Cardinality Constrained Portfolio Optimization via Alternating Direction Method of Multipliers

Zhang-Lei Shi, Xiao Peng Li, Chi-Sing Leung, *Senior Member, IEEE*, and Hing Cheung So, *Fellow, IEEE*

***Abstract*— Inspired by sparse learning, the Markowitz mean-variance model with a sparse regularization term is popularly used in sparse portfolio optimization. However, in penalty-based portfolio optimization algorithms, the cardinality level of the resultant portfolio relies on the choice of the regularization parameter. This brief formulates the mean-variance model as a cardinality (ℓ_0-norm) constrained nonconvex optimization problem, in which we can explicitly specify the number of assets in the portfolio. We then use the alternating direction method of multipliers (ADMMs) concept to develop an algorithm to solve the constrained nonconvex problem. Unlike some existing algorithms, the proposed algorithm can explicitly control the portfolio cardinality. In addition, the dynamic behavior of the proposed algorithm is derived. Numerical results on four real-world datasets demonstrate the superiority of our approach over several state-of-the-art algorithms.**

***Index Terms*— ℓ_0-norm, alternating direction method of multipliers (ADMMs), mean-variance model, sparse portfolio.**

I. Introduction

Recently, neural network approaches [1]–[3] are proposed for finance/asset management. For instance, we can use the radial basis function (RBF) model to study the market trend [1]. Portfolio optimization [4]–[7] is one kind of finance/asset management methods. It aims at determining the investment percentages on N assets based on historical data. The percentages form an N-dimensional vector $\boldsymbol{w}$, known as portfolio vector.

Optimizing a portfolio can be viewed as parameter estimation in an adaptive system. Based on historical data, we determine the investment percentages on the selected assets. Afterward, we use the resultant portfolio to perform the investment for an operating period. In the last decades, portfolio optimization has received considerable attention in the machine learning community [2], [8]–[12]. For instance, in [9] and [11], ℓ_1-norm regularization algorithms are proposed. However, the drawback of using the ℓ_1-norm regularization methods is that we cannot explicitly and directly control the number of selected assets in the resultant portfolio.

The Markowitz mean-variance theory [5], [13] is an essential theory to model the return and risk of a portfolio. It aims at constructing a diversified portfolio that balances the return and risk [5], [13]. One research direction in the mean-variance model is to design a robust algorithm that can handle the uncertainty of the estimated model [14]–[16]. For example, in [15], a robust method for estimating a modified covariance matrix is presented. The modified covariance matrix is a linear combination of two covariance matrices, estimated by shrinkage transformation and a random matrix theory-based filter.

Another direction focuses on improving the quality of the investment scheme by adding constraints on the portfolio vector or adding regularization terms into the objective function [17]–[20]. This direction can improve the generalization ability of the portfolio vector. Here, "good generalization ability" means that the portfolio vector has good performance against market volatility.

Since a dense portfolio creates some difficulties in management and has high transaction costs [6], [17], modern portfolio theory focuses on sparse portfolio optimization [6], [21], [22]. Inspired by sparse learning, some portfolio selection algorithms introduce an ℓ_1-norm regularization term into the objective function [17], [18], [23]. The sparse portfolio optimization can be considered as a special form of feature extraction, in which we have a special form of the objective function and some constraints. However, due to the existence of constraints, conventional feature extraction techniques may not be appropriate for sparse portfolio optimization. In addition, the main drawback of using the ℓ_1-norm regularization is that we need to tune the regularization parameter to obtain a plausible portfolio cardinality.

In order to explicitly control the portfolio cardinality, cardinality (ℓ_0-norm) constrained portfolio optimization algorithms are proposed [6], [24], where the cardinality of the portfolio is the number of selected assets. Nevertheless, ℓ_0-norm optimization problems are NP-hard. To circumvent this issue, some frameworks suggest using relaxation or approximation techniques [6], [24]. For instance, given a target cardinality, we can relax the cardinality constraint into an ℓ_1-norm related convex constraint [6]. It is worth noting that, with relaxation, the target cardinality behaves as a sparsity control parameter. Thus, the relaxation algorithm cannot explicitly control the cardinality level. In addition, in the relaxation algorithm, the number of decision variables is N^2, rather than N.

The cardinality constrained portfolio optimization can be recast as a mixed-integer programming (MIP) problem [21], [22]. In the MIP, the sum of binary variables is the desired cardinality level. Thus, there is no sparsity-related parameter to tune. Instead, the MIP needs to tune the upper bound for the absolute value of all elements in the portfolio vector.

The alternating direction method of multipliers (ADMMs) is a popular learning scheme in many applications [25]–[28]. The ADMM decomposes the original problem into several subproblems. The resultant subproblems can be solved efficiently, especially when they have closed-form expressions. The ADMM can be used to determine the size of a neural network. For instance, in [26] and [27], ℓ_1-norm-based ADMM algorithms are developed to construct flat structure neural networks, but they cannot directly and explicitly control the size of the resultant network. In [28], the ADMM concept with an ℓ_0 constraint is used to construct deep neural networks. However, in [28], there is no theoretical study on the convergence and dynamic behaviors. Since there is a sum-to-one constraint in portfolio optimization, the results of [26]–[28] cannot be used in portfolio design.

Manuscript received 23 June 2021; revised 1 February 2022 and 29 April 2022; accepted 6 July 2022. This work was supported by a grant from the Research Grants Council of Hong Kong Special Administrative Region, China, under Project CityU 11207922. *(Corresponding authors: Xiao Peng Li; Chi-Sing Leung.)*

Zhang-Lei Shi is with the College of Science, China University of Petroleum (Huadong), Qingdao 266580, China (e-mail: zlshi@upc.edu.cn).

Xiao Peng Li, Chi-Sing Leung, and Hing Cheung So are with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China (e-mail: x.p.li@my.cityu.edu.hk; eeleungc@cityu.edu.hk; hcso@ee.cityu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3192065.

Digital Object Identifier 10.1109/TNNLS.2022.3192065

In [29], an approximate ℓ_0-norm is developed for sparse index tracking, which is similar to sparse portfolio problems. Besides, the ℓ_1-norm and ℓ_0-norm are acted as a regularization term to control the cardinality level [9], [11], [12], [17], [18]. Based on ADMM, an ℓ_1-regularization method is developed for minimizing the transaction cost [8]. To sum up, those aforementioned methods cannot directly and explicitly control the cardinality level.

This brief proposes an ADMM-based algorithm for sparse portfolio optimization. We use the ADMM concept to decompose the original sparse portfolio optimization into three subproblems. In our formulation, each subproblem has a closed-form solution and our algorithm can explicitly control the number of selected assets. In addition, we theoretically study the convergence behavior. Experiments are conducted on four real-world datasets. The experimental results demonstrate that the proposed algorithm is superior to several ℓ_0- or ℓ_1-norm-based sparse portfolio optimization schemes.

This brief is organized as follows. Portfolio optimization and ADMM concept are described in Section II. In Section III, the proposed ADMM-based algorithm is developed. Numerical results are reported in Section IV. Finally, conclusions are drawn in Section V.

II. Background

A. Notations

We use a lower case or upper case letter to represent a scalar, while vectors and matrices are denoted by bold lower case and upper case letters, respectively. The transpose operator is denoted as $(\cdot)^{\mathrm{T}}$, and $\mathbf{I}$ represents the identity matrix. In addition, $\mathbf{1}$ and $\mathbf{0}$ represent the vector of ones and the vector of zeros, respectively. Other mathematical symbols are defined in their first appearance.

B. Portfolio Optimization

Given N risky assets, let $\mathbf{R} \in \mathbb{R}^{D\times N}$ be the daily return matrix, where each row vector in $\mathbf{R}$ is the return vector of the N assets in a particular day. From the daily return matrix, we can obtain the mean daily return vector as well as the covariance matrix $\mathbf{\Gamma} \in \mathbb{R}^{N\times N}$ of the daily return vectors. In the Markowitz mean-variance model, we usually assume that $\mathbf{\Gamma}$ is positive definite [30]–[32]. When the number of daily return vectors is not large enough, the estimated covariance matrix may be positive semidefinite. In such a situation, the estimated covariance matrix can be modified to positive definite by adding $\epsilon\mathbf{I}$ to the covariance matrix, where ϵ is a small positive number. The classic Markowitz mean-variance model is a constrained quadratic programming problem, given by

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}, \ \text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1 \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^N$ is the portfolio weight vector. The term $\boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w}$ corresponds to the risk, while the term $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}$ is referred to as the return. Parameter $\lambda > 0$ is called the risk parameter. It balances the risk and the return in the model. In general, a larger λ leads to a higher return. When $\lambda = 0$, the model is called global minimum variance portfolio [17], [33], which minimizes the risk only.

Since the solution of (1) is not a sparse vector, modern portfolio optimization methods aim at controlling the portfolio cardinality [6], [21], [34]. To construct a sparse portfolio, one idea is to use regularization methods, i.e., adding an ℓ_0/ℓ_1-norm term into the objective function.

With the ℓ_0-norm regularization [19], [20], the portfolio optimization problem becomes

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} + \beta_0\|\boldsymbol{w}\|_0, \ \text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1 \tag{2}$$

where $\beta_0 > 0$ is the regularization parameter. Due to the NP-hard issue of the ℓ_0-norm minimization, the ℓ_1-norm is widely used to replace the ℓ_0-norm [17], [18]. The sparse portfolio optimization problem then becomes

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} + \beta_1\|\boldsymbol{w}\|_1, \ \text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1 \tag{3}$$

where $\beta_1 > 0$ is the regularization parameter. Another formulation is to constrain the ℓ_1-norm of the portfolio vector [35], given by

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} \ \text{s.t.}\ \ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1 \ \text{and}\ \|\boldsymbol{w}\|_1 \le \theta \tag{4}$$

where $\theta \ge 1$. Note that the formulations in (2)–(4) cannot directly and explicitly control the cardinality level.

To explicitly control the cardinality level, the cardinality constrained portfolio optimization is formulated as

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}, \ \text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1 \ \text{and}\ \|\boldsymbol{w}\|_0 \le K \tag{5}$$

where K is the desired number of nonzero elements in $\boldsymbol{w}$. Note that the problem stated in (5) is nonsmooth and nonconvex.

The cardinality constrained model in (5) can be recast as the following MIP formulation [21], [24], given by

$$\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\mathbf{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} \tag{6a}$$

$$\text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{1} = 1, \ \boldsymbol{e}^{\mathrm{T}}\mathbf{1} \le K \tag{6b}$$

$$-\xi e_i \le w_i \le \xi e_i, \quad i = 1, \ldots, N \tag{6c}$$

$$e_i \in \{0, 1\}, \quad i = 1, \ldots, N \tag{6d}$$

where $\xi > 0$ is a large number that represents an upper bound for the absolute value of all elements in the optimal solution to model (6). In general, solving (6) requires the combination of a continuous optimization procedure and an integer programming procedure. The MIP algorithm [21], [24] involves an exhaustive search procedure.

In [6], a relaxation method is proposed. It transforms (5) as a convex semidefinite programming problem [36], given by

$$\min_{\mathbf{W}} \text{Tr}(\mathbf{\Gamma}\mathbf{W}) - \lambda\mathbf{1}^{\mathrm{T}}\mathbf{W}\boldsymbol{u} \tag{7a}$$

$$\text{s.t.}\ \text{Tr}(\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathbf{W}) = 1, \ \|\mathbf{W}\|_1 \le K\text{Tr}(\mathbf{W}), \ \mathbf{W} \in \mathbb{S}_N^+ \tag{7b}$$

where $\mathbf{W}$ is an $N \times N$ matrix, $\|\mathbf{W}\|_1$ is the sum of the absolute values of the matrix elements, and $\mathbb{S}_N^+$ represents the set of positive semidefinite matrices with dimensions $N \times N$. The resultant portfolio is obtained from the eigenvector corresponding to the largest eigenvalue of the solution of (7). The major problem of this formulation is that the resultant portfolio may not be sparse and its cardinality level may be greater than K. Also, in this positive semidefinite formulation, the number of decision variables is N^2. Thus, this formulation is not suitable for large N.

C. ADMM

The ADMM algorithm [25], [28], [37] addresses the following optimization problem:

$$\min_{\boldsymbol{x},\boldsymbol{y}} f(\boldsymbol{x}) + h(\boldsymbol{y}), \ \text{s.t.}\ \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{y} = \boldsymbol{c} \tag{8}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^m$ contain the decision variables, $\boldsymbol{c} \in \mathbb{R}^d$ is a constant vector, and $\mathbf{A} \in \mathbb{R}^{d\times n}$ and $\mathbf{B} \in \mathbb{R}^{d\times m}$. In ADMM, a Lagrangian function $\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\gamma})$ is first defined, given by

$$\mathcal{L}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{\gamma}) = f(\boldsymbol{x}) + h(\boldsymbol{y}) + \boldsymbol{\gamma}^{\mathrm{T}}(\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{y} - \boldsymbol{c}) + \frac{\rho}{2}\|\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{y} - \boldsymbol{c}\|_2^2 \tag{9}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^d$ is the Lagrange multiplier and $\rho > 0$. The general ADMM scheme is given by

$$\boldsymbol{x}^{t+1} = \arg\min_{\boldsymbol{x}} \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{y}^t, \boldsymbol{\gamma}^t\right) \tag{10a}$$

$$\boldsymbol{y}^{t+1} = \arg\min_{\boldsymbol{y}} \mathcal{L}\left(\boldsymbol{x}^{t+1}, \boldsymbol{y}, \boldsymbol{\gamma}^t\right) \tag{10b}$$

$$\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^t + \rho\left(\mathbf{A}\boldsymbol{x}^{t+1} + \mathbf{B}\boldsymbol{y}^{t+1} - \boldsymbol{c}\right). \tag{10c}$$

At each iteration, we need to solve the three subproblems 10(a)–10(c) sequentially. In [38] and [39], it is shown that the $\boldsymbol{\gamma}$-update (10c) can be generalized into

$$\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^{t} + s\rho(\mathbf{A}\boldsymbol{x}^{t+1} + \mathbf{B}\boldsymbol{y}^{t+1} - \boldsymbol{c}) \tag{11}$$

where $s \in (0, ((5)^{1/2} + 1/2))$. For many situations, $s = 1$.

There are three important issues in the ADMM. The first issue is whether the optimal solutions for the subproblems stated in (10a) and (10b) can be found or not. The second issue is whether the optimal solutions of the subproblems stated in (10a) and (10b) have closed-form solutions or not. The last one is whether the three alternating steps stated in (10) converge or not. It should be noticed that the first and second issues address two different aspects. The first issue focuses on the existence of methods to find the optimal solutions to the subproblems. The methods may be based on iterative algorithms or closed-form formulas. The second issue focuses on closed-form solutions.

III. ADMM for Cardinality Constrained Mean-Variance Portfolio Optimization

A. Algorithm Development

This section develops our ADMM-based algorithm for (5). First, we utilize the quadratic penalty method [40], [41] to address the equality constraint. In this case, the new objective function is $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} + (C/2)(\boldsymbol{w}^{\mathrm{T}}\mathbf{1} - 1)^2$. According to the quadratic penalty method, $C > 0$ cannot be too small. Otherwise, the constraint may be violated. Here, the meaning of "large" is related to the magnitudes of $\boldsymbol{\Gamma}$ and $\boldsymbol{u}$. For a large enough C, the resultant solution is close to the optimal solution of the original problem. For our application and the datasets, since the magnitudes of $\boldsymbol{\Gamma}$ and $\boldsymbol{u}$ are small, we find that $C = 1$ is sufficiently large. With the quadratic penalty method, the model (5) becomes

$$\min_{\boldsymbol{w},\boldsymbol{z}} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} + \frac{C}{2}(\boldsymbol{w}^{\mathrm{T}}\mathbf{1}-1)^2 + \mathcal{I}(\boldsymbol{z}),\ \text{s.t. } \boldsymbol{w} = \boldsymbol{z} \tag{12}$$

where $\mathcal{I}(\boldsymbol{z})$ is an indicator function. For the indicator function, if $\|\boldsymbol{z}\|_0 \le K$, then $\mathcal{I}(\boldsymbol{z}) = 0$. Otherwise, $\mathcal{I}(\boldsymbol{z}) = +\infty$.

Let $\mathcal{F}(\boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w} + (C/2)(\boldsymbol{w}^{\mathrm{T}}\mathbf{1} - 1)^2$. The Lagrangian function of (12) is given by

$$\mathcal{L}(\boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\gamma}) = \mathcal{F}(\boldsymbol{w}) + (\boldsymbol{w}-\boldsymbol{z})^{\mathrm{T}}\boldsymbol{\gamma} + \frac{\rho}{2}||\boldsymbol{w}-\boldsymbol{z}||_2^2 + \mathcal{I}(\boldsymbol{z}) \tag{13}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^N$ is the Lagrange multiplier vector and $\rho > 0$ is the penalty parameter. According to ADMM, the algorithm is formulated as the following three alternating steps, given by

$$\boldsymbol{z}^{t+1} = \arg\min_{\boldsymbol{z}} \mathcal{L}\left(\boldsymbol{w}^{t}, \boldsymbol{z}, \boldsymbol{\gamma}^{t}\right) \tag{14a}$$

$$\boldsymbol{w}^{t+1} = \arg\min_{\boldsymbol{w}} \mathcal{L}\left(\boldsymbol{w}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t}\right) \tag{14b}$$

$$\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^{t} + \rho\left(\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1}\right). \tag{14c}$$

We call the three updating steps in (14a)–(14c) as ℓ_0-ADMM.

1) Development of $\boldsymbol{z}$-Update: To solve (14a), we can ignore all the constant terms that do not include $\boldsymbol{z}$ in (13). The $\boldsymbol{z}$-update is then reduced to

$$\begin{aligned}\boldsymbol{z}^{t+1} &= \arg\min_{\boldsymbol{z}} \mathcal{F}(\boldsymbol{w}^{t}) + \left(\boldsymbol{w}^{t} - \boldsymbol{z}\right)^{\mathrm{T}}\boldsymbol{\gamma}^{t} + \frac{\rho}{2}\left|\left|\boldsymbol{w}^{t} - \boldsymbol{z}\right|\right|_2^2 + \mathcal{I}(\boldsymbol{z}) \\ &= \arg\min_{\boldsymbol{z}} \frac{\rho}{2}\left|\left|\boldsymbol{w}^{t} - \boldsymbol{z} + \frac{\boldsymbol{\gamma}^{t}}{\rho}\right|\right|_2^2 + \mathcal{I}(\boldsymbol{z}).\end{aligned} \tag{15}$$

Define $\boldsymbol{\delta} = \boldsymbol{w}^{t} + \boldsymbol{\gamma}^{t}/\rho$. Hence, solving (15) is equivalent to solving the following optimization problem, given by:

$$\min_{\boldsymbol{z}} \|\boldsymbol{\delta} - \boldsymbol{z}\|_2^2 \ \text{ s.t. } \|\boldsymbol{z}\|_0 \le K. \tag{16}$$

Let Λ be the index set that indicates the nonzero elements of $\boldsymbol{z}$. Also, let Λ^C be another index set that indicates the zero elements of $\boldsymbol{z}$. The objective value $g(\boldsymbol{z})$ of (16) is then given by

$$g(\boldsymbol{z}) = \sum_{i\in\Lambda}(z_i - \delta_i)^2 + \sum_{i'\in\Lambda^C}(z_{i'} - \delta_{i'})^2. \tag{17}$$

For $i' \in \Lambda^C$, we have $z_{i'} = 0$. In addition, in minimizing $g(\boldsymbol{z})$, for $i \in \Lambda$, we should set $z_i = \delta_i$. Thus, we have

$$g(\boldsymbol{z}) = \sum_{i'\in\Lambda^C} \delta_{i'}^2. \tag{18}$$

To minimize $g(\boldsymbol{z})$, the index set Λ^C should contain the indices of the $N - K$ smallest (in absolute value) components of $\boldsymbol{\delta}$. That is, the index set Λ should contain the indices of the K largest (in absolute value) components of $\boldsymbol{\delta}$. Thereby, the solution to (16) is

$$\boldsymbol{z}^{t+1} = \boldsymbol{H}_K(\boldsymbol{\delta}) \tag{19}$$

where $\boldsymbol{H}_K$ is an elementwise hard thresholding operator

$$\boldsymbol{H}_K(\delta_i) = \begin{cases} 0, & \text{if } |\delta_i| < q \\ \delta_i, & \text{if } |\delta_i| \ge q. \end{cases} \tag{20}$$

In (20), q is the Kth largest element of $\{|\delta_1|, \ldots, |\delta_N|\}$. If there are less than K nonzero elements in $\boldsymbol{\delta}$, then q is the smallest nonzero element of $\{|\delta_1|, \ldots, |\delta_N|\}$.

2) Development of $\boldsymbol{w}$-Update: To solve (14b), we consider the following problem:

$$\boldsymbol{w}^{t+1} = \arg\min_{\boldsymbol{w}} \mathcal{F}(\boldsymbol{w}) + \left(\boldsymbol{w} - \boldsymbol{z}^{t+1}\right)^{\mathrm{T}}\boldsymbol{\gamma}^{t} + \frac{\rho}{2}\left|\left|\boldsymbol{w} - \boldsymbol{z}^{t+1}\right|\right|_2^2 + \mathcal{I}(\boldsymbol{z}^{t+1}). \tag{21}$$

In the above-mentioned problem, $\boldsymbol{z}^{t+1}$ and $\boldsymbol{\gamma}^{t}$ are considered as constants. Thus,

$$\begin{aligned}\boldsymbol{w}^{t+1} &= \arg\min_{\boldsymbol{w}} \mathcal{F}(\boldsymbol{w}) + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\gamma}^{t} + \frac{\rho}{2}\left|\left|\boldsymbol{w} - \boldsymbol{z}^{t+1}\right|\right|_2^2 \\ &= \arg\min_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{w} - \left(\lambda\boldsymbol{u} + \rho\boldsymbol{z}^{t+1} - \boldsymbol{\gamma}^{t} + C\mathbf{1}\right)^{\mathrm{T}}\boldsymbol{w}\end{aligned} \tag{22}$$

where $\boldsymbol{\Phi} = \boldsymbol{\Gamma} + \frac{\rho}{2}\mathbf{I} + (C/2)\mathbf{1}\mathbf{1}^{\mathrm{T}}$. Since $\boldsymbol{\Phi}$ is positive definite, (21) has the optimal solution, given by

$$\boldsymbol{w}^{t+1} = (2\boldsymbol{\Gamma} + \rho\mathbf{I} + C\mathbf{1}\mathbf{1}^{\mathrm{T}})^{-1}(\lambda\boldsymbol{u} + \rho\boldsymbol{z}^{t+1} - \boldsymbol{\gamma}^{t} + C\mathbf{1}). \tag{23}$$

In addition, as $\boldsymbol{\Phi}$ is positive definite, we have

$$\nabla\mathcal{F}(\boldsymbol{w}^{t+1}) + \boldsymbol{\gamma}^{t} + \rho(\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1}) = \mathbf{0}. \tag{24}$$

3) $\boldsymbol{\gamma}$-Update: We can update $\boldsymbol{\gamma}^{t+1}$ according to (14c), given by

$$\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^{t} + \rho(\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1}). \tag{25}$$

4) Choice of ρ: In the proposed algorithm, ρ is able to affect the convergence speed [25], [42], [43]. It can be set to a fixed constant or an adaptive value according to the following scheme [42]–[44]:

$$\rho^{t+1} = \min\left\{\alpha\rho^{t}, \rho_{\max}\right\} \tag{26}$$

where $0 < \alpha < 2$ is the scaling parameter and $\rho_{\max}$ is the upper bound for ρ. Recent research [43], [45], [46] shows that an adaptive strategy on ρ can reduce the required number of iterations for convergence. As ℓ_0-norm problems are nonconvex and nonsmooth, algorithms usually lead to a suboptimal solution. With an adaptive strategy, the objective value of the solution vector is better. In our case, the objective value is $\mathcal{F}(\boldsymbol{w}^*)$ and the solution vector is $\boldsymbol{w}^*$. It is worth mentioning that the convergence proof under this adaptive scheme remains challenging in nonconvex situations [43], [45].

5) Summary of the Algorithm: We summarize the developed ℓ_0-ADMM for portfolio optimization in Algorithm 1. It should be noticed that all three update steps have closed-form expressions.

Algorithm 1 ℓ_0-ADMM for Portfolio Optimization

Input: $\boldsymbol{\Gamma}$, $\boldsymbol{u}$, K, λ
Initialize: C, ρ^0, ρ_{max}, α, $\boldsymbol{w}^0$, $\boldsymbol{z}^0$, $\boldsymbol{\gamma}^0$, t
while not converge **do**
(1) $\boldsymbol{z}$-update: $\boldsymbol{z}^{t+1} = \boldsymbol{H}_K(\boldsymbol{\delta})$, where $\boldsymbol{\delta} = \boldsymbol{w}^t + \boldsymbol{\gamma}^t/\rho^t$.
(2) $\boldsymbol{w}$-update: $\boldsymbol{w}^{t+1} = (2\boldsymbol{\Gamma}+\rho^t\mathbf{I}+C\mathbf{1}\mathbf{1}^{\mathrm{T}})^{-1}(\lambda\boldsymbol{u}+\rho^t\boldsymbol{z}^{t+1}-\boldsymbol{\gamma}^t+C\mathbf{1})$.
(3) $\boldsymbol{\gamma}$-update: $\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^t + \rho^t(\boldsymbol{w}^{t+1}-\boldsymbol{z}^{t+1})$
(4) ρ-update: $\rho^{t+1} = \min\{\alpha\rho^t, \rho_{max}\}$
(5) $t = t+1$
end while
Output: $\boldsymbol{w}^*$

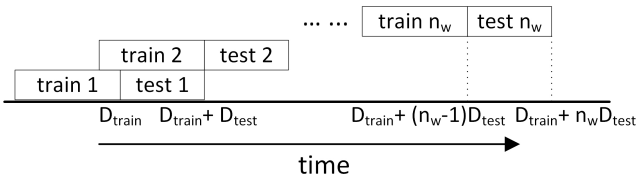

Fig. 1. Illustration of rolling windows.

B. Convergence Behavior

This section presents the convergence behavior of the proposed ℓ_0-ADMM. First, in ℓ_0-ADMM, we have the following two properties.

P1: For each t, there exists an $\eta > 0$ such that

$$\mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big) \leq -\eta\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2. \quad (27)$$

P2: $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ is lower bounded.

Proof: The proofs of **P1** and **P2** are given in Appendix A. ■

Based on **P1** and **P2**, we get Theorem 1.

Theorem 1: Since the suggested ℓ_0-ADMM satisfies **P1** and **P2**, $\{\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)\}$ converges.

Proof: Based on **P1**, $\{\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)\}$ is monotonically nonincreasing. From **P2**, $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ is lower bounded. Thus, the convergence of $\{\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)\}$ can be guaranteed. ■

Besides, the dynamic behavior of the sequence $\{\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\}$ is provided in Theorem 2.

Theorem 2: As $t \to \infty$, $\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|^2 \to 0$, $\|\boldsymbol{z}^{t+1} - \boldsymbol{z}^t\|^2 \to 0$, and $\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|^2 \to 0$.

Proof: The proof is given in Appendix B. ■

IV. Experimental Results and Discussion

A. Settings

1) Datasets and Rolling Window: Four well-known datasets are considered and they are Nasdaq 100, S&P 500, Russell 1000, and Russell 2000. The data involve 1699 trading days from May 1, 2009 to January 29, 2016 and are extracted from Yahoo Finance. Following the common practice, suspended and newly enlisted assets within the time period are excluded [2], [47]. In the experiments, we consider the rolling window concept [2], [17] shown in Fig. 1. We use a training window to create a portfolio and then test its performance with the associated test window. The size of the training window, denoted as D_{train}, is set to 500 days. The test window size, denoted as D_{test}, is 60 days [15]. We call the test window size as a rebalancing period. Details of the four datasets are summarized in Table I.

TABLE I
Details of Four Datasets. for All Datasets, Time Period Is From May 1, 2009 to January 29, 2016

Dataset	N	K range	Total days	D_{train}	D_{test}
Nasdaq 100	76	[30,60]	1699	500	60
S&P 500	414	[30,90]	1699	500	60
Russell 1000	652	[30,90]	1699	500	60
Russell 2000	893	[30,90]	1699	500	60

TABLE II
Details and Properties of All Comparison Methods

Algorithm	Basic idea	Explicitly control sparsity?
MIP [21]	ℓ_0-norm constraint	✓
ℓ_1-NC [35]	ℓ_1-norm constraint	×
GSRP [20]	ℓ_0-norm regularization	×
ℓ_1-Bregman [18]	ℓ_1-norm regularization	×
ℓ_1-ADMM [17]	ℓ_1-norm regularization	×
ℓ_0-ADMM (Ours)	ℓ_0-norm constraint	✓

2) Risk Parameter and Portfolio Size: To compare the performance under different risk situations, we select $\lambda = \{0.001, 0.005\}$. For the Nasdaq 100 dataset, we vary the portfolio size K from 30 to 60. For the other datasets, we vary K from 30 to 90.

3) Comparison Algorithms: We implement five comparison algorithms. They are ℓ_1-ADMM [17] and ℓ_1-Bregman[1] [18] [see (3)], ℓ_1-norm-constrained (ℓ_1-NC) [35] [see (4)], generalized sparse risk parity (GSRP) [20] [see (2)], and MIP [21] (see (6)]. Note that except for the MIP, we cannot explicitly control the resultant cardinality level. That is, in the ℓ_1-ADMM, ℓ_1-Bregman, ℓ_1-NC, and GSRP, we need to tune their regularization parameters to meet the desired cardinality level. Table II summarizes the details of the comparison algorithms and our method.

In [6], the relaxation algorithm in (7) has a much higher computational and space complexities. It involves N^2 decision variables, while the comparison algorithms and our ℓ_0-ADMM involve N decision variables only. In addition, the relaxation algorithm cannot explicitly control the cardinality level. Therefore, we do not include this relaxation algorithm as a comparison algorithm.

4) Parameter Setting: For ℓ_1-Bregman, ℓ_1-ADMM, and our ℓ_0-ADMM, the initial values of decision variables are set to zero. For our algorithm, we set $\rho^0 = 0.0004$, $\rho_{max} = 20$, and $\alpha = 1.2$. Besides, $C = 1$ is selected empirically for the proposed algorithm. For the three methods, the maximum number of iterations is 100. For ℓ_1-ADMM and our ℓ_0-ADMM, if $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2 < 10^{-4}\|\boldsymbol{w}^{t-1}\|_2$, then the algorithms stop. For ℓ_1-Bregman, if $|\mathbf{1}^{\mathrm{T}}\boldsymbol{w}^t - 1| < 5 \times 10^{-6}$ the algorithm stops. For ℓ_1-NC, GSRP, and MIP, we use the default settings in Mosek [48] or CVX [49].

B. Performance Measurement

Two well-known measurements are used for evaluation. One is the out-of-sample mean return (OSMR), denoted as μ, that is, the mean return of test periods. For the τth testing window, let $\boldsymbol{r}_\tau \in \mathbb{R}^N$ be the return vector over the testing period, where $[\boldsymbol{r}_\tau]_i$ is the return for holding the ith assets for D_{test} days. The OSMR is defined as

$$\mu = \frac{1}{T}\sum_{\tau=1}^{T} \boldsymbol{w}_\tau^{\mathrm{T}}\boldsymbol{r}_\tau \quad (28)$$

where T is the number of testing periods.

[1]The ℓ_1-regularized subproblem of ℓ_1-Bregman is solved by ADMM.

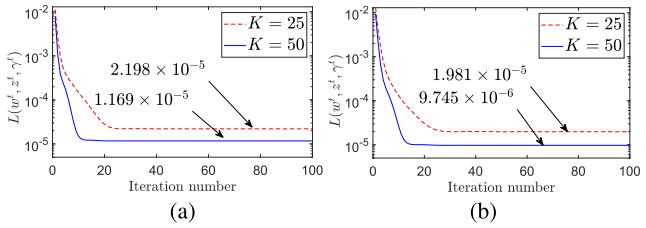

Fig. 2. Convergence of $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ in ℓ_0-ADMM. (a) S&P 500. (b) Russell 1000.

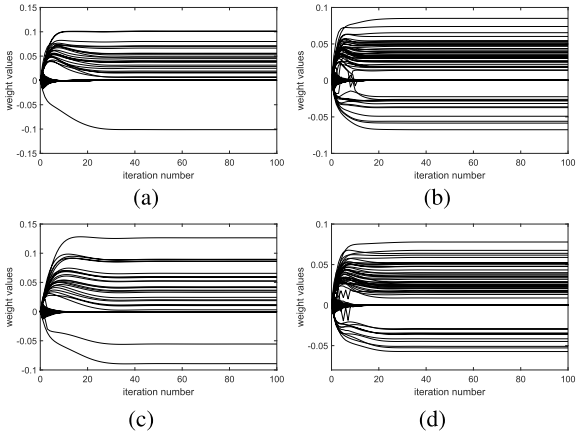

Fig. 3. Dynamics of weight values. (a) S&P 500 ($K = 25$). (b) S&P 500 ($K = 50$). (c) Russell 1000 ($K = 25$). (d) Russell 1000 ($K = 50$).

Another one is the out-of-sample Sharpe ratio (OSSR), denoted as $\mathcal{S}$. In finance management, a higher return usually results in a higher risk (variation of the returns). The OSSR [50] is an indicator that balances the risk and return, given by

$$\mathcal{S} = \frac{\mu}{\sigma}, \text{ where } \sigma = \sqrt{\frac{1}{T-1}\sum_{\tau=1}^{T}\left(\boldsymbol{w}_\tau^{\mathrm{T}}\boldsymbol{r}_\tau - \mu\right)^2} \tag{29}$$

where σ is the standard derivation of out-of-sample returns, i.e., the variation of returns.

In finance management, for two portfolios with a similar return, we should select the one with a higher Sharpe ratio. Similarly, for two portfolios with a similar Sharpe ratio, we should select the one with a higher return.

C. Convergence Behavior

This section uses empirical results to verify Theorems 1 and 2. We consider the S&P 500 and Russell 1000 datasets with $K = 25$ and $K = 50$. In Theorem 1, we theoretically show that in our ℓ_0-ADMM, $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ converges. Fig. 2 depicts its convergence behavior. In terms of $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$, our algorithm converges within around 60 iterations and the value of $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ decreases with number of iterations. The above-mentioned behavior confirms Theorem 1.

In Theorem 2, we theoretically show that as $t \to \infty$, $\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|^2 \to 0$, $\|\boldsymbol{z}^{t+1} - \boldsymbol{z}^t\|^2 \to 0$, and $\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|^2 \to 0$. Fig. 3 shows the dynamics of the estimated weights. From the Fig. 3, after around 60 iterations, there are no big changes in the estimated weights. The above-mentioned behavior confirms Theorem 2. Since the estimated weights can be negative, the vertical axis cannot be in the logarithmic scale.

In addition, we present the convergence rates of ℓ_0-ADMM and ℓ_1-ADMM in Fig. 4. Since there is no $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ in the ℓ_1-ADMM

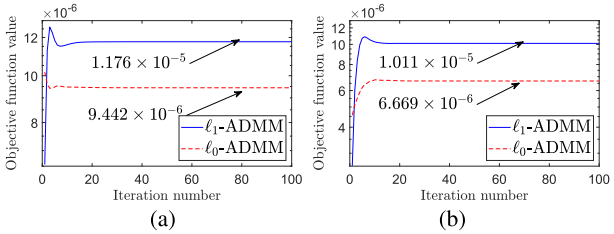

Fig. 4. Convergence behaviors of ℓ_0-ADMM and ℓ_1-ADMM. (a) S&P 500. (b) Russell 1000.

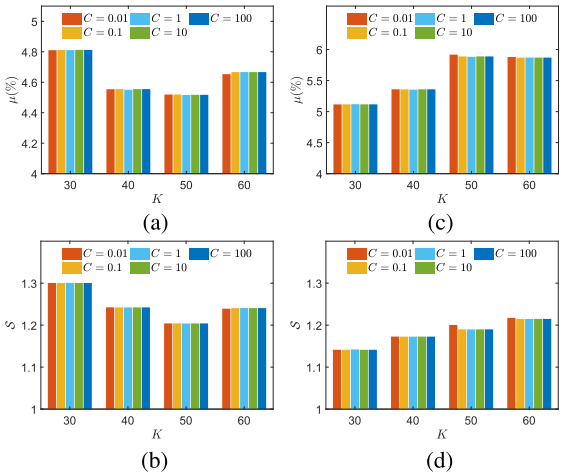

Fig. 5. Influence of C on S&P 500 dataset. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

algorithm, we show their objective function values, i.e., $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}$. We see that the two ADMM-based algorithms have a similar convergence speed.

D. Influence of Parameter C

In our formulation, there is a penalty parameter C. This section investigates the influence of C. We test $C \in \{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2\}$ on various cardinality levels. The results of the S&P 500 dataset are reported in Fig. 5. According to the experimental results, at the same cardinality level, there are a few changes in the performance over different C values. Other datasets have similar behavior.

E. Influences of Portfolio Cardinality and Risk Parameter

The sparse portfolio optimization is a multiobjective problem. That is, a good portfolio should be with a small cardinality level, a high return, and a high Sharpe ratio. This section studies the behaviors of our method at various cardinality levels and risk parameter values.

We consider four risk parameter values and a number of cardinality levels. The results are depicted in Figs. 6 and 7 for S&P 500 and Russell 1000, respectively. From Figs. 6 and 7, we have the following observations.

1) *Cardinality:* For the same risk parameter value λ, there is no general trend on returns and Sharpe ratio values for various cardinality levels. It is worth noting that a large cardinality level (large K) leads to high transaction costs and creates difficulties in management. As a result, we should consider using a portfolio of small cardinality in practice. From Figs. 6(b) and 7(b), even for small cardinalities like 30 and 35, the Sharpe ratios of our approach are still larger than 1.

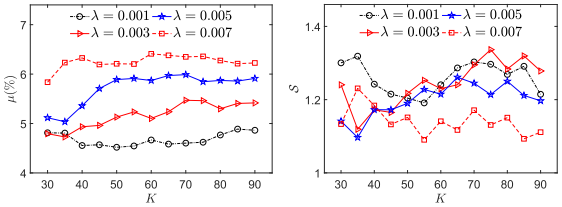

Fig. 6. Performance of our ℓ_0-ADMM for S&P 500 dataset. (a) Return. (b) Sharpe ratio.

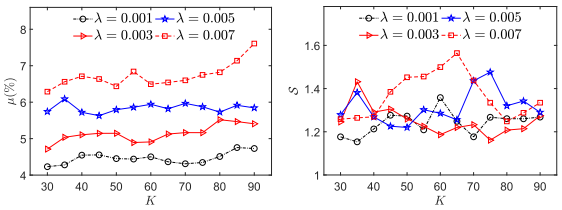

Fig. 7. Performance of our ℓ_0-ADMM for Russell 1000 dataset. (a) Return. (b) Sharpe ratio.

2) *Risk Parameter:* From Figs. 6 and 7, in general, for a given cardinality level, a larger λ leads to a higher return value. However, there is no general trend on Sharpe ratio values for various λ values.

Since the sparse portfolio optimization is a multiobjective problem, the choices of λ and cardinality level depend on the investor's preference. For instance, in the S&P 500 dataset (Fig. 6), if the investor would like to focus on the return first, he/she may choose $\lambda = 0.007$ and the cardinality level equal to 60. With such a choice, we have a 6.41% return and the Sharpe ratio is equal to 1.14. Or he/she may choose $\lambda = 0.007$ and the cardinality level is equal to 35. With such a choice, we have the highest Sharpe ratio.

On the other hand, if the investor would like to focus on the Sharpe ratio first, he/she may choose $\lambda = 0.003$ and the cardinality level is equal to 75. In this setting, we have the highest Sharpe ratio around 1.34 and a reasonable return around 5.46%.

F. Influence of Parameter s

One might argue that from (11) when we gradually increase ρ from a small value, scaling $\boldsymbol{\gamma}$ at the same time might improve the algorithm performance. To investigate this, we conduct experiments on the S&P 500 dataset with $s \in \{0.2, 0.4, 0.6, 0.8, 1\}$. The results are reported in Fig. 8. From the figure, there are no conclusive trends. For example, from Figs. 8(a) and (b), with $\lambda = 0.001$, when $K \in \{40, 50\}$, setting s to 0.2 provides the largest values of μ and $\mathcal{S}$. However, for $K \in \{30, 60\}$, the best value of s is 1. For other settings and other datasets, we also cannot make conclusive trends on the influence of parameter s. Therefore, in this brief, we set $s = 1$ for the proposed ℓ_0-ADMM.

G. Performance Comparison

This section compares the proposed ℓ_0-ADMM with five comparison algorithms: ℓ_1-ADMM [17], ℓ_1-Bregman [18], ℓ_1-NC [35], GSRP [20], and MIP [21]. Note that except for the ℓ_0-ADMM and MIP, we need to tune the regularization parameter or constraint parameter, such that the cardinality meets the desired value. The results of different methods are depicted in Figs. 9–12. Before making a detailed discussion, we provide the following overview.

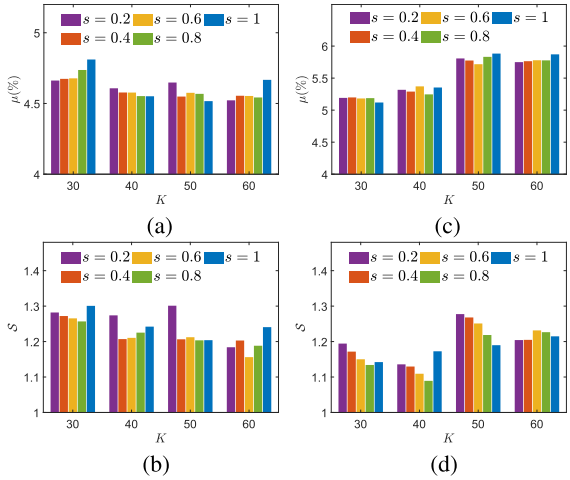

Fig. 8. Influence of s. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

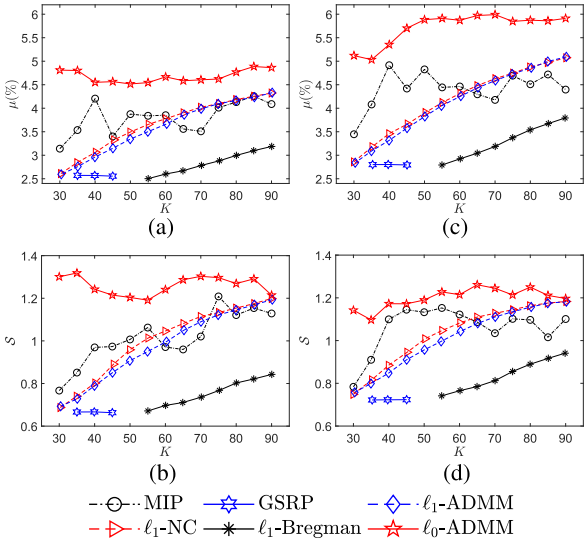

Fig. 9. Performance comparison on S&P 500 with $D_{\text{test}} = 60$. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

1) In general, under the same portfolio cardinality, our ℓ_0-ADMM has a higher return and Sharpe ratio.
2) The performance of the ℓ_1-ADMM and ℓ_1-NC are comparable.
3) In the ℓ_1-Bregman, ℓ_1-NC, and GSRP, we can tune the regularization parameter or the constraint parameter to control the cardinality level. However, not all cardinality levels can be achieved. That is, in some cases, no matter how we tune the parameters, we cannot achieve the desired cardinality levels. For example, as shown in Fig. 9, the achievable cardinality levels of GSRP are from 30 to 50, while the achievable cardinality levels of ℓ_1-Bregman are from 50 to 90. Also, as shown in Fig. 12, the achievable cardinality levels of ℓ_1-NC are from 70 to 90. Based on the above-mentioned observation, in the rest of this section, we mainly compare the ℓ_0-ADMM with ℓ_1-ADMM and MIP.

1) Cardinality: Now, we discuss the performance of different algorithms when fixing λ values under different cardinality values.

We consider the S&P 500 dataset with $\lambda = 0.001$. From Fig. 9(a), the return of our ℓ_0-ADMM is around 4.8% for all cardinality levels. However, when the ℓ_1-ADMM is used, in order to have around 4%

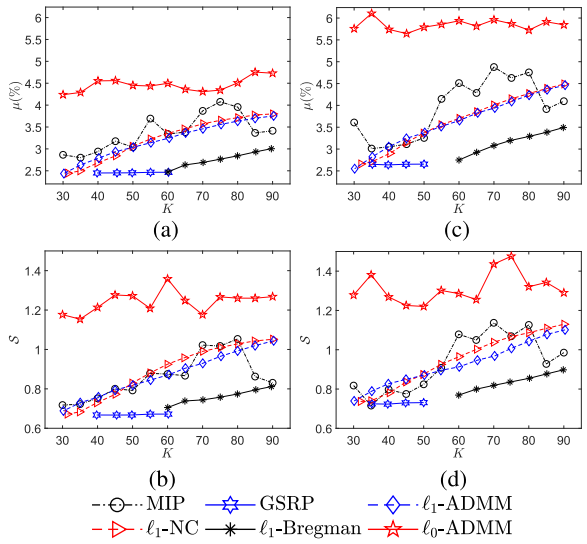

Fig. 10. Performance comparison on Russell 1000 with $D_{\text{test}} = 60$. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

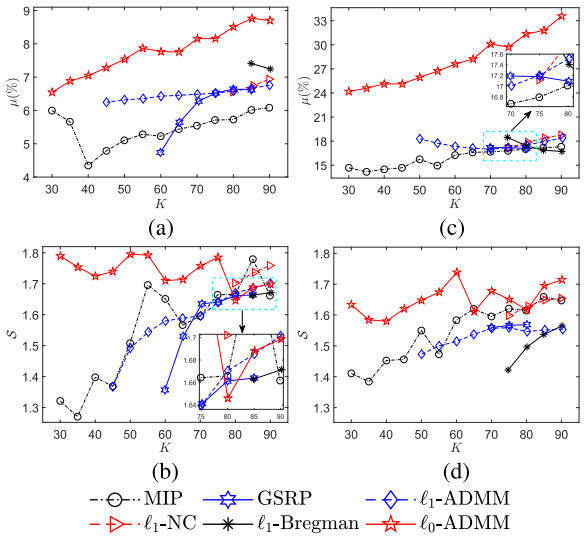

Fig. 12. Performance comparison on Russell 2000 with $D_{\text{test}} = 60$. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

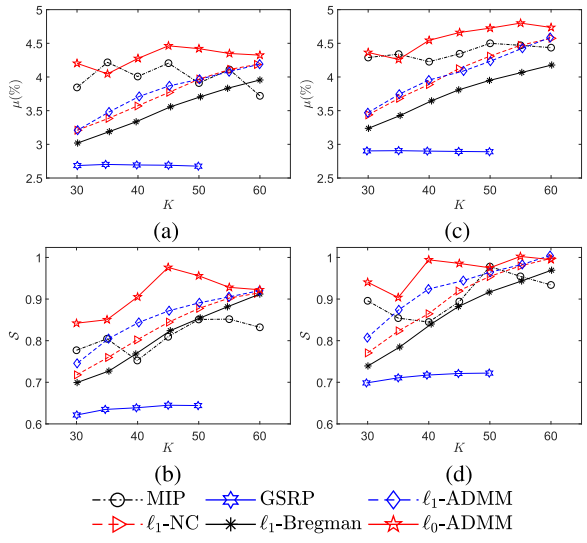

Fig. 11. Performance comparison on Nasdaq 100 with $D_{\text{test}} = 60$. (a) μ with $\lambda = 0.001$. (b) $\mathcal{S}$ with $\lambda = 0.001$. (c) μ with $\lambda = 0.005$. (d) $\mathcal{S}$ with $\lambda = 0.005$.

return, we need to increase the cardinality level to 90. When the MIP is used, in order to have around 4.5% return, we need to increase the cardinality level to 45. For the GSRP, the return is around 2.55% only. For the ℓ_1-Bregman, the largest return value is around 3.187% at the cardinality level equal to 90. In addition, our ℓ_0-ADMM has better Sharpe ratio values, as shown in Fig. 9(b).

On the Russell 1000 dataset with $\lambda = 0.005$, we also observe that our ℓ_0-ADMM has a better performance. From Fig. 10(c), the return of our ℓ_0-ADMM is around 5.7% for all cardinality levels. In particular, when the cardinality level is 35, the return and the Sharpe ratio of our method are 6.1% and 1.4, respectively. For the comparison algorithms, the MIP provides the best performance at the cardinality level equal to 70. However, at this cardinality level, the MIP provides a 4.8% return only and its Sharpe ratio is equal to 1.137. For the ℓ_1-ADMM, when the cardinality level is 90, its return and the Sharpe ratio are quite low. For the ℓ_1-NC, when the cardinality level is 90, its return and the Sharpe ratio are 4.491% and 1.13, respectively.

For the Nasdaq 100 and Russell 2000 datasets, the results are reported in Figs. 11 and 12, respectively. In general, our algorithm has better return and Sharpe ratios. In a few cases, the comparison methods are comparable to or a bit better than our ℓ_0-ADMM. For example, in the Nasdaq 100 dataset with $\lambda = 0.001$ [Fig. 11(a) and (b)], when we fix the cardinality level to 35, the return of the MIP is slightly higher than that of our ℓ_0-ADMM. However, at this cardinality level, our ℓ_0-ADMM can achieve a higher Sharpe ratio.

Another example is in the Russell 2000 dataset with $\lambda = 0.001$, as shown in Figs. 12(a) and (b). From Fig. 12(b), when the cardinality level is 85, the Sharpe ratio of the MIP is 1.779, which is a bit higher than that of our ℓ_0-ADMM. However, at this cardinality level, our ℓ_0-ADMM has a much better return, as shown in Fig. 12(a).

2) Risk Parameter: Risk parameter λ balances the return and risk. We use the results on the S&P 500 dataset (Fig. 9) for discussion. In general, a larger λ leads to a better return.

From Fig. 9, for $\lambda = 0.001$ and $K = 30$, the return of our ℓ_0-ADMM is 4.811% and the Sharpe ratio is 1.300, while the return of MIP is 3.139% and the Sharpe ratio is 0.768. When we increase λ to 0.005 with $K = 30$, the return of our algorithm increases to 5.119% and the Sharpe ratio slightly decreases to 1.143, while the return of ℓ_1-ADMM increases to 3.446% only and the Sharpe ratio is 0.783.

For both $\lambda = 0.001$ and $\lambda = 0.005$, the largest returns of the MIP are achieved at $K = 40$. However, with $\lambda = 0.001$, the return of the MIP is 4.205%, which is lower than that of ℓ_0-ADMM. When $\lambda = 0.005$, the profit of MIP is 4.915%, while our algorithm has a 5.353% return.

For $\lambda = 0.001$ and $K = 90$, the return of our algorithm is 4.863%, while the return of ℓ_1-ADMM is 4.331%. When we increase λ to 0.005, the return of our algorithm is 5.912%, while the return of ℓ_1-ADMM is 5.094%.

It should be noticed that in the above-mentioned discussion cases, our method has better Sharpe ratio values. Also, similar behaviors are observed in other datasets.

H. Difference Between ℓ_0-ADMM and ℓ_1-ADMM

For ℓ_1-ADMM and our ℓ_0-ADMM, both of them use the ADMM concept to construct the training process. From the experimental results, our ℓ_0-ADMM can construct a better sparse portfolio. In addition, the advantage of using our approach is that we can explicitly control the cardinality level.

The ℓ_1-ADMM has poor performance, especially at low cardinality levels. The reason is that, in model (3), $\beta_1\|\boldsymbol{w}\|_1$ is a penalty term and it has two effects. The first one is the resultant cardinality level. Also, it controls the relative importance between $\|\boldsymbol{w}\|_1$ and the original objective $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}$. To obtain a sparser portfolio, we need to use a large β_1. That is, the weighting of $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{w} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{w}$ is very small. Hence, at a low cardinality level, we obtain a low return μ and Sharpe ratio $\mathcal{S}$. In addition, there is no direct relationship between β_1 and the resultant cardinality level. To obtain a specific cardinality level, we need to try a number of β_1 values.

V. Conclusion

In this brief, we propose an ADMM-based algorithm for the mean-variance portfolio optimization problem based on the ℓ_0-norm constraint. The algorithm is able to explicitly control the portfolio cardinality. Our method consists of three alternating updates. Each of them has a closed-form solution. In addition, the convergence behavior is studied. Experimental results are conducted on four real-world datasets. Compared with several ℓ_0-norm and ℓ_1-norm schemes, the proposed approach is superior in terms of returns and Sharpe ratios.

Appendix A
Proof of P1 and P2

A. Proof of P1:

For $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$, we consider

$$\begin{aligned}\Delta &= \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big)\\ &= \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big)\\ &\quad + \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big)\\ &\quad + \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big). \end{aligned} \tag{30}$$

From (14a), $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}, \boldsymbol{\gamma}^t)$ is minimized with respect to $\boldsymbol{z}$ by $(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)$. Thus,

$$\mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big) \le 0. \tag{31}$$

Since $\mathcal{L}(\boldsymbol{w}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)$ is strongly convex with respect to $\boldsymbol{w}$, the following inequality holds:

$$\begin{aligned}\mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) \ge\ & \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) + \tfrac{m}{2}\big\|\boldsymbol{w}^t - \boldsymbol{w}^{t+1}\big\|_2^2\\ &+ \nabla_{\boldsymbol{w}}\mathcal{L}|^{\mathrm{T}}_{(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)}(\boldsymbol{w}^t - \boldsymbol{w}^{t+1}). \end{aligned} \tag{32}$$

In addition, $\mathcal{L}(\boldsymbol{w}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)$ is a positive quadratic function of $\boldsymbol{w}$, then $\nabla_{\boldsymbol{w}}\mathcal{L}|_{(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)} = \boldsymbol{0}$. The relationship between $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)$ and $\mathcal{L}(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t)$ are given by

$$\mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big) \le -\tfrac{m}{2}\big\|\boldsymbol{w}^t - \boldsymbol{w}^{t+1}\big\|_2^2. \tag{33}$$

For the $\boldsymbol{\gamma}$-update, the difference of the function value is

$$\begin{aligned}&\mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big)\\ &\qquad = \big(\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1}\big)^{\mathrm{T}}\big(\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\big). \end{aligned} \tag{34}$$

Recalling (24), we have

$$\nabla\mathcal{F}(\boldsymbol{w}^{t+1}) + \boldsymbol{\gamma}^t + \rho\big(\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1}\big) = \boldsymbol{0}. \tag{35}$$

Based on (14c) and (35), we attain

$$\boldsymbol{\gamma}^{t+1} = -\nabla\mathcal{F}(\boldsymbol{w}^{t+1}) \text{ and } \boldsymbol{\gamma}^t = -\nabla\mathcal{F}(\boldsymbol{w}^t). \tag{36}$$

From (14c), we also conclude that

$$\boldsymbol{w}^{t+1} - \boldsymbol{z}^{t+1} = \tfrac{1}{\rho}(\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t). \tag{37}$$

Plugging (35)–(37) into (34), we have

$$\begin{aligned}&\mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^t\big)\\ &\quad = \tfrac{1}{\rho}\big\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|_2^2 = \tfrac{1}{\rho}\big\|\nabla\mathcal{F}\big(\boldsymbol{w}^{t+1}\big) - \nabla\mathcal{F}\big(\boldsymbol{w}^t\big)\|_2^2\\ &\quad = \tfrac{1}{\rho}\big\|2\boldsymbol{\Gamma}\boldsymbol{w}^{t+1} + C\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}}\boldsymbol{w}^{t+1} - 2\boldsymbol{\Gamma}\boldsymbol{w}^t - C\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}}\boldsymbol{w}^t\big\|_2^2\\ &\quad \le \tfrac{1}{\rho}\big\|2\boldsymbol{\Gamma} + C\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}}\big\|_2^2\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2 \le \tfrac{M^2}{\rho}\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2 \end{aligned} \tag{38}$$

where $M = \|2\boldsymbol{\Gamma} + C\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}}\|_2^2$ is the Lipschitz continuous constant of $\nabla\mathcal{F}(\boldsymbol{w})$. Plugging (31), (33), and (38) into (30), we see that

$$\mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big) - \mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big) \le \big(\tfrac{M^2}{\rho} - \tfrac{m}{2}\big)\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2. \tag{39}$$

Hence, the function value is monotonically nonincreasing with $\rho \ge (2M^2/m)$. Let $\eta = -((M^2/\rho) - (m/2))$. The proof is completed. ■

B. Proof of P2:

Since $\mathcal{F}(\boldsymbol{w})$ is convex and has a Lipschitz continuous gradient, we have

$$\mathcal{F}(\boldsymbol{z}^t) - \mathcal{F}(\boldsymbol{w}^t) \le \nabla\mathcal{F}(\boldsymbol{w}^t)^{\mathrm{T}}(\boldsymbol{z}^t - \boldsymbol{w}^t) + \tfrac{M}{2}\|\boldsymbol{z}^t - \boldsymbol{w}^t\|_2^2. \tag{40}$$

That is, $\mathcal{F}(\boldsymbol{z}^t) - \frac{M}{2}\|\boldsymbol{z}^t - \boldsymbol{w}^t\|_2^2 \le \mathcal{F}(\boldsymbol{w}^t) + \nabla\mathcal{F}(\boldsymbol{w}^t)^{\mathrm{T}}(\boldsymbol{z}^t - \boldsymbol{w}^t)$. As $\boldsymbol{\gamma}^t = -\nabla\mathcal{F}(\boldsymbol{w}^t)$, we obtain

$$\begin{aligned}\mathcal{F}(\boldsymbol{z}^t) - \tfrac{M}{2}\big\|\boldsymbol{z}^t - \boldsymbol{w}^t\big\|_2^2 &\le \mathcal{F}(\boldsymbol{w}^t) - (\boldsymbol{z}^t - \boldsymbol{w}^t)^{\mathrm{T}}\boldsymbol{\gamma}^t\\ &= \mathcal{F}(\boldsymbol{w}^t) + (\boldsymbol{w}^t - \boldsymbol{z}^t)^{\mathrm{T}}\boldsymbol{\gamma}^t. \end{aligned} \tag{41}$$

From (41), the function value $\mathcal{L}(\boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\gamma})$ at $(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t)$ is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t) &= \mathcal{F}(\boldsymbol{w}^t) + (\boldsymbol{w}^t - \boldsymbol{z}^t)^{\mathrm{T}}\boldsymbol{\gamma}^t + \tfrac{\rho}{2}\big||\boldsymbol{w}^t - \boldsymbol{z}^t\big|\big|_2^2 + \mathcal{I}(\boldsymbol{z}^t)\\ &\ge \mathcal{F}(\boldsymbol{z}^t) + \big(\tfrac{\rho}{2} - \tfrac{M}{2}\big)\big\|\boldsymbol{z}^t - \boldsymbol{w}^t\big\|_2^2 + \mathcal{I}(\boldsymbol{z}^t). \end{aligned} \tag{42}$$

Recall that $\mathcal{F}(\boldsymbol{z}) = \boldsymbol{z}^{\mathrm{T}}\boldsymbol{\Gamma}\boldsymbol{z} - \lambda\boldsymbol{u}^{\mathrm{T}}\boldsymbol{z} + (C/2)(\boldsymbol{z}^{\mathrm{T}}\boldsymbol{1} - 1)^2$. As $\boldsymbol{\Gamma}$ is positive definite, $\boldsymbol{\Gamma} + (C/2)\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}}$ is symmetric positive definite. Hence, $\mathcal{F}(\boldsymbol{z})$ can be expressed as $\mathcal{F}(\boldsymbol{z}) = \|\boldsymbol{b} - \mathbf{A}\boldsymbol{z}\|_2^2 + c$, where $\mathbf{A}$ is a full rank matrix with the property of $\boldsymbol{\Gamma} + (C/2)\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}}\mathbf{A}$, $\boldsymbol{b} = (1/2)(\mathbf{A}^{\mathrm{T}})^{-1}(\lambda\boldsymbol{u} + C\boldsymbol{1})$, and $c = (C/2) - \|\boldsymbol{b}\|_2^2$. Clearly, $\mathcal{F}(\boldsymbol{z})$ is lower bounded. In addition, $\mathcal{I}(\boldsymbol{z}^t)$ is lower bounded by 0. Hence, $\mathcal{L}(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t) > -\infty$ is lower bounded if $\rho \ge M$. Note that $\rho \ge (2M^2/m)$ must hold in **P1**. Hence, we should select a ρ value such that $\rho \ge \max\{M, (2M^2/m)\}$. The proof is completed. ■

Appendix B
Proof of Theorem 2

P1 indicates that

$$\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2 \le \tfrac{1}{\eta}\big(\mathcal{L}\big(\boldsymbol{w}^t, \boldsymbol{z}^t, \boldsymbol{\gamma}^t\big) - \mathcal{L}\big(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1}\big)\big). \tag{43}$$

Since $\mathcal{L}(\boldsymbol{w}^{t+1}, \boldsymbol{z}^{t+1}, \boldsymbol{\gamma}^{t+1})$ converges, we have $\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|_2^2 \to 0$, as $t \to \infty$.

Regarding $\boldsymbol{\gamma}^t$, from (38), $\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|_2^2 \le M^2\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|_2^2$. Therefore, we have $\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|_2^2 \to 0$, as $t \to \infty$.

Finally, for $\{\boldsymbol{z}^t\}$, we have

$$\begin{aligned}&\big\|\boldsymbol{z}^{t+1} - \boldsymbol{z}^t\big\|_2^2\\ &\quad = \Big\|\boldsymbol{w}^{t+1} - \tfrac{1}{\rho}\big(\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\big) - \boldsymbol{w}^t + \tfrac{1}{\rho}\big(\boldsymbol{\gamma}^t - \boldsymbol{\gamma}^{t-1}\big)\Big\|_2^2\\ &\quad \le \Big(\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2 + \tfrac{1}{\rho}\big\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\big\|_2 + \tfrac{1}{\rho}\big\|\boldsymbol{\gamma}^t - \boldsymbol{\gamma}^{t-1}\big\|_2\Big)^2\\ &\quad \le 3\Big(\big\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\big\|_2^2 + \tfrac{1}{\rho}\big\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\big\|_2^2 + \tfrac{1}{\rho}\big\|\boldsymbol{\gamma}^t - \boldsymbol{\gamma}^{t-1}\big\|_2^2\Big). \end{aligned} \tag{44}$$

In (44), the last inequality comes from the fact that $2ab \le a^2 + b^2$ for any real a and b. Since $\lim_{t\to\infty}\|\boldsymbol{\gamma}^{t+1} - \boldsymbol{\gamma}^t\|_2^2 = 0$ and $\lim_{t\to\infty}\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|_2^2 = 0$, and from (44), we can conclude that $\|\boldsymbol{z}^{t+1} - \boldsymbol{z}^t\|_2^2 \to 0$, as $t \to \infty$. The proof is completed. ■

REFERENCES

[1] Z.-R. Lai, D.-Q. Dai, C.-X. Ren, and K.-K. Huang, "Radial basis functions with adaptive input and composite trend representation for portfolio selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6214–6226, Dec. 2018.

[2] M.-F. Leung and J. Wang, "Minimax and biobjective portfolio selection based on collaborative neurodynamic optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 2825–2836, Jul. 2021.

[3] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, "Deep direct reinforcement learning for financial signal representation and trading," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 653–664, Mar. 2017.

[4] X. P. Li, Z.-L. Shi, C.-S. Leung, and H. C. So, "Sparse index tracking with K-sparsity or ε-deviation constraint via ℓ_0-norm minimization," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 16, 2022, doi: 10.1109/TNNLS.2022.3171819.

[5] H. Markowitz, "Portfolio selection," *J. Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[6] Y. Lee, M. J. Kim, J. H. Kim, J. R. Jang, and W. C. Kim, "Sparse and robust portfolio selection via semi-definite relaxation," *J. Oper. Res. Soc.*, vol. 71, no. 5, pp. 687–699, May 2020.

[7] W. Chen, H. Zhang, M. K. Mehlawat, and L. Jia, "Mean–variance portfolio optimization using machine learning-based stock price prediction," *Appl. Soft Comput.*, vol. 100, Mar. 2021, Art. no. 106943.

[8] P. Das, N. Johnson, and A. Banerjee, "Online lazy updates for portfolio selection with transaction costs," in *Proc. AAAI Artif. Intell.*, Washington, DC, USA, Jul. 2013, pp. 202–208.

[9] Z.-R. Lai, P.-Y. Yang, L. Fang, and X. Wu, "Short-term sparse portfolio optimization based on alternating direction method of multipliers," *J. Mach. Learn. Res.*, vol. 19, no. 1, pp. 2547–2574, Oct. 2018.

[10] F. D. Paiva, R. T. N. Cardoso, G. P. Hanaoka, and W. M. Duarte, "Decision-making for financial trading: A fusion approach of machine learning and portfolio selection," *Expert Syst. Appl.*, vol. 115, pp. 635–655, Jan. 2019.

[11] Z. Zhao and D. P. Palomar, "Large-scale regularized portfolio selection via convex optimization," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2019, pp. 1–5.

[12] S. Perrin and T. Roncalli, "Machine learning optimization algorithms & portfolio allocation," in *Machine Learning for Asset Management: New Developments and Financial Applications*, E. Jurczenko, Ed. Ed. Bridgewater, NJ, USA: Wiley, Jun. 2020, pp. 261–328.

[13] H. M. Markowitz, "Foundations of portfolio theory," *J. Finance*, vol. 46, no. 2, pp. 469–477, Jun. 1991.

[14] W. C. Kim, J. H. Kim, S. H. Ahn, and F. J. Fabozzi, "What do robust equity portfolio models really do?" *Ann. Oper. Res.*, vol. 205, no. 1, pp. 141–168, May 2013.

[15] S. Deshmukh and A. Dubey, "Improved covariance matrix estimation with an application in portfolio optimization," *IEEE Signal Process. Lett.*, vol. 27, pp. 985–989, 2020.

[16] W. C. Kim, J. H. Kim, and F. J. Fabozzi, "Deciphering robust portfolios," *J. Banking Finance*, vol. 45, pp. 1–8, Aug. 2014.

[17] P. J. Kremer, S. Lee, M. Bogdan, and S. Paterlini, "Sparse portfolio selection via the sorted ℓ_1-norm," *J. Bank Financ.*, vol. 110, Jan. 2020, Art. no. 105687.

[18] S. Corsaro and V. De Simone, "Adaptive ℓ_1-regularization for short-selling control in portfolio selection," *Comput. Optim. Appl.*, vol. 72, no. 2, pp. 457–478, Dec. 2019.

[19] B. Fastrich, S. Paterlini, and P. Winker, "Constructing optimal sparse portfolios using regularization methods," *Comput. Manage. Sci.*, vol. 12, no. 3, pp. 417–434, Jul. 2015.

[20] L. Wu, Y. Feng, and D. P. Palomar, "General sparse risk parity portfolio design via successive convex optimization," *Signal Process.*, vol. 170, May 2020, Art. no. 107433.

[21] D. Bertsimas and R. Shioda, "Algorithm for cardinality-constrained quadratic optimization," *Comput. Optim. Appl.*, vol. 43, no. 1, pp. 1–22, May 2009.

[22] Y. Tian, S. Fang, Z. Deng, and Q. Jin, "Cardinality constrained portfolio selection problem: A completely positive programming approach," *J. Ind. Manag. Optim.*, vol. 12, no. 3, p. 1041, Jul. 2016.

[23] J. Brodie, I. Daubechies, C. De Mol, D. Giannone, and I. Loris, "Sparse and stable Markowitz portfolios," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 30, pp. 12267–12272, Jul. 2009.

[24] J. Gao and D. Li, "Optimal cardinality constrained portfolio selection," *Oper. Res.*, vol. 61, no. 3, pp. 745–761, Jun. 2013.

[25] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.

[26] Q. Zhang, X. Hu, and B. Zhang, "Comparison of ℓ_1-norm SVR and sparse coding algorithms for linear regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 8, pp. 1828–1833, Aug. 2014.

[27] H. Wang, R. Feng, Z.-F. Han, and C.-S. Leung, "ADMM-based algorithm for training fault tolerant RBF networks and selecting centers," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3870–3878, Aug. 2018.

[28] T. Zhang *et al.*, "A systematic DNN weight pruning framework using alternating direction method of multipliers," in *Proc. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018, pp. 184–199.

[29] K. Benidis, Y. Feng, and D. P. Palomar, "Sparse portfolios for high-dimensional financial index tracking," *IEEE Trans. Signal Process.*, vol. 66, no. 1, pp. 155–170, Jan. 2018.

[30] M. Ç. Pınar, "On robust mean-variance portfolios," *Optimization*, vol. 65, no. 5, pp. 1039–1048, Jan. 2016.

[31] J. Shanken, "On the exclusion of assets from tests of the mean variance efficiency of the market portfolio: An extension," *J. Finance*, vol. 41, no. 2, pp. 331–337, Jun. 1986.

[32] L. T. Nielsen, "Portfolio selection in the mean-variance model: A note," *J. Finance*, vol. 42, no. 5, pp. 1371–1376, Dec. 1987.

[33] R. Jagannathan and T. Ma, "Risk reduction in large portfolios: Why imposing the wrong constraints helps," *J. Finance*, vol. 58, no. 4, pp. 1651–1683, Jul. 2003.

[34] D. X. Shaw, S. Liu, and L. Kopman, "Lagrangian relaxation procedure for cardinality-constrained portfolio optimization," *Optim. Methods Softw.*, vol. 23, no. 3, pp. 411–420, Jun. 2008.

[35] V. DeMiguel, L. Garlappi, F. J. Nogales, and R. Uppal, "A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms," *Manage. Sci.*, vol. 55, no. 5, pp. 798–812, May 2009.

[36] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.

[37] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *J. Sci. Comput.*, vol. 78, no. 1, pp. 29–63, Jan. 2019.

[38] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*. New York, NY, USA: Springer, 1984.

[39] R. Glowinski and P. L. Tallec, *Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics*. Philadelphia, PA, USA: SIAM, 1989.

[40] D. G. Luenberger, "Convergence rate of a penalty-function scheme," *J. Optim. Theory Appl.*, vol. 7, no. 1, pp. 39–51, Jan. 1971.

[41] B. Stephen and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[42] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1-problems in compressive sensing," *SIAM J. Sci. Comput.*, vol. 33, no. 1, pp. 250–278, Feb. 2011.

[43] S. Magnússon, P. C. Weeraddana, M. G. Rabbat, and C. Fischione, "On the convergence of alternating direction Lagrangian methods for nonconvex structured optimization problems," *IEEE Trans. Control Netw. Syst.*, vol. 3, no. 3, pp. 296–309, Sep. 2015.

[44] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, Aug. 2008.

[45] Z. Xu, S. De, M. Figueiredo, C. Studer, and T. Goldstein, "An empirical study of ADMM for nonconvex problems," 2016, *arXiv:1612.03349*.

[46] Z. Peng, J. Chen, and W. Zhu, "A proximal alternating direction method of multipliers for a minimization problem with nonconvex constraints," *J. Global Optim.*, vol. 62, no. 4, pp. 711–728, Mar. 2015.

[47] G. Guastaroba and M. G. Speranza, "Kernel search: An application to the index tracking problem," *Eur. J. Oper. Res.*, vol. 217, no. 1, pp. 54–68, Feb. 2012.

[48] (2019). *The MOSEK Optimization Toolbox for MATLAB Manual. Version 9.0.* [Online]. Available: http://docs.mosek.com/9.0/toolbox/index.html

[49] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx

[50] W. F. Sharpe, "Mutual fund performance," *J. Bus.*, vol. 39, no. 1, pp. 119–138, Jan. 1966.