therefore implicitly depends on $\boldsymbol{z}$. This dependence is inherent to the existing CPA and is not specific to the nonconvexity of MPP. The novelty of this study lies in establishing an equivalent dual formulation for the nonconvex lower-level problem (Theorem 2), while the subsequent subgradient expression follows the existing CPA.*

3. CPA

This section explains how the CPA proposed by [14] is tailored to solving the MPP problem. Moreover, we incorporate the NL proposed by [20] for faster computation into CPA.

3.1. Algorithm description

We apply CPA [14] to the nonconvex MPP problem.

Let θ_{LB} be the lower bound of the objective value of Problem (2.26). We define the initial feasible region set, $\mathcal{F}_1$, as follows:

$$\mathcal{F}_1 := \{(\boldsymbol{z}, \theta) \in \mathcal{Z} \times \mathbb{R} \mid \theta \geq \theta_{LB}\}, \tag{3.1}$$

where θ is a variable. Then, we set $t \leftarrow 1$.

At the t-th iteration step ($t \geq 1$), we solve the following surrogate upper-level Problem (2.26):

$$\min_{z,\theta} \ \theta \quad \text{s.t. } (z, \theta) \in \mathcal{F}_t, \tag{3.2}$$

where $\mathcal{F}_t$ is a feasible region at the t-th iteration step such that $\mathcal{F}_t \subseteq \mathcal{F}_1$. Equation (3.1) guarantees that the objective value of Problem (3.2) is bounded from below. Thus, Problem (3.2) has an optimal solution (z_t, θ_t), unless $\mathcal{F}_t = \emptyset$.

Next, unlike previous studies on convex programming [14–17], we solve the lower-level primal Problem (2.29) with $z = z_t$ and obtain a global optimal solution $(\boldsymbol{y}_z^*, \boldsymbol{u}^*(z), \boldsymbol{v}^*)$. Then, we solve the dual formulation Problem (2.31) with $z = z_t$ and $\boldsymbol{u}^*(z)$. Thereafter, we estimate the objective value, $f(z_t)$, and the subgradient, $g(z_t)$.

If $f(z_t) \leq \theta_t + \varepsilon$ with a small $\varepsilon(\geq 0)$, we can obtain z_t^* as the ε-optimal solution, and the algorithm terminates. Otherwise, we add the following constraint with $z = z_t$:

$$\mathcal{F}_{t+1} \leftarrow \mathcal{F}_t \cap \{(z, \theta) \in \mathcal{Z} \times \mathbb{R} \mid \theta \geq f(z_t) + g(z_t)^\top (z - z_t)\}. \tag{3.3}$$

This update cuts off the solution (z_t, θ_t) at each iteration step because of $\theta_t < f(z_t)$. We set $t \leftarrow t + 1$ and solve the upper-level Problem (3.2). We iterate this step until we obtain the ε-optimal solution (z^*).

The procedure of our algorithm is described in Algorithm 1, where LB_t and UB_t denote the current lower bound and the incumbent upper bound, respectively.

Algorithm 1 CPA for Problem (2.26)

Require: Tolerance $\varepsilon > 0$
Ensure: ε-optimal solution z^*

```
t ← 1, UB_0 ← ∞
Define feasible region F_1 as in Equation (3.1)
while true do
    Solve Problem (3.2) to obtain (z_t, θ_t)
    LB_t ← θ_t
    Solve Problem (2.29) with z = z_t to obtain u*(z)
    Solve Problem (2.31) with z = z_t and u*(z) to evaluate f(z_t) and w*
    if f(z_t) ≤ UB_{t−1} then
        UB_t ← f(z_t)
        z* ← z_t
    else
        UB_t ← UB_{t−1}
    end if
    if UB_t − LB_t ≤ ε then
        return z*
    end if
    Compute g(z_t) and update feasible region F_{t+1} as in Equation (3.3)
    t ← t + 1
end while
```

Theorem 4 by [15] and Theorem 3 by [16] guarantee that their CPA produces an ε-optimal solution and converges in a finite number of iterations for convex optimization problems.

Theorem 3. *Algorithm 1 terminates after finitely many iterations and returns an ε-optimal solution of Problem (2.26).*

Proof. By Lemma 2 and Theorem 2, for any fixed $z \in \mathcal{Z}$, Algorithm 1 evaluates the exact value $f(z)$ and a valid subgradient $g(z) \in \partial f(z)$. Since the feasible set $\mathcal{Z}$ is finite, and each iteration adds a valid cut based on $g(z)$, Algorithm 1 terminates after finitely many iterations and yields an ε-optimal solution. □

Finally, we can obtain MPP ($\boldsymbol{x}^*_{z^*} = \boldsymbol{y}^*_{z^*}/(\boldsymbol{1}^\top \boldsymbol{y}^*_{z^*})$) by solving Problem (2.29) with $z = z^*$.

3.2. NL for fast computation

This subsection describes NL proposed by [20] for fast computation of approximate solutions to Problem (2.29) in Step 3 of Algorithm 1. The revisited Problem (2.29) is the nonconvex QCQP because $\boldsymbol{u}^\top \boldsymbol{u} = 1$ is a nonconvex constraint.

$$
\left|\begin{array}{ll}
\min\limits_{\boldsymbol{y}_z,\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}_z^\top \boldsymbol{y}_z + \boldsymbol{v}^\top \boldsymbol{v} \\
\text{s.t.} & \boldsymbol{v} = \boldsymbol{E}_z \boldsymbol{y}_z,\ \boldsymbol{u} = \boldsymbol{R}_z \boldsymbol{y}_z,\ \boldsymbol{u}^\top \boldsymbol{u} = 1, \\
& \boldsymbol{y}_z \in \mathcal{Y}_z.
\end{array}\right.
$$

Obtaining an optimal solution typically requires substantial computational time.

To overcome this, we replace the nonconvex constraint with a convex constraint $(\boldsymbol{u}^\top)\boldsymbol{u}^{k-1} = 1$ in which $\boldsymbol{u}^{k-1}$ is a constant value with $k = 1$. Then, we reformulate the problem as the following convex program:

$$
\left|\begin{array}{ll}
\min\limits_{\boldsymbol{y}_z,\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}_z^\top \boldsymbol{y}_z + \boldsymbol{v}^\top \boldsymbol{v} \\
\text{s.t.} & \boldsymbol{v} = \boldsymbol{E}_z \boldsymbol{y}_z,\ \boldsymbol{u} = \boldsymbol{R}_z \boldsymbol{y}_z,\ (\boldsymbol{u}^\top)\boldsymbol{u}^{k-1} = 1, \\
& \boldsymbol{y}_z \in \mathcal{Y}_z.
\end{array}\right. \tag{3.4}
$$

We can quickly solve Problem (3.4) as a convex quadratic programming problem using a commercial solver such as Gurobi. However, the obtained solution $(\hat{\boldsymbol{y}}^k_z, \hat{\boldsymbol{u}}^k, \hat{\boldsymbol{v}}^k)$ may not satisfy the nonconvex constraint, as it may result in $(\hat{\boldsymbol{u}}^k)^\top \hat{\boldsymbol{u}}^k > 1$. To avoid this, we scale $(\boldsymbol{y}^k_z, \boldsymbol{u}^k, \boldsymbol{v}^k) \leftarrow \frac{(\hat{\boldsymbol{y}}^k_z, \hat{\boldsymbol{u}}^k, \hat{\boldsymbol{v}}^k)}{\sqrt{(\hat{\boldsymbol{u}}^k)^\top \hat{\boldsymbol{u}}^k}}$ so that $(\boldsymbol{u}^k)^\top \boldsymbol{u}^k = 1$. Iterating the above procedure with $k \leftarrow k + 1$ generates a sequence $\{(\boldsymbol{y}^k_z, \boldsymbol{u}^k, \boldsymbol{v}^k)\}$.

We summarize NL in Algorithm 2.

Algorithm 2 NL for Problem (2.29)

Require: Initial input $(\boldsymbol{y}_z^0, \boldsymbol{u}^0, \boldsymbol{v}^0)$ with $(\boldsymbol{u}^0)^\top \boldsymbol{u}^0 = 1$ and tolerance $\epsilon > 0$
Ensure: Solution $(\boldsymbol{y}_z^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$

$k \leftarrow 1$
while true **do**
  Solve Problem (3.4) with input $\boldsymbol{u}^{k-1}$ to obtain $(\hat{\boldsymbol{y}}_z^k, \hat{\boldsymbol{u}}^k, \hat{\boldsymbol{v}}^k)$
  Set $(\boldsymbol{y}_z^k, \boldsymbol{u}^k, \boldsymbol{v}^k) \leftarrow (\hat{\boldsymbol{y}}_z^k, \hat{\boldsymbol{u}}^k, \hat{\boldsymbol{v}}^k)/\sqrt{(\hat{\boldsymbol{u}}^k)^\top \hat{\boldsymbol{u}}^k}$
  if $\sqrt{(\hat{\boldsymbol{u}}^k)^\top \hat{\boldsymbol{u}}^k} \leq 1 + \epsilon$ **then**
    $(\boldsymbol{y}_z^*, \boldsymbol{u}^*, \boldsymbol{v}^*) \leftarrow (\boldsymbol{y}_z^k, \boldsymbol{u}^k, \boldsymbol{v}^k)$
    return $(\boldsymbol{y}_z^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$
  else
    $k \leftarrow k + 1$
  end if
end while

For the original problem, Proposition 3.2 of [20] established finite termination of the procedure. For the proposed ℓ_2-regularized Problem (2.29), we focus on the boundedness of the generated sequence and the first-order optimality of its accumulation points, which are stated below.

Proposition 1. *Fix a binary vector z, and suppose that Problem (2.29) is feasible and that the set $\mathcal{Y}_z$ is bounded. Let $\{(\boldsymbol{y}_z^k, \boldsymbol{u}^k, \boldsymbol{v}^k)\}$ be the sequence generated by the NL algorithm applied to Problem (2.29) from a feasible initial point. Then the sequence $\{(\boldsymbol{y}_z^k, \boldsymbol{u}^k, \boldsymbol{v}^k)\}$ is bounded. Moreover, when $\epsilon = 0$, every accumulation point of this sequence satisfies the Karush–Kuhn–Tucker (KKT) conditions of Problem (2.29).*

Proof. See Appendix C. □

Notably, NL is a heuristic algorithm, not an algorithm for obtaining an ϵ-optimal solution. However, it was reported that NL could find ϵ-optimal solutions in computational experiments by comparing optimal solutions [20]. Therefore, when we apply NL to CPA, we can expect to rapidly obtain a near-optimal solution to Problem (2.29).

Observation 2. *Incorporating NL into CPA can significantly accelerate the lower-level computation. As the approximate solution $\tilde{u}(\boldsymbol{z})$ obtained by NL may differ from a global optimal solution $u^*(\boldsymbol{z})$ of the lower-level problem, the associated dual solution and subgradient are not guaranteed to be exact for the true value function $f(\boldsymbol{z})$. As a result, the corresponding subgradient cut is no longer ensured to be valid for $f(\boldsymbol{z})$, and the finite termination and ε-optimality guarantees established for the exact CPA framework (Theorem 3) are not generally preserved. Accordingly, CPA with NL should be interpreted as a heuristic acceleration method introduced for computational tractability.*

4. Setup of computational experiments

4.1. Existing algorithm for performance comparison

This subsection describes an existing heuristic algorithm proposed by [6] for performance comparison. The algorithm can find better-quality approximate solutions within shorter computation times, although these solutions are not ε-optimal.

This algorithm involves a two-step approach (TS). The first step is to solve the relaxation problem without a cardinality constraint. The second step is to solve the problem with a cardinality constraint under reduced binary variables based on the previously obtained solution. Both steps are calculated by NL.

First, we derive a solution $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$ by iteratively calculating the following Problem (4.1) via NL.

$$\left|\begin{array}{ll} \min\limits_{\boldsymbol{y},\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}^\top \boldsymbol{y} + \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y},\ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y}, (\boldsymbol{u}^\top)\boldsymbol{u}^{k-1} = 1, \\ & \boldsymbol{y} \in \mathcal{Y}. \end{array}\right. \tag{4.1}$$

Second, we use the obtained solution $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$ to update the input value as $(\boldsymbol{y}^0, \boldsymbol{u}^0, \boldsymbol{v}^0) \leftarrow (\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$. We repeatedly solve the following problem with the reduced binary variables via NL.

$$\left|\begin{array}{ll} \min\limits_{\boldsymbol{y},\boldsymbol{z},\boldsymbol{u},\boldsymbol{v}} & \dfrac{1}{2\gamma}\boldsymbol{y}^\top \boldsymbol{y} + \boldsymbol{v}^\top \boldsymbol{v} \\ \text{s.t.} & \boldsymbol{v} = \boldsymbol{E}\boldsymbol{y},\ \boldsymbol{u} = \boldsymbol{R}\boldsymbol{y},\ (\boldsymbol{u}^\top)\boldsymbol{u}^{k-1} = 1, \\ & \boldsymbol{y} \in \mathcal{Y},\ \boldsymbol{z} \in \mathcal{Z}, \\ & z_n = y_n = 0, \quad (n \in \mathcal{N} \text{ and } y_n^0 = 0). \end{array}\right. \tag{4.2}$$

It has been noted that this algorithm is not guaranteed to achieve an ε-optimal solution; however, the calculation error of the obtained solution is small compared with that of the exact approach [6]. In addition, they reported that the algorithm could terminate the calculation within 3600 s for even a large-scale dataset, such as $N = 1500$, using a commercial solver such as Gurobi.

An advantage of this algorithm is that it can reduce the number of binary variables by more than 90%: only about 40 binary variables remain when $N = 1500$. However, as smaller values of γ strengthen the diversification effect of the ℓ_2 regularization, the computational time increases because the reduction in the number of binary variables weakens.

We summarize TS in Algorithm 3.

Algorithm 3 TS for Problem (2.23)

Require: Problems (2.12), (4.1), (4.2) available; NL procedure for inner solves

Ensure: Solution $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$

1: Solve Problem (2.12) using iterative solution of (4.1) via NL to obtain $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$
2: Set $(\boldsymbol{y}^0, \boldsymbol{u}^0, \boldsymbol{v}^0) \leftarrow (\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$
3: Solve Problem (2.23) with inputs $(\boldsymbol{y}^0, \boldsymbol{u}^0)$ using the iterative solution of (4.2) via NL to obtain final $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$
4: **return** $(\boldsymbol{y}^*, \boldsymbol{u}^*, \boldsymbol{v}^*)$

4.2. Datasets and settings

Table 1 lists the monthly return datasets ($N \in \{25, 200, 1400, 3000\}$) that we could obtain without missing data from July 2012 to June 2023 for each target index or market[‖].

[‖]https://finance.yahoo.co.jp

Table 1. Dataset.

N	Data acquisition target
25	Composition stocks from the TOPIX Core 30 index
200	Composition stocks from the Nikkei 225 index
1400	Companies listed in the Tokyo Prime Market
3000	Companies listed in the Tokyo Prime Market and the New York Stock Exchange

We employed six factors from the studies by [24], [25], and [26]**. We used the following factors in the cases of $K = 1, 3$, and 6 as in Table 2:

Table 2. List of factors.

Factors	$K = 1$	$K = 3$	$K = 6$
MKT: Return of the market index	√	√	√
SMB: Size factor		√	√
HML: Style factor		√	√
RML: Profitability factor			√
CMA: Investment factor			√
WML: Momentum factor			√

We prepared seven instances as $T = 48$, July 2012 through June 2016, July 2013 through June 2017, . . ., and July 2019 through June 2022 in an in-sample (Section 5.1) and then rebalanced the MPP for June 2016, June 2017, . . ., and June 2022 with $N = 200$ and 1400 in the out-of-sample period (Section 5.2). The timeline is shown in Figure 1.

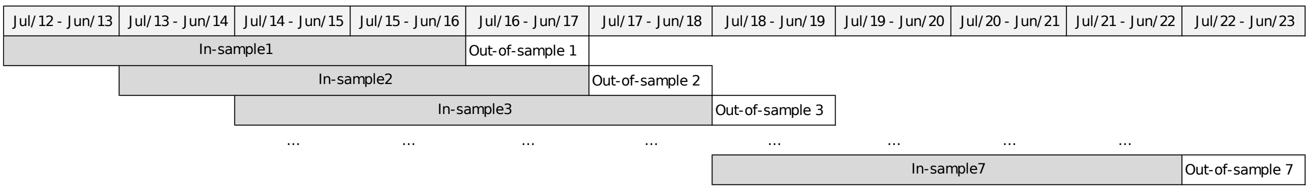


Figure 1. In-sample and out-of-sample periods.

Then, ρ was estimated as the average of the historical TOPIX return with $T = 48$ for June 2016, June 2017,. . ., June 2022.

All the computations were performed with Gurobi 10.0.1 using a personal computer (Core i9-9980, 32.0 GB, and 2.40 GHz) under the setting of $\varepsilon = 1.0\mathrm{e} - 5$ and $\epsilon = 1.0\mathrm{e} - 6$. In addition, we used the initial value, $y_n^0 \leftarrow 1/\sqrt{|\mathcal{N}(z)|}$ $(n \in \mathcal{N}(z))$, in Step 1 of Algorithm 2, as in the studies by [6, 20]. The maximum computational time was set to 7200 s for the experiments in Sections 5.1.1 and 5.1.2, and to 3600 s for those in Sections 5.1.3 and 5.1.4.

For the computational implementation, we used the logical implication formulation in (2.22) instead of the big-M formulation in (2.21).

**https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

Notably, each data instance satisfies $T = 48 < N \in \{200, 1400, 3000\}$ and $\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} = \boldsymbol{x}^\top \boldsymbol{R}^\top \boldsymbol{R}\boldsymbol{x}$ is not guaranteed to be strictly positive. However, $T = 48 > |\mathcal{N}(z)| \in \{5, 10, \cdots, 30\}$ holds due to the cardinality constraint: thus, all obtained $\boldsymbol{x}_z^\top \boldsymbol{R}_z^\top \boldsymbol{R}_z \boldsymbol{x}_z$ were positive in the computational results.

4.3. Methods for comparison

We compare the calculation results of the following four calculation methods.

CPA	:	Solves Problem (2.23) using Algorithm 1 and Step 3 using the `NonConvex` function in Gurobi.
CPA+	:	Solves Problem (2.23) using Algorithm 1, with the lower-level problem solved by Algorithm 2.
TS	:	Solves Problem (2.23) using Algorithm 3.
NC	:	Solves Problem (2.23) using the `NonConvex` function in Gurobi.

4.4. Evaluation metrics

The following column labels are used in Section 5.1.

μObj	:	Average values of the objective values (Obj) of seven instances for each calculation method in Problem (2.23).
μGap	:	Average values of the relative errors of the objective values of CPA+ compared with that of TS, CPA, or NC are as follows: $\dfrac{(\text{Obj of TS, CPA or NC}) - (\text{Obj of CPA+})}{\text{Obj of CPA+}}$.
μTime	:	Average values of the calculation time.
μCut	:	Average values of the cut-off number of CPA and CPA+.
#Opt	:	Number of instances, out of seven, solved within the maximum computational time.
μMIPGap	:	Average value of MIPGap. MIPGap is the difference between the lower and upper bounds of the optimal objective value divided by the upper bound in the instances finished within the maximum computational time.

The following column labels are used in Section 5.2.

$\mu R^2(\boldsymbol{x})$	:	Average values of $R^2(\boldsymbol{x})$ of seven instances in an in-sample.
Return	:	Annual return in the out-of-sample period.
Risk	:	Annualized standard deviation in the out-of-sample period.
SR	:	Return/Risk.
UR	:	Average deviation of the realized return over the expected return in an out-of-sample: $\displaystyle\sum_{t=1}^{84} \frac{\max(\sum_{n=1}^{N} R_{n,t}x_{n,t} - \rho, 0)}{84}$.

where $x_{n,1} = x_{n,2} = \cdots = x_{n,12} \neq x_{n,13} = x_{n,14} = \cdots = x_{n,24} \neq x_{n,25} = \cdots = x_{n,84}$ $(n \in \mathcal{N})$ because we rebalance every 12 months, as explained in Section 4.2.

5. Results

This section presents the comparison results for the four computational methods and investment performance under different values of γ.

5.1. Algorithm performance

This subsection describes the performance of four computational methods.

5.1.1. Aggregation results of four methods

This subsubsection provides an aggregated comparison of four methods: CPA+, TS, CPA, and NC, using a small-scale instance with $(N, K, S) = (25, 3, 3)$.

Table 3. Solution quality and computational efficiency on $(N, K, S) = (25, 3, 3)$.

	CPA+	TS	CPA	NC	CPA+	TS	CPA	NC
γ	μObj	μObj (μGap(%))		μObj (μGap(bp)) [††]	μTime(s) (#Opt)			
$\frac{0.01}{\sqrt{N}}$	235.894	271.798 (15.32)	235.891 (0.00)	235.891 (-0.12)	0.1(7)	1(7)	109.4(7)	2444.7(7)
$\frac{0.1}{\sqrt{N}}$	24.443	28.044 (14.83)	24.443 (0.00)	24.443 (-0.14)	0.1(7)	0.7(7)	79.5(7)	1916.7(7)
$\frac{1}{\sqrt{N}}$	3.294	3.826 (16.19)	3.294 (0.00)	3.294 (-0.24)	0.1(7)	0.7(7)	218.2(7)	1039.1(7)
$\frac{10}{\sqrt{N}}$	1.173	1.245 (6.19)	1.189 (1.44)	1.173 (-0.20)	8.5(7)	1(7)	4720.8(4)	2522.7(7)
$\frac{100}{\sqrt{N}}$	0.903	0.914 (1.23)	0.915 (1.39)	0.903 (-0.22)	800.2(7)	0.3(7)	7200.0(0)	5646.1(7)

As shown in Table 3, CPA+ attains objective values comparable to those obtained by the exact methods CPA and NC, while consistently outperforming the existing heuristic method TS. Moreover, CPA+ achieves these high-quality solutions within much shorter computation times than the exact methods.

In contrast, CPA cannot complete within the time limit, whereas NC terminates, albeit with a longer computation time. These results suggest that their practical applicability is limited even for small problem sizes, and that CPA+ offers the best trade-off between solution quality and computational efficiency among the methods considered.

In addition, Appendix D reports the optimal objective values for all seven instances for each γ obtained from NC.

5.1.2. The limits of exact algorithms

This subsubsection examines whether the exact methods CPA and NC remain computationally viable as the combinatorial complexity increases with $(N, K) = (25, 3)$. In addition, "∞" in μMIPGap and "—" in μObj indicate that no instance was solved within the time limit in Table 4.

Table 4 shows that neither CPA nor NC can complete even a single instance within the time limit when the cardinality constraint is moderately increased to $S \in \{5, 10\}$, compared to Table 3. Moreover,

[††] The gap is reported in basis points (bp) rather than percentage points when the magnitude of the difference is sufficiently small.

the reported optimality gaps are large and unstable, indicating that the obtained solutions are far from being certified optimal.

Table 4. Scalability limits of exact methods with $S \in \{5, 10\}$ and $(N, K) = (25, 3)$.

$S = 5$						
Methods	CPA			NC		
γ	μTime(s) (#Opt)	μMIPGap(%)	μObj	μTime(s) (#Opt)	μMIPGap(%)	μObj
$\frac{0.01}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	74.40	234.58
$\frac{0.1}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	76.71	24.76
$\frac{1}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	84.83	3.42
$\frac{10}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	90.69	1.21
$\frac{100}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	92.73	0.95
$S = 10$						
Methods	CPA			NC		
γ	μTime(s) (#Opt)	μMIPGap(%)	μObj	μTime(s) (#Opt)	μMIPGap(%)	μObj
$\frac{0.01}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	78.85	142.41
$\frac{0.1}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	89.98	18.89
$\frac{1}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	92.45	2.95
$\frac{10}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	95.48	1.4
$\frac{100}{\sqrt{N}}$	7200 (0)	∞	—	7200 (0)	94.51	0.98

These results demonstrate that, despite their theoretical guarantees, CPA and NC are not practically applicable beyond tiny problem sizes. Accordingly, the subsequent analysis focuses on CPA+ and TS.

5.1.3. Early identification of high-quality solutions by CPA+

This subsubsection investigates why CPA+ consistently yields better solutions than TS, which remains the only practical baseline remaining after excluding exact methods in Section 5.1.2.

Figure 2 illustrates the progression of the upper and lower bounds (UB_t and LB_t) in one instance with $(N, K, S) = (200, 3, 10)$.

A high-quality solution is identified within a few iterations, as evidenced by the rapid stabilization of UB_t. In contrast, LB_t improves more gradually, implying that additional iterations are required to certify ε-optimality. This behavior explains why CPA+ can deliver high-quality solutions early, even when full convergence is not achieved within the time limit.

To examine whether this early identification of high-quality solutions is robust across different values of γ, Table 5 reports the performance of CPA+ and TS for a wide range of regularization strengths. Additionally, the four statistics indicate the mean, standard deviation (s. d.), maximum (max), and minimum (min) of the objective values.

Across most values of γ, CPA+ achieves better or comparable objective values. Although TS slightly outperforms CPA+ for large γ, the difference in objective values is marginal, indicating that the local solutions found by CPA+ remain highly competitive.

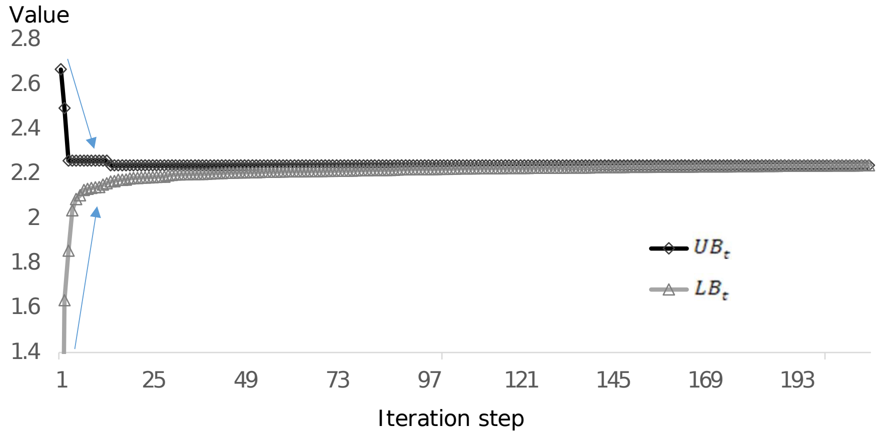


Figure 2. Progression of UB_t and LB_t in CPA+ for one instance with $(N, K, S) = (200, 3, 10)$.

Table 5. CPA+ versus TS across regularization levels γ with $(N, K, S) = (200, 3, 10)$.

γ		CPA+	TS	γ		CPA+	TS
$\frac{0.01}{\sqrt{N}}$	#Opt	7	4	$\frac{10}{\sqrt{N}}$	#Opt	0	6
	μCut	4.4	—		μCut	411.4	—
	μTime(s)	0.1	1552		μTime(s)	3600	520.2
	μMIPGap(%)	0	—		μMIPGap(%)	43.4	—
	mean	158.41	160.52		mean	1.04	1.05
	s. d.	35.62	37.53		s. d.	0.07	0.08
	max	231.31	236.43		max	1.11	1.13
	min	112.14	113.20		min	0.90	0.90
$\frac{0.1}{\sqrt{N}}$	#Opt	7	4	$\frac{100}{\sqrt{N}}$	#Opt	0	7
	μCut	4.4	—		μCut	280.1	—
	μTime(s)	0.2	1553.9		μTime(s)	3600	2.8
	μMIPGap(%)	0	—		μMIPGap(%)	305.5	—
	mean	16.65	16.89		mean	0.74	0.74
	s. d.	3.58	3.78		s. d.	0.04	0.04
	max	24.00	24.52		max	0.79	0.79
	min	12.08	12.08		min	0.68	0.68
$\frac{1}{\sqrt{N}}$	#Opt	7	5				
	μCut	187.9	—				
	μTime(s)	226.2	1046.5				
	μMIPGap(%)	0	—				
	mean	2.48	2.61				
	s. d.	0.36	0.43				
	max	3.23	3.32				
	min	2.07	2.20				

To further examine the solution quality in cases where CPA+ does not terminate, Table 6 reports the objective values obtained by CPA+ and TS for each instance for $\gamma \in \{\sqrt{10/N}, \sqrt{100/N}\}$ with $(N, K, S) = (200, 3, 10)$. Here, **boldface** indicates better objective values.

For $\gamma = \frac{10}{\sqrt{N}}$, CPA+ achieves slightly better objective values than TS in most instances. For $\gamma = \frac{100}{\sqrt{N}}$, TS marginally outperforms CPA+; however, the difference in objective values is negligible. These results indicate that, even when convergence cannot be certified, CPA+ can identify solutions of comparable quality to those obtained by TS. Thus, we conclude that CPA+ can quickly search for better-quality solutions in a few iteration steps; however, it has to expend many iteration steps to reach the convergence condition.

Table 6. Instance-wise comparison of solution quality with $(N, K, S) = (200, 3, 10)$.

			Instance						
γ	Methods		1	2	3	4	5	6	7
$\frac{10}{\sqrt{N}}$	CPA+	Obj	**1.003**	1.019	1.108	**0.896**	**1.063**	**1.098**	**1.068**
		MIPGAP(%)	(37.48)	(54.75)	(34.5)	(32.7)	(23.27)	(67.01)	(53.96)
	TS	Obj	1.011	1.019	1.108	0.897	1.067	1.115	1.134
$\frac{100}{\sqrt{N}}$	CPA+	Obj	0.717	0.737	0.764	0.678	0.742	0.791	0.777
		MIPGAP(%)	(326.09)	(301.04)	(220.44)	(265.71)	(316.91)	(353.55)	(354.51)
	TS	Obj	0.717	0.737	0.764	0.678	**0.74**	**0.79**	**0.776**

5.1.4. Parameter sensitivity analysis

This subsubsection presents the computational results for various sets of parameters (γ, ρ, S, N, K).

We first examine the effect of the expected return constraint. Table 7 shows the results for $(N, K, S) = (200, 3, 10)$ with ρ, $\rho + 0.02$, and $\rho + 0.03$.

As the expected return constraint $(\sum_{n=1}^{N} \tilde{r}_n x_n \geq \rho)$ in (2.5) is tightened, the feasible region becomes narrower. Therefore, provided feasibility is maintained, the optimal objective value of the minimization problem cannot decrease. In Table 7, this change is also reflected in the computational behavior, mainly through the reduction of the MIP gap, suggesting that the algorithm moves more quickly toward termination in some cases, especially for moderate and large values of γ.

Tables 8, 9, and 10 summarize the sensitivity of CPA+ and TS to the cardinality constraint S, the number of candidate assets N, and the number of factors K. Across all settings, CPA+ consistently achieves better or comparable objective values than TS, while preserving the same qualitative performance trends. Although larger values of S, N, and K generally increase computational difficulty, the relative advantage of CPA+ remains unchanged.

Table 7. Results of changing the expected return (ρ) for $(N, K, S) = (200, 3, 10)$ via CPA+.

		γ				
		$\frac{0.01}{\sqrt{N}}$	$\frac{0.1}{\sqrt{N}}$	$\frac{1}{\sqrt{N}}$	$\frac{10}{\sqrt{N}}$	$\frac{100}{\sqrt{N}}$
$\geq \rho$+0	#Opt	7	7	7	0	0
	μCut	4.4	4.4	187.9	411.4	280.1
	μTime(s)	0.1	0.2	226.2	3600	3600
	μMIPGap(%)	0	0	0	43.4	305.5
	μObj	158.41	16.65	2.48	1.05	0.74
$\geq \rho$+0.02	#Opt	7	7	7	0	0
	μCut	4.9	5	122.3	498.4	531.7
	μTime(s)	0.6	0.6	71.3	3600	3600
	μMIPGap(%)	0	0	0	30.5	211.3
	μObj	203.5	21.14	2.88	1.09	0.84
$\geq \rho$+0.03	#Opt	7	7	7	1	0
	μCut	5	5.1	86.6	556.4	697.4
	μTime(s)	0.2	0.2	30.3	3086.3	3600
	μMIPGap(%)	0	0	0	21.8	171.5
	μObj	233.14	24.13	3.23	1.13	0.86

Table 8. Comparison results for different cardinality constraints S with $(N, K) = (200, 3)$.

	$S = 5$			$S = 10$			$S = 15$		
	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)
γ	μObj(#Opt)			μObj(#Opt)			μObj(#Opt)		
$\frac{0.01}{\sqrt{N}}$	251.46(7)	261.36(4)	3.49	158.41(7)	160.52(4)	1.15	120.08(7)	120.35(5)	0.24
$\frac{0.1}{\sqrt{N}}$	25.99(7)	27.00(5)	3.45	16.65(7)	16.89(4)	1.31	12.83(7)	12.92(5)	0.7
$\frac{1}{\sqrt{N}}$	3.40(7)	3.76(6)	9.79	2.48(7)	2.61(5)	4.96	2.12(0)	2.18(5)	2.54
$\frac{10}{\sqrt{N}}$	1.11(1)	1.16(6)	4.07	1.04(0)	1.05(6)	1.27	1.00(0)	1.00(5)	0.24
$\frac{100}{\sqrt{N}}$	0.78(0)	0.78(7)	–0.44	0.74(0)	0.74(7)	–0.06	0.74(0)	0.74(7)	–0.01

	$S = 20$			$S = 25$			$S = 30$		
	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)
γ	μObj(#Opt)			μObj(#Opt)			μObj(#Opt)		
$\frac{0.01}{\sqrt{N}}$	98.33(7)	98.68(5)	0.36	84.37(7)	84.51(5)	0.17	74.63(7)	74.67(5)	0.04
$\frac{0.1}{\sqrt{N}}$	10.66(7)	10.73(5)	0.68	9.28(7)	9.31(5)	0.39	8.31(7)	8.33(5)	0.22
$\frac{1}{\sqrt{N}}$	1.93(0)	1.94(5)	0.98	1.79(0)	1.80(5)	0.29	1.70(0)	1.70(5)	0.03
$\frac{10}{\sqrt{N}}$	0.94(0)	0.97(5)	4.93	0.96(0)	0.96(5)	–0.21	0.95(0)	0.95(6)	0.02
$\frac{100}{\sqrt{N}}$	0.74(0)	0.74(7)	0	0.74(0)	0.74(7)	0	0.74(0)	0.74(7)	0

Table 9. Comparison results for different candidate asset numbers N with $(K, S) = (3, 10)$.

	$N = 200$			$N = 1400$			$N = 3000$		
	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)
γ	μObj(#Opt)			μObj(#Opt)			μObj(#Opt)		
$\frac{0.01}{\sqrt{N}}$	158.41(7)	160.52(4)	1.15	221.92(7)	277.18(0)	23.92	153.13(7)	285.63(0)	79.26
$\frac{0.1}{\sqrt{N}}$	16.65(7)	16.89(4)	1.31	23.07(7)	31.93(1)	48.7	15.70(7)	29.53(0)	79.33
$\frac{1}{\sqrt{N}}$	2.48(7)	2.61(5)	4.96	3.45(7)	4.31(2)	24.94	2.73(5)	4.68(0)	71.71
$\frac{10}{\sqrt{N}}$	1.04(0)	1.05(6)	1.27	1.06(2)	1.22(1)	14.46	1.12(2)	1.39(0)	24.37
$\frac{100}{\sqrt{N}}$	0.74(0)	0.74(7)	–0.06	0.62(0)	0.58(6)	–6.87	0.64(0)	0.62(2)	–2.49

Table 10. Comparison results for different numbers of factors K with $(N, S) = (200, 10)$.

	$K = 1$			$K = 3$			$K = 6$		
	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)	CPA+	TS	μGap (%)
γ	μObj(#Opt)			μObj(#Opt)			μObj(#Opt)		
$\frac{0.01}{\sqrt{N}}$	146.99(7)	146.99(7)	0	158.41(7)	160.52(4)	1.15	158.82(7)	158.87(6)	0.03
$\frac{0.1}{\sqrt{N}}$	15.59(7)	15.59(7)	0	16.65(7)	16.89(4)	1.31	16.77(7)	16.99(5)	1.42
$\frac{1}{\sqrt{N}}$	2.44(5)	2.55(6)	4.65	2.48(7)	2.61(5)	4.96	2.44(7)	2.50(5)	3.06
$\frac{10}{\sqrt{N}}$	1.13(0)	1.12(7)	–0.23	1.04(0)	1.05(6)	1.27	0.98(0)	0.95(7)	–3.04
$\frac{100}{\sqrt{N}}$	0.89(0)	0.89(7)	–0.07	0.74(0)	0.74(7)	–0.06	0.67(0)	0.67(7)	–0.29

Overall, these results confirm that the performance characteristics of CPA+ observed in the previous subsections are robust to variations in key model parameters.

5.2. *Out-of-sample investment performance*

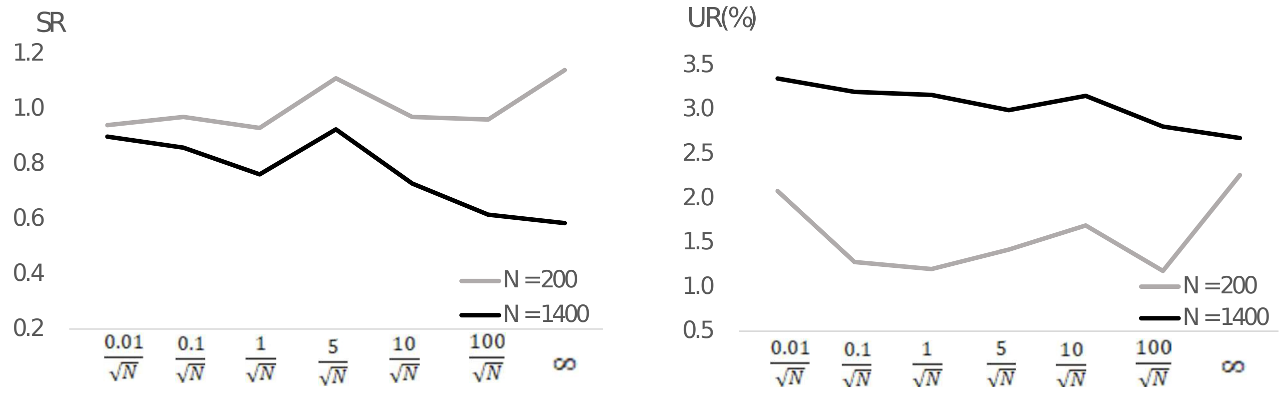
This subsection investigates the effect of the ℓ_2 regularization parameter γ on the out-of-sample performance of MPP. The regularization parameter plays a critical role in balancing the mitigation of overfitting with preservation of predictive structure.

Table 11 summarizes in-sample predictability $\mu R^2(\boldsymbol{x})$ for different values of γ, K, and N. As all γ, K, and N increase, $\mu R^2(\boldsymbol{x})$ increases, suggesting potential overfitting.

Table 11. In-sample predictability ($\mu R^2(\boldsymbol{x})$) across regularization levels and model sizes (%).

K		1			3			6		
N	$S \backslash \gamma$	$\frac{0.01}{\sqrt{N}}$	$\frac{10}{\sqrt{N}}$	∞	$\frac{0.01}{\sqrt{N}}$	$\frac{10}{\sqrt{N}}$	∞	$\frac{0.01}{\sqrt{N}}$	$\frac{10}{\sqrt{N}}$	∞
200	10	1.2	8.4	18.0	5.7	19.1	33.8	10.1	27.2	42.2
	20	1.3	9.1	18.1	5.7	21.0	33.8	9.8	27.7	42.2
	30	1.3	8.9	18.1	5.7	19.5	33.8	9.7	27.2	42.2
1400	10	1.3	13.3	48.6	2.8	25.6	60.5	9.5	46.1	71.1
	20	1.2	18.1	53.5	2.4	34.5	65.1	8.7	50.2	73.7
	30	1.2	31.5	53.9	2.8	40.4	65.5	8.2	51.6	73.8

Figure 3 illustrates the effect of the regularization parameter γ on out-of-sample performance under different numbers of candidate assets $N = \{200, 1400\}$ and $S = 30$. In Figure 3(a), which evaluates return stability via SR, the unregularized case performs best for $N = 200$, where the coefficient of determination is low (Table 11). In contrast, for $N = 1400$, regularization is essential to improve SR, reflecting the stronger presence of overfitting. Figure 3(b) reports UR, which measures return predictability. For $N = 200$, UR exhibits no clear monotonic pattern with respect to γ, whereas for $N = 1400$, UR increases monotonically as regularization strengthens. This indicates that strong regularization gradually suppresses predictive performance when predictability is initially high (Table 11).



(a) SR **(b)** UR

Figure 3. Effect of the regularization parameter γ on out-of-sample performance for $N = \{200, 1400\}$ and $S = 30$.

Overall, these results suggest that regularization becomes increasingly important as the coefficient of determination rises with larger N. In such cases, incorporating ℓ_2 regularization enhances both return stability and investment performance, making the regularized MPP particularly effective in large-scale investment candidate assets.

Table 12 provides detailed numerical results across different values of γ.

The qualitative patterns observed in Figure 3 remain consistent across S. On the other hand, risk likely increases as γ decreases. Thus, moderate regularization improves SR without substantially degrading UR, whereas excessively large γ leads to marginal gains in stability at the cost of predictability. In addition, Appendix E shows that for small candidate sets, γ provides little additional benefit.

The impact of the cardinality constraint S is comparatively modest. While variations in S affect performance levels, they do not alter the fundamental trade-off induced by γ. Accordingly, the dominant factor governing out-of-sample behavior is the strength of regularization rather than the sparsity level.

Table 12. Investment performance with $(N, K) = (1400, 6)$.

S	Metric\ γ	$\frac{0.01}{\sqrt{N}}$	$\frac{0.1}{\sqrt{N}}$	$\frac{1}{\sqrt{N}}$	$\frac{3}{\sqrt{N}}$	$\frac{5}{\sqrt{N}}$	$\frac{9}{\sqrt{N}}$	$\frac{10}{\sqrt{N}}$	$\frac{100}{\sqrt{N}}$	∞
10	$\mu R^2(\boldsymbol{x})(\%)$	9.5	11.5	18.2	22	28.4	34.9	46.1	64.8	71.1
	Return(%)	26.6	18.7	22	15.1	16	21.7	16.2	10.3	10.2
	Risk(%)	28	27.7	27.2	26	27.2	26.1	26.6	20.7	19.1
	SR	0.95	0.67	0.81	0.58	0.59	0.83	0.61	0.5	0.54
	UR(%)	3.93	3.5	3.81	3.26	3.57	3.6	3.34	2.35	2.12
20	$\mu R^2(\boldsymbol{x})(\%)$	8.7	9.3	16.5	20.7	29.1	34.9	50.2	68.6	73.7
	Return(%)	22.8	17.6	23.3	20.8	20.9	18.2	18.4	10.4	10.5
	Risk(%)	25.4	23.5	25	22.6	24.5	23.9	22.8	19.2	17.6
	SR	0.9	0.75	0.93	0.92	0.85	0.76	0.81	0.54	0.59
	UR(%)	3.46	3.03	3.53	3.17	3.35	3.14	3.06	2.22	2.01
30	$\mu R^2(\boldsymbol{x})(\%)$	8.1	8.5	15.5	21.3	29.6	41	51.6	68.8	73.8
	Return(%)	21.9	20.2	18.3	18.1	20.8	16.7	15.5	11.2	10.3
	Risk(%)	24.4	23.5	24	22.7	22.5	21.8	21.3	18.3	17.7
	SR	0.9	0.86	0.76	0.8	0.92	0.77	0.73	0.61	0.58
	UR(%)	3.35	3.2	3.16	2.99	3.15	2.81	2.68	2.1	2.01

Figure 4 shows the nonzero portfolio weights ($\{x_n \,|x_n > 0 \ (n \in \mathcal{N})\}$) as a box plot with varying γ in one instance for $(S, N, K) = (10, 1400, 6)$. The MPP approximates an equally weighted portfolio[‡‡] as γ decreases, as described in Remark 2. Therefore, although our reformulation regarding the ℓ_2 regularization term in Problem (2.12) is non-general, the proposed MPP can bring about the diversification of investment stocks, in particular from $\gamma = \infty$ to $\gamma = \frac{10}{\sqrt{N}}$.

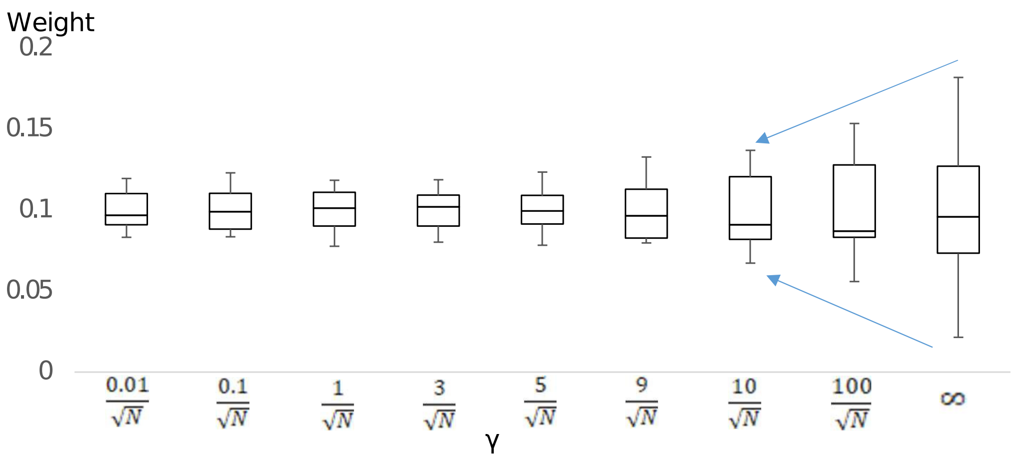


Figure 4. Effect of ℓ_2 regularization on portfolio weight structure with $(S, N, K) = (10, 1400, 6)$.

‡‡The equally weighted value in the case of $S = 10$ is 0.1.

6. Conclusions

This study investigates a cardinality-constrained MPP formulation with an ℓ_2 regularization term and shows that it can be handled within the CPA framework through an appropriate bilevel reformulation. Although the resulting problem is nonconvex, we demonstrate that by exploiting a globally optimal solution of the primal lower-level problem, this formulation admits an equivalent convex formulation that preserves the finite termination and ε-optimality guarantees of CPA. Moreover, we derive a subgradient tailored to MPP, which enables the construction of valid cutting planes despite the nonconvexity of the original formulation.

Numerical experiments indicate that the proposed approach, when combined with NL, computes high-quality solutions within short computation times for large-scale instances where exact solution methods become computationally infeasible, and achieves consistently strong performance relative to existing heuristics. As CPA combined with NL is used as a heuristic acceleration method rather than an exact algorithm, we do not claim convergence-rate results or explicit approximation error bounds for this variant. Establishing such guarantees remains an important topic for future research.

Finally, our results demonstrate that moderate ℓ_2 regularization improves out-of-sample predictability, whereas excessive regularization weakens the predictive structure of MPP. The joint use of ℓ_2 regularization and cardinality constraints, therefore, provides a practical balance between predictability, stability, and practical feasibility within the MPP framework.

A future direction of this study is the adaptive selection of hyperparameters (S, γ) in the cardinality constraint and the ℓ_2 regularization term. While fixed values were used in out-of-sample analysis, dynamically updating these parameters at each rebalancing may further improve investment performance. Existing approaches include search algorithms tailored to specific models [13] and grid search methods, which are feasible even for problems with binary variables [27]. Investigating their applicability within the proposed framework remains an important topic for future research.

Another important extension is to examine the role of ℓ_2 regularization from a multi-objective perspective within the MPP framework, enabling portfolio selection from a Pareto front and reducing dependence on specific hyperparameter choices.

Author contributions

Katsuhiro Tanaka: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Software, Validation, Visualization, Writing – original draft, Writing – review and editing. Rei Yamamoto: Formal analysis, Funding acquisition, Methodology, Resources, Supervision, Validation, Writing – review and editing.

Use of Generative-AI tools declaration

During the preparation of this manuscript, the authors used ChatGPT for language polishing and expression improvement.

Acknowledgments

This work was supported by a Grant-in-Aid for Scientific Research (C) 22K04604.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. A. Lo, K. MacKinlay, Maximizing predictability in the stock and bond markets, *Macroecon. Dyn.*, **1** (1997), 102–134. https://doi.org/10.1017/S1365100597002046
2. P. G. Coulombe, M. Göebel, Maximally machine-learnable portfolios, SSRN working paper, 2023. http://doi.org/10.2139/ssrn.4428178
3. R. Yamamoto, D. Ishii, H. Konno, A maximal predictability portfolio model: Algorithm and performance evaluation, *Int. J. Theor. Appl. Finance*, **10** (2007), 1095–1109. https://doi.org/10.1142/S0219024907004561
4. R. D. Harris, J. Shen, F. Yilmaz, Maximally predictable currency portfolios, *J. Int. Money Finance*, **128** (2022), 102702. https://doi.org/10.1016/j.jimonfin.2022.102702
5. B. Q. Ta, V. T. Huynh, K. Q. H. Nguyen, P. N. Nguyen, B. H. Ho, Maximal predictability portfolio optimization model and applications to vietnam stock market, in *Credible Asset Allocation, Optimal Transport Methods, and Related Topics*, vol. 429 of Studies in Systems, Decision and Control. Springer International Publishing, 2022, 559–578. https://doi.org/10.1007/978-3-030-97273-8_37
6. K. Tanaka, R. Yamamoto, An efficient approach for maximal predictability portfolio with cardinality constraints (in Japanese), *Trans. Oper. Res. Soc. Jpn.*, **66** (2023), 1–22. https://doi.org/10.15807/torsj.66.1
7. H. Konno, Y. Morita, R. Yamamoto, A maximal predictability portfolio using absolute deviation reformulation, *Comput. Manag. Sci.*, **7** (2010), 47–60. https://doi.org/10.1007/s10287-008-0075-2
8. H. Konno, Y. Takaya, R. Yamamoto, A maximal predictability portfolio using dynamic factor selection strategy, *Int. J. Theor. Appl. Finance*, **13** (2010), 355–366. https://doi.org/10.1142/S0219024910005802
9. A. E. Hoerl, R. W. Kennard, Ridge regression: Biased estimation for nonorthogonal problems, *Technometrics*, **12** (1970), 55–67. https://doi.org/10.2307/1267351
10. J. Gotoh, A. Takeda, On the role of norm constraints in portfolio selection, *Comput. Manag. Sci.*, **8** (2011), 323–353. https://doi.org/10.1007/s10287-011-0130-2
11. B. Bruder, N. Gaussel, J. Richard, T. Roncalli, Regularization of portfolio allocation, SSRN working paper, 2013. http://doi.org/10.2139/ssrn.2767358
12. V. DeMiguel, L. Garlappi, F. Nogales, R. Uppal, A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms, *Manag. Sci.*, **55** (2009), 798–812. https://doi.org/10.1287/mnsc.1080.0986

13. A. Takeda, M. Niranjan, J. Gotoh, Y. Kawahara, Simultaneous pursuit of out-of-sample performance and sparsity in index tracking portfolios, *Comput. Manag. Sci.*, **10** (2013), 21–49. https://doi.org/10.1007/s10287-012-0158-y
14. D. Bertsimas, R. Cory-Wright, A scalable algorithm for sparse portfolio selection, *INFORMS J. Comput.*, **34** (2022), 1489–1511. https://doi.org/10.1287/ijoc.2021.1127
15. K. Kobayashi, Y. Takano, K. Nakata, Bilevel cutting-plane algorithm for cardinality-constrained mean-CVaR portfolio optimization, *J. Global Optim.*, **81** (2021), 498–528. https://doi.org/10.1007/s10898-021-01048-5
16. K. Kobayashi, Y. Takano, K. Nakata, Cardinality-constrained distributionally robust portfolio optimization, *Eur. J. Oper. Res.*, **309** (2023), 1173–1182. https://doi.org/10.1016/j.ejor.2023.01.037
17. S. Wang, L. Pang, S. Wang, H. Zhang, Distributionally robust mean-CVaR portfolio optimization with cardinality constraint, *J. Oper. Res. Soc. China*, **14** (2026), 179–209. https://doi.org/10.1007/s40305-023-00512-1
18. D. Bertsimas, R. Cory-Wright, J. Pauphilet, Solving large-scale sparse PCA to certifiable (near) optimality, *J. Mach. Learn. Res.*, **23** (2022), 1–35.
19. D. Bertsimas, J. Lamperski, J. Pauphilet, Certifiably optimal sparse inverse covariance estimation, *Math. Program.*, **184** (2020), 491–530. https://doi.org/10.1007/s10107-019-01419-7
20. J. Gotoh, K. Fujisawa, Convex optimization approaches to maximally predictable portfolio selection, *Optimization*, **63** (2014), 1713–1735. https://doi.org/10.1080/02331934.2012.741237
21. O. Ledoit, M. Wolf, Improved estimation of the covariance matrix of stock returns with an application to portfolio selection, *J. Empir. Finance*, **10** (2003), 603–621. https://doi.org/10.1016/S0927-5398(03)00007-0
22. D. Bertsimas, J. Pauphilet, B. Van Parys, Sparse classification: a scalable discrete optimization perspective, *Mach. Learn.*, **110** (2021), 3177–3209. https://doi.org/10.1007/s10994-021-06085-5
23. H. Saishu, K. Kudo, Y. Takano, Cutting-plane algorithm for estimation of sparse Cox proportional hazards models, *TOP*, **32** (2024), 57–82. https://doi.org/10.1007/s11750-023-00658-4
24. E. F. Fama, K. R. French, Common risk factors in the returns on stocks and bonds, *J. Financ. Econ.*, **33** (1993), 3–56. https://doi.org/10.1016/0304-405X(93)90023-5
25. E. F. Fama, K. R. French, A five-factor asset pricing model, *J. Financ. Econ.*, **116** (2015), 1–22. https://doi.org/10.1016/j.jfineco.2014.10.010
26. M. M. Carhart, On persistence in mutual fund performance, *J. Finance*, **52** (1997), 57–82. https://doi.org/10.1111/j.1540-6261.1997.tb03808.x
27. K. Tanaka, R. Yamamoto, Ellipsoidal buffered area under the curve maximization model with variable selection in credit risk estimation, *Comput. Manag. Sci.*, **20** (2023), 18. https://doi.org/10.1007/s10287-023-00450-6