Effective algorithms for optimal portfolio deleveraging problem with cross impact

Hezhi Luo | Yuanyuan Chen | Xianye Zhang | Duan Li | Huixian Wu

1Department of Mathematics, Zhejiang Sci-Tech University, Hangzhou, Zhejiang, China
2Department of Finance and Insurance, Nanjing University, Nanjing, Jiangsu, China
3School of Data Science, City University of Hong Kong, Hong Kong, China
4Department of Mathematics, Hangzhou Dianzi University, Hangzhou, Zhejiang, China

Correspondence
Yuanyuan Chen, Department of Finance and Insurance, Nanjing University, Nanjing, Jiangsu 210093, China.
Email: yychen@nju.edu.cn

Funding information
Natural Science Foundation of Zhejiang Province, Grant/Award Numbers: LY18A010011, LZ21A010003; Research Grants Council, University Grants Committee, Grant/Award Numbers: 14202017, 14213716; National Natural Science Foundation of China, Grant/Award Numbers: 12271485, 11871433, 72001105, 72171109

Abstract
We investigate the optimal portfolio deleveraging (OPD) problem with permanent and temporary price impacts, where the objective is to maximize equity while meeting a prescribed debt/equity requirement. We take the real situation with cross impact among different assets into consideration. The resulting problem is, however, a nonconvex quadratic program with a quadratic constraint and a box constraint, which is known to be NP-hard. In this paper, we first develop a successive convex optimization (SCO) approach for solving the OPD problem and show that the SCO algorithm converges to a KKT point of its transformed problem. Second, we propose an effective global algorithm for the OPD problem, which integrates the SCO method, simple convex relaxation, and a branch-and-bound framework, to identify a global optimal solution to the OPD problem within a prespecified \( \varepsilon \)-tolerance. We establish the global convergence of our algorithm and estimate its complexity. We also conduct numerical experiments to demonstrate the effectiveness of our proposed algorithms with both real data and randomly generated medium- and large-scale OPD instances.

Keywords
branch-and-bound, convex relaxation, cross-asset price impact, nonconvex quadratic program, optimal portfolio deleveraging, successive convex optimization
In the financial market, leverage has been widely used by investors to significantly increase their returns of an investment. Although high leverage may be beneficial in boom periods, it may cause serious cash flow problems in recessionary periods. During the recessionary periods, investors often need to unwind their portfolios to reduce leverage. As pointed out by D’Hulster (2009) and Cont and Wagalath (2016), a sudden deleveraging of large financial portfolios has been recognized as a destabilizing factor in recent financial crises. As a result, how to deleverage large portfolios is an important question not only to investors, but also to the whole financial market.

To deleverage a large portfolio is in fact to liquidate a large portfolio to meet the leverage requirement. When liquidating a large portfolio, the price impact of trading must be taken into account. Madhavan (2000) suggests that the price impact of trading can be decomposed into the temporary and permanent price impacts. While the temporary price impact measures the instantaneous price pressure that is resulted from trading, the permanent price impact measures the change in the equilibrium price that depends on the cumulative trading amount, and is independent of the trading rate. See Carlin et al. (2007) for a detailed discussion on the price impact.

Brown et al. (2010) first study the optimal deleveraging problem with both permanent and temporary price impacts of trading. With the assumption that both permanent and temporary price impact matrices are positive diagonal matrices and the convexity assumption that the temporary price impact is greater than one half of the permanent price impact for each asset in the portfolio, Brown et al. (2010) derive an analytical optimal trading strategy for the optimal deleveraging problem. Empirical studies (Holthausen et al., 1990; Sias et al., 2001; Schöneborn & Schied, 2009), however, show that permanent price impact may dominate in block transactions and hence the convexity assumption does not always hold.

Without this convexity assumption, Chen et al. (2014) study the optimal deleveraging problem considered in Brown et al. (2010), which reduces to a separable and nonconvex quadratic program with quadratic and box constraints. They propose an efficient Lagrangian method to solve this nonconvex quadratic program and present conditions under which their method provides an optimal solution. Furthermore, Chen et al. (2015) extend the Lagrangian method to the optimal deleveraging problem with nonlinear temporary price impact. However, Chen et al. (2014, 2015) still make the assumption that both permanent and temporary price impact matrices are diagonal in their models. In other words, no cross impact among different assets is considered in Brown et al. (2010) or Chen et al. (2014, 2015).

However, the existence of cross impact among different assets has been both empirically documented and theoretically justified. Fleming et al. (1998) show that portfolio rebalancing trades from privately informed investors can lead to cross impacts, even between assets fundamentally uncorrelated. Kyle and Xiong (2001) argue that correlated liquidity shocks due to financial constraints can lead to cross liquidity effects. Andrade et al. (2008) show that a trade imbalance in one stock affects not only its own price, but also the prices of other stocks with their equilibrium model and data from the Taiwan Stock Exchange. Pasquariello and Vega (2015) empirically suggest that the strategical trading activity of sophisticated speculators may induce the cross impact, and find that the impact matrix could be asymmetric with negative nondiagonal elements in their empirical analysis with data from NYSE and NASDAQ. Benzaquen et al. (2017) state that the cross-correlations among different stocks are, to a large extent, mediated by trades with the data set of 275 US stocks. Recently, Li et al. (2020) apply genetic algorithm to the optimal deleveraging problem with nondiagonal symmetric impact matrices, without further discussion on (i)
how the nondiagonal impact elements affect the trading strategies, and (ii) the convergence and complexity of their algorithm.

In this paper, we investigate how to solve the optimal portfolio deleveraging (OPD) problem with cross impact, by considering general permanent and temporary price impact matrices, which could be not only nondiagonal, but also asymmetric. The resulting problem is then a nonseparable and nonconvex program with a quadratic objective function and nonhomogeneous quadratic and box constraints, which is known to be NP-hard.

We first develop a first-order algorithm to find a Karush–Kuhn–Tucker (KKT) point for the OPD problem with cross impact. Various algorithms for finding KKT points of nonconvex quadratic programming (QP) are summarized in the survey papers (cf. Floudas and Visweswaran (1994); Gould and Toint (2002)). In particular, Palacios-Gomez et al. (1982) present a successive linear optimization approach to find a KKT point of QP. Ye (1998) propose an interior-point method to find a KKT point of linearly constrained quadratic programming (LCQP). Le Thi and Pham Dinh (1997) propose a D.C. (difference of convex functions) algorithm to find a KKT point of LCQP. Lanckriet and Sriperumbudur (2009) propose a convex–concave procedure (CCP) for a general D.C. program with D.C. constraints (denoted by DCP-DC) by linearizing the concave part of the D.C. structure, which requires an initial feasible point of the original problem. Later on, to remove the need for an initial feasible point in CCP, Pham Dinh and Le Thi (2014), Le Thi et al. (2014), Lipp and Boyd (2016), and Strekalovsky and Minarchenko (2018) propose the extended D.C. algorithms and the penalty CCP with updated penalty parameter for DCP-DC by using $l_\infty/l_1$ penalty methods and linearization techniques. In this work, we first reformulate the OPD problem as an equivalent D.C. program with a D.C. constraint by making use of the spectral decomposition and the simultaneous diagonalization (SD) method of two semi-definite matrices (cf. Newcomb (1961)). We then, with a constructed initial point, adopt the idea of Lanckriet and Sriperumbudur (2009) to propose a convex quadratic approximation for the transformed problem. Based on the solution to this convex approximation problem, we develop a successive convex optimization (SCO) approach for solving this transformed problem. We prove that the solution sequence generated by the SCO approach converges to a KKT point of the transformed problem.

Based on the local solution, we develop an effective algorithm to find a globally optimal solution for the OPD problem with cross impact. There have been quite a few works on the global algorithm to solve a quadratic program with a quadratic constraint (QCQP). They, unfortunately, cannot be directly applied to solve our problem in this paper. For example, based on the celebrated S-lemma (cf. Polik and Terlaky (2007)), such a QCQP problem can be reformulated as a semi-definite programming problem (cf. Ben-Tal and Teboulle (1995); Moré (1993)). Ben-Tal and den Hertog (2014) show that it can also be transformed into an equivalent second-order cone programming problem when the quadratic forms are simultaneously diagonalizable. By exploiting the block separability of the canonical form, Jiang et al. (2018) show that all generalized trust region subproblems with an optimal value bounded from below are second-order cone programming (SOCP) representable. However, these methods do not deal with the QCQP with box constraints. Nesterov (1998) and Ye (1997) study the semi-definite relaxation method for certain quadratic programs with quadratic and/or box constraints. However, the quadratic constraint in their problems does not contain a linear term. Audet et al. (2000) propose a branch-and-cut algorithm for generic nonconvex QCQPs. Linderoth (2005) propose a simplicial branch-and-bound (B&B) algorithm by reformulating the linear outer approximation of the nonconvex constraints. However, as reported in Audet et al. (2000) and Linderoth (2005), these two algorithms are targeted to solve small-scale instances. Burer and Vandenbussche (2008, 2009) and Chen and Burer (2012) propose the B&B algorithms for nonconvex quadratic programs with linear and/or box constraints by using
semi-definite relaxations of the first-order KKT conditions of the problem with finite KKT-branching. More recently, Luo et al. (2019) develop a new global algorithm for a nonconvex quadratic program with linear and convex quadratic constraints by combining several simple optimization techniques such as the alternative direction method, the B&B framework and the convex relaxation. This approach is further extended to the worst-case linear optimization with uncertainties that arises in estimating the systemic risk in financial systems (cf. Luo et al. (2021); Ding et al. (2021)). However, there is no nonconvex quadratic constraint in the problem these algorithms considered.

Inspired by Luo et al. (2019), we develop an effective global algorithm for the OPD problem with cross impact in this paper. For such a purpose, we first derive a simple convex relaxation for the transformed problem by replacing the concave quadratic term in the transformed problem with its convex envelope, and estimate the gap between the relaxation and the transformed problem. We then combine the SCO algorithm with the B&B framework and the convex relaxation, to develop an efficient global algorithm (called SCOBB) that can find a globally optimal solution to the transformed problem within a prespecified $\varepsilon$-tolerance. We establish the convergence of the SCOBB algorithm and estimate its complexity. Specifically, we show that the SCOBB algorithm has a worst-case complexity bound $\mathcal{O}(N \prod_{i=1}^{r} \frac{\sqrt{r+\rho_i} s(z_{iu} - z_{il})^2}{\Delta \varepsilon})$, where $N$ is the complexity to solve the relaxed subproblem (a convex QP).

The contribution of this paper is thus twofold. First, from the perspective of the optimal deleveraging problem of a portfolio, we relax the nonrealistic assumption of diagonal price impact matrices, which is usually made for mathematical and computational convenience, and derive analytical properties and numerical strategies on how to deleverage a portfolio with the consideration of cross impacts among different assets. With the nondiagonal cross impact matrices, we show that the asset, which is more liquid and less positively correlated with the other assets, would be prioritized for selling. We furthermore consider a two-period model with potential future liquidity shock, to find how the size and the probability of potential distress affect the first-period selling volume and selling priority. Second, from the perspective of the solution to a nonconvex quadratic program, we propose two efficient algorithms: the SCO algorithm and the SCOBB algorithm. We show that the SCO algorithm converges to a KKT point of the transformed problem, while the SCOBB algorithm can find its global $\varepsilon$-optimal solution in $\mathcal{O}(N \prod_{i=1}^{r} \frac{\sqrt{r+\rho_i} s(z_{iu} - z_{il})^2}{\Delta \varepsilon})$ time.

Furthermore, we demonstrate the application of our algorithms to the OPD problem with the historical data from NASDAQ in our numerical experiments. Although the estimation of cross impacts in practice could be quite inaccurate, we show the robustness of our algorithm. Within a certain estimation error range, we find in our numerical experiments that the strategy outputted by our algorithm would be better than that overlooking nondiagonal elements, in terms of the percentage of meeting the terminal leverage requirement. To illustrate the efficiency of our algorithms, we also randomly generate medium- and large-scale instances of the OPD problem to compare our two algorithms with the global optimization package BARON (cf. Sahinidis (1996)). Numerical experiments illustrate that our SCOBB algorithm provides the globally optimal deleverage strategies with acceptable computational time for medium- and large-scale instances. Although we cannot prove that our SCO algorithm provides a global optimal solution, it can solve large-scale instances within several minutes, and always provides the global optimal solution in our numerical experiments.

Our work is also related to the literature on the portfolio execution problem. Bertsimas et al. (1999) develop an approximation algorithm to solve the multi-asset portfolio problem. Almgren
and Chriss (2000) briefly discuss the multi-asset portfolio problem with a mean–variance objective in their appendix and obtain a solution for the special case without cross impact. Tsoukalas et al. (2019) analyze the optimal execution problem of a portfolio manager trading multiple assets with the cross impact. Cartea et al. (2019) investigate how to execute a basket of co-integrated assets with temporary and permanent price impacts. The main difference between these works and ours is the motivation of the problem. While portfolio execution problem is to liquidate the whole portfolio, our optimal deleveraging problem is to liquidate some of the assets inside a portfolio to meet the leverage requirement.

The remaining of this paper is organized as follows. In Section 2, we formulate the model of the OPD problem with the cross impact and derive analytical results on the optimal trading strategy. In Section 2.3, we propose an equivalent reformulation for the OPD problem based on SD method. In Section 3, we propose the SCO algorithm for the transformed problem and investigate its convergence properties. In Section 4, by integrating the SCO method with the SD-based convex relaxation, we develop a B&B algorithm to find the global optimal solution to the transformed problem. We establish the global convergence of the proposed algorithm and estimate its complexity. We present the two-period model and the corresponding results on preemptive deleveraging and trading priority in Section 5. We test the performance of the proposed algorithms with historical data from NASDAQ and randomly generated instances in Section 6. Finally, we present conclusions and future research directions in Section 7. The proofs of all the technical results are given in Appendix A.

2 | OPTIMAL PORTFOLIO DELEVERAGE PROBLEM

In this section, we first present the model formulation for the OPD problem, as in Brown et al. (2010) and Chen et al. (2014). The main difference between our model and theirs is that we consider nondiagonal price impact matrices. We then explore some properties of the optimal trading strategy for our model.

2.1 | Problem formulation

We consider a risk-neutral investor who trades a portfolio of \( m \) assets in continuous time over a finite horizon. The execution prices of the assets are modeled by

\[
p(t) = \bar{p}(t) + \Lambda y(t) = q + \Gamma x(t) + \Lambda y(t),
\]

(1)

which is a multidimensional version of the pricing equation used in Carlin et al. (2007). Here, \( p(t), \bar{p}(t), x(t), y(t) \in \mathbb{R}^m \) are the vectors of execution prices, fundamental values, holdings, and trading rates of the \( m \) assets at time \( t \), respectively. The matrix \( \Gamma = (\gamma_{ij})_{m \times m} \) documents permanent price impact coefficients of the cumulative trading amount \( (x(t) - x(0)) \) on the fundamental values and thus the execution prices, where the element \( \gamma_{ij} \) represents the permanent price impact parameter of asset \( i \) on asset \( j \). Meanwhile, the deleveraging trades also temporarily affect the execution prices by temporary price impact, which would not affect the fundamental values or the future prices. We denote the matrix \( \Lambda = (\lambda_{ij})_{m \times m} \) as the temporary price impact matrix of trading associated with the trading rate \( y(t) \), whose element \( \lambda_{ij} \) represents the temporary price impact parameter of asset \( i \) on asset \( j \). Prior to trading, both the initial holdings and initial prices are
positive: \( x(0_-) = x_0 > 0 \) and \( \bar{p}(0_-) = p(0_-) = p_0 > 0 \). It is obvious that the holdings and prices satisfy

\[
dx(t) = y(t) dt, \quad d\bar{p}(t) = \Gamma dx(t) = \Gamma y(t) dt. \tag{2}
\]

Similar as Brown et al. (2010) and Chen et al. (2014), we consider an investor who wishes to maximize the equity at the end of the time horizon \([0,1]\). The equity is the difference between the value of the portfolio and the liability. Prior to trading, the investor has an initial liability \( l_0 > 0 \) and an initial equity

\[
e(0_-) = e_0 = p_0^T x_0 - l_0 > 0.
\]

By deleveraging with a trading rate \( y(t), t \in [0,1] \), the trader would generate cash

\[
k = \int_0^1 -p(t)^T y(t) dt
\]

to reduce his/her liability. The investor would thus identify the optimal trading rate \( y(t) \) by solving the following problem:

\[
\max_{y(t)} e(1) = \bar{p}(1)^T x(1) - l_0 - \int_0^1 p(t)^T y(t) dt \tag{3}
\]

\[
s. t. \quad dx(t) = y(t) dt, \quad d\bar{p}(t) = \Gamma y(t) dt.
\]

\[
x(0) = x_0, \quad \bar{p}(0) = p_0.
\]

**Proposition 2.1.**

(i) If the permanent price impact matrix \( \Gamma \) is symmetric and the matrix \( (\Lambda + \Lambda^T) \) has a full rank, then the optimal trading rate to maximize the terminal equity is constant.

(ii) If the matrix \( (\Lambda + \Lambda^T) \) has a full rank, then the optimal trading rate to maximize the terminal equity takes the form of

\[
y(t) = e^{tC} \cdot y^*(0),
\]

where the time \( t \in [0,1] \), the matrix \( C \) is defined by \( C = (\Lambda + \Lambda^T)^{-1}(\Gamma^T - \Gamma) \). Together with the boundary condition \(-x_0 \leq y(0) \leq 0\), \( y^*(0) \) is then the optimal solution to the following problem:

\[
\max_{y(0) \in \mathbb{R}^m} e_0 + x_0^T \bar{p} y(0) - y(0)^T (\bar{\Lambda} - B) y(0) \tag{4}
\]

\[
s. t. \quad -x_0 \leq y(0) \leq 0,
\]

where

\[
\Gamma = \Gamma(e^C - I) C^{-1}, \quad \bar{\Lambda} = \left( \int_0^1 (e^{tC})^T \Lambda e^{tC} dt \right), \quad B = (C^{-1})^T \left( \int_0^1 (e^{tC} - I)^T \Gamma e^{tC} dt \right).
\]
Remark 2.2. The boundary condition \(-x_0 \leq y(0) \leq 0\) in part (ii) corresponds to the fact that we restrict the trader from short selling or increasing positions at the beginning of this deleveraging problem. This setting is common in deleveraging models (see e.g., Brown et al. (2010); Chen et al. (2014)).

It is worth mentioning that from the proof of Proposition 2.1, we see that in general case, the optimal trading rate would take the form of

\[
y(t) = e^{t(\Lambda + \Lambda^T) + (\Gamma^T - \Gamma)} \cdot y(0) + \int_0^t e^{(t-s)(\Lambda + \Lambda^T) + (\Gamma^T - \Gamma)} ds \cdot \zeta,
\]

where \((\cdot)^+\) is the Moore–Penrose inverse, and \(\zeta\) is a vector from the null space of the matrix \((\Lambda + \Lambda^T)\). Proposition 2.1 provides a sufficient, but not necessary condition, under which the optimal trading rate is constant. Based on such a proposition and to simplify the problem, we assume that the investor would admit a constant trading rate during this optimal deleveraging problem.

Assumption 2.3. The trader admits a constant trading rate \(y(t) = y\) in the deleveraging problem.

You may notice that by admitting such an assumption, we choose a class of suboptimal solution to the original problem (3) and thus may induce an optimality gap in the general case. However, Proposition 2.1 also suggests that the constant rate could be optimal under certain conditions and a linear approximation to the optimal solution under a more general circumstance. In fact, as shown in Example 6.1 with real data, we find that the relative gap between the objective value of our suboptimal solution and the global optimal value is around 0.15%. This fact also suggests that the loss of optimality induced by imposing Assumption 2.3 is acceptable. Under such an assumption, the optimal deleveraging problem is converted to an optimization problem to identify the optimal trading rate \(y\) to maximize the terminal equity. Under Assumption 2.3, we observe from Equation (2) that at time \(t\), the holdings satisfy

\[
x(t) - x_0 = \int_0^t y(s)ds = yt, \quad t \in [0,1].
\]

After trading, the asset holdings are \(x(1) = x_0 + y\), and the asset prices become \(\tilde{p}(1) = q + \Gamma x(1) = p_0 + \Gamma y\). The investor would generate the amount of cash as

\[
\kappa(y) = \int_0^1 -p(t)^T y(t)dt = \int_0^1 -(p_0 + \Gamma y t + \Lambda y)^T ydt = -p_0^T y - y^T \left( \Lambda + \frac{1}{2} \Gamma \right) y.
\]

The liability after trading can be obtained by subtracting the cash amount expressed above from \(l_0\), and is a function of \(y\),

\[
l_1(y) = l_0 - \kappa(y) = l_0 + p_0^T y + y^T \left( \Lambda + \frac{1}{2} \Gamma \right) y.
\]

Thereupon, we write down the terminal equity after trading as

\[
e_1(y) = \tilde{p}(1)^T x(1) - l_1(y) = -y^T \left( \Lambda - \frac{1}{2} \Gamma \right) y + x_0^T \Gamma y + p_0^T x_0 - l_0.
\]
Because of the margin requirements imposed by lenders or the regulatory constraints, as pointed out by Brown et al. (2010), the trader has to guarantee the ratio of debt to equity to be no higher than a predetermined bound $\rho_1$. In the beginning of time horizon, the ratio of debt to equity exceeds $\rho_1$. In other words, we assume that the leverage requirement is not satisfied before trading, that is, $\rho_1 e_0 < l_0$. The trader has to guarantee the leverage ratio after trading not exceeding $\rho_1$, that is,

$$\frac{l_1(y)}{e_1(y)} \leq \rho_1 \iff \rho_1 e_1(y) - l_1(y) \geq 0. \quad (7)$$

Furthermore, with the same spirit as Brown et al. (2010), we are interested in modeling liquidity shocks that force the trader to quickly sell asset. As a result, the trader is restricted from increasing positions and from short selling during this fire sale. This corresponds to the sale constraint $-x_0 \leq y \leq 0$. The problem is, thus, to find an optimal trading strategy that maximizes the equity $e_1$ subject to both the leverage constraint and the sale constraint. With the time horizon being [0,1], we then formulate the OPD problem as the following QP problem:

$$\max_{y \in \mathbb{R}^m} e_1(y) = -y^T \left( \Lambda - \frac{1}{2} \Gamma \right) y + x_0^T \Gamma y + e_0$$

s. t. $- y^T \left( \Lambda + \frac{1}{2} \Gamma + \rho_1 \left( \Lambda - \frac{1}{2} \Gamma \right) \right) y - (p_0 - \rho_1 \Gamma^T x_0)^T y - l_0 + \rho_1 e_0 \geq 0,$

$$-x_0 \leq y \leq 0. \quad (8)$$

The first constraint corresponds to the leverage requirement (7) with the liability and equity derived in Equations (5) and (6).

It is worth mentioning that because the prices of any two assets may interact with each other in real market, as we discussed in Section 1, the two matrices $\Gamma$ and $\Lambda$ could be nondiagonal and asymmetric. To guarantee the meaningfulness of our problem, we assume that the leverage requirement can be always satisfied by liquidating all the assets:

**Assumption 2.4.** $l_1(-x_0) < 0$.

Note that the cumulative trading amount $y = -x_0$ implies that the investor liquidates all the assets. As a result, this assumption states that the investor should have no liability if he liquidates all his assets. With this assumption, our model focuses on the traders who do not need to liquidate all his assets to meet the liability obligation. If an investor cannot meet his liability obligation by liquidating all his assets, he would be bankrupted, and should just liquidate all his assets, instead of considering any optimization problem. Under this assumption, we show in the following proposition that if the assets are sold out, then the investor’s equity is positive, and that the leverage ratio is less than the predetermined level $\rho_1$.

**Proposition 2.5.** $e_1(-x_0) > 0$ and $\rho_1 e_1(-x_0) - l_1(-x_0) > 0$.

This proposition also implies that under Assumption 2.4, problem (8) has a nonempty feasible strategy set, because the strategy $y = -x_0$ is feasible.

Note that we do not have any assumption on the price impact matrices $\Gamma$ and $\Lambda$. As a result, the problems with positive diagonal matrices in Brown et al. (2010) and Chen et al. (2014) are
actually special cases of problem (8). To deal with the difficulties introduced by the cross terms in the impact matrices, we present an equivalent reformulation for problem (8) based on the spectral decomposition and SD method in Section 2.3, and develop algorithms in the subsequent sections. Before that, we first discuss the properties of our optimal strategy.

### 2.2 Properties of optimal trading strategy

We first show that the leverage constraint of problem (8) at the optimal solution is active under certain condition.

**Proposition 2.6.** If the price impact matrices $\Gamma$ and $\Lambda$ are non-negative and the permanent price impact matrix $\Gamma$ is symmetric, then the leverage constraint of problem (8) is active at its optimal solution.

Proposition 2.6 states that the optimal trading strategy precisely achieves the maximal allowed leverage ratio when the price impact matrices are non-negative and the permanent price impact matrix is symmetric. Under such a condition, it is suboptimal to further reduce the leverage ratio because of the trading cost caused by market impact. Intuitively, during a fire sale, traders may hold the view that they should always, instead of under certain condition, liquidate the assets to precisely satisfy the leverage requirement, instead of selling more. However, if the price impact of one asset on another asset is negative, which is to say, there is hedging property inside the portfolio, selling one asset can improve the value of the other asset. The trader, thus, might be willing to sell more to improve his final equity. Note that, as we do not impose any assumption on the price impact matrices, the trader is possibly willing to sell more in our model if some price impact parameters are negative.

Proposition 2.6 also implies that the positive diagonal impact matrices discussed in Chen et al. (2014) and Brown et al. (2010) can guarantee the activation of the leverage constraint. In addition, note that we only provide a sufficient, but not necessary condition for the activation of the leverage constraint. A symmetric permanent price impact matrix is, thus, not necessary for the activation of the leverage constraint. We show in Example 2.9 later that the leverage constraint could be active for problem (8) with asymmetric price impact matrices with several negative elements. This can be also verified by Example 6.1 in our numerical experiments with real data in Section 6.1.

Next, we present a result on the asset trade priority, which are also discussed in Chen et al. (2014) and Brown et al. (2010). We discuss this property without the assumption of diagonal price impact matrices. Define the average cross-stock impact matrices as

$$
\hat{\Gamma} = (\hat{\gamma}_{lk})_{m\times m} := \frac{1}{2}(\Gamma + \Gamma^T), \quad \hat{\Lambda} = (\hat{\lambda}_{lk})_{m\times m} := \frac{1}{2}(\Lambda + \Lambda^T). \tag{9}
$$

The average cross-stock impact matrices describe the long-term and short-term co-movement between different assets’ prices and trades. In particular, if $\hat{\gamma}_{lk} > \hat{\gamma}_{jk}$, then we can state that the mutual impact between assets $l$ and $k$ is higher than that between assets $j$ and $k$. As pointed out by Andrade et al. (2008), the mutual price impact is higher among assets with more positively correlated cash flows than among stocks with less positively correlated cash flows. We can, thus, understand these average cross-stock impacts as the indicator of the correlations for the cash flow of the short-term or long-term assets.
Proposition 2.7. Suppose assets $i$ and $j$ have the same initial price and holding: $p_{0,i} = p_{0,j}$ and $x_{0,i} = x_{0,j}$. If the following four conditions hold, then the $i$th asset is prioritized for selling, that is, $y^*_i \leq y^*_j$.

(i) $\hat{\lambda}_{ii} \leq \hat{\lambda}_{ij} = \hat{\lambda}_{ji}$ and $\hat{\gamma}_{ii} \leq \hat{\gamma}_{ij} = \hat{\gamma}_{ji}$.

(ii) For $\forall k \neq i, j$, $\hat{\lambda}_{ik} \leq \hat{\lambda}_{jk}$ and $\hat{\gamma}_{ik} \leq \hat{\gamma}_{jk}$.

(iii) $\gamma_{ii} - \gamma_{ij} \leq \gamma_{jj} - \gamma_{jj}$.

(iv) For $\forall k \neq i, j$, $\gamma_{ki} - \gamma_{ik} \leq \gamma_{kj} - \gamma_{jk}$.

With the consideration of nondiagonal elements, Proposition 2.7 reveals how the cross-stock impacts affect the trading priority in the portfolio deleveraging problem. Put simply, Proposition 2.7 states that, if the $i$th asset is regarded as more liquid and less positively correlated with the other assets, than the $j$th asset, it is prioritized for selling. We next discuss the implications of the four conditions in Proposition 2.7.

As we discussed in the paragraph right before Proposition 2.7, the condition (ii) in Proposition 2.7 implies that the $i$th asset is less positively correlated with the other assets. For example, if all the other assets, except the $i$th asset, belong to an identical industry, the $i$th asset is highly likely to be less positively correlated with the other assets, than the $j$th asset. Selling such an asset would be less likely to affect the other assets’ price, compared with selling an asset from the specific industry. Meanwhile, the price of a more liquid asset is supposed to be less significantly affected by the liquidity shocks. The condition (i) in Proposition 2.7 thus implies that the $i$th asset is more liquid than the $j$th asset. The difference $\gamma_{jj} - \gamma_{ji}$ depicts to what extent the price of asset $j$ is more significantly affected by the trades of asset $j$ than the price of the $i$th asset. Similarly, the difference $\gamma_{kj} - \gamma_{jk}$ depicts to what extent the price impact of the trades of asset $k$ on asset $j$ is more significant than the price impact of the $j$th asset’s trades on the asset $k$. As a result, the conditions (i)–(iv) in Proposition 2.7 imply that the $i$th asset is less positively correlated with the other assets and more liquid than the $j$th asset.

To illustrate the sufficiency and applications of the conditions in Propositions 2.6 and 2.7, we construct the following two examples.

Example 2.8. Consider an instance of problem (8) with $m = 3$ and the following parameters:

$$
\Lambda = \begin{pmatrix}
0.0052 & 0.0083 & 0.0087 \\
0.0037 & 0.0085 & 0.0059 \\
0.0037 & 0.0094 & 0.0093
\end{pmatrix}, \ \Gamma = \begin{pmatrix}
0.0041 & 0.0054 & 0.0074 \\
0.0054 & 0.0067 & 0.0079 \\
0.0074 & 0.0079 & 0.0048
\end{pmatrix}, \ x_0 = (1, 1, 1)^T, \ p_0 = (7, 7, 8)^T.
$$

Suppose that the initial liability equity ratio is $l_0/e_0 = 25$ and the required leverage ratio is $\rho_1 = 18$. One can easily calculate that the initial liability is $l_0 = 21.1538$, and the initial equity is $e_0 = 0.8462$, thus satisfying Assumption 2.4.

We can apply Propositions 2.6 and 2.7 to conclude that the investor should sell the assets to exactly meet the leverage ratio $\rho_1$, and prioritize selling asset 1, rather than asset 2, during the deleveraging trading. Note that all the elements in the impact matrices are positive in this example. Selling any asset in the portfolio would decrease the prices of all the assets, and thus decrease the investor’s final equity. Therefore, the investor would sell assets to precisely meet the liquidity requirement. In addition, compared with the second asset, the first asset is more liquid with lower self price impacts, and less correlated with the third asset with lower averaged cross-
impacts. Therefore, the investor would sell more asset 1 than asset 2 to reduce unfavorable price movements, and thus the decrease of his final equity due to his deleveraging trading.

In fact, by solving this example by Algorithm 2 in Section 4, one can obtain the global optimal solution \( y^* = (-0.7842, -0.0001, -0.0943)^T \) with the optimal value \( e_1(y^*) = 0.8287 \). It is easy to verify that the value of the leverage constraint at the optimal solution equals 0.000005, and that \( y^*_1 < y^*_2 \). Thus, the leverage constraint of this example is indeed active at its optimal solution, and the first asset is prioritized for selling rather than asset 2.

Example 2.9. Consider an instance of problem (8) with \( m = 3 \) and the following parameters:

\[
\Lambda = \begin{pmatrix}
0.0012 & -0.0040 & 0.0046 \\
0.0130 & 0.0079 & 0.0083 \\
-0.0022 & -0.0023 & 0.0081 \\
\end{pmatrix},
\Gamma = \begin{pmatrix}
0.0060 & 0.0108 & -0.0062 \\
0.0036 & 0.0084 & 0.0014 \\
-0.0096 & 0.0058 & 0.0090 \\
\end{pmatrix},
x_0 = (1,1,1)^T, p_0 = (7,7,4)^T.
\]

Suppose that the initial liability equity ratio is \( l_0/e_0 = 25 \) and the required leverage ratio is \( \rho_1 = 12 \). One can easily calculate that the initial liability is \( l_0 = 17.31 \) and the initial equity is \( e_0 = 0.69 \), thus satisfying Assumption 2.4.

Note that \( \Gamma \) and \( \Lambda \) are asymmetric matrices with negative off-diagonal elements. The trader could increase significantly the price of asset 1 by selling the third asset, due to the fact of \( \lambda_{31} < 0 \) and \( \gamma_{31} < 0 \). The condition in Proposition 2.6 does not hold. However, selling asset 3 could also decrease the price of asset 3 itself and that of asset 2. The trader might still choose not to sell more than the leverage requirement. In fact, by using Algorithm 2 to solve this problem, we obtain its global optimal solution \( y^* = (-1, -0.2241, -0.1548)^T \) with the optimal value \( e_1(y^*) = 0.6855 \), and the value of the leverage constraint at the optimal solution equals \( 10^{-6} \). That is to say, the leverage ratio is active at the optimal solution. In addition, it can be also seen from the optimal solution \( y^* \) that \( y^*_1 = -1 < -0.2241 = y^*_2 \). In fact, one can easily check that the four conditions in Proposition 2.7 are satisfied for assets 1 and 2. The first asset is more liquid and less positively correlated with the third asset than asset 2. As a result, the first asset is prioritized for selling than the second asset.

2.3 Reformulation via simultaneous diagonalizability

We observe that the two matrices \( \Lambda - \frac{1}{2} \Gamma \) and \( \Lambda + \frac{1}{2} \Gamma \), respectively, in the objective function and the constraint of problem (8) may be indefinite, and even asymmetric. This makes problem (8) difficult to be solved. To remedy this, in this subsection, we present an equivalent reformulation for problem (8) by making use of the spectral decomposition and the simultaneous diagonalizability of two semi-definite matrices. The idea of simultaneous diagonalizability is also employed to solve the QCQP problems in Ben-Tal and den Hertog (2014) and Jiang et al. (2018).

By means of the symmetric matrices \( \tilde{\Gamma} \) and \( \tilde{\Lambda} \) defined in Equation (9), we first reformulate problem (8) into the following form:

\[
\begin{align*}
\min_{y \in \mathbb{R}^m} & \quad f(y) := y^T \left( \tilde{\Lambda} - \frac{1}{2} \tilde{\Gamma} \right) y - x_0^T \Gamma y \\
\text{s. t.} & \quad g(y) := y^T \left( \tilde{\Lambda} + \frac{1}{2} \tilde{\Gamma} + \rho_1 \left( \tilde{\Lambda} - \frac{1}{2} \tilde{\Gamma} \right) \right) y + (p_0 - \rho_1 \Gamma^T x_0)^T y + l_0 - \rho_1 e_0 \leq 0, \\
& \quad -x_0 \leq y \leq 0,
\end{align*}
\]

(10)
where \( \hat{\Gamma} + \frac{1}{2} \hat{\Lambda} \) and \( \hat{\Gamma} - \frac{1}{2} \hat{\Lambda} \) are symmetric but not necessarily positive definite or semidefinite. Clearly, problems (8) and (10) are equivalent in the sense that they have the same optimal solution. When \( \hat{\Gamma} + \frac{1}{2} \hat{\Lambda} \) and \( \hat{\Gamma} - \frac{1}{2} \hat{\Lambda} \) are positive semidefinite, problem (10) is a convex QP problem which is polynomially solvable in its second-order cone program reformulation. Generally, problem (10) is still a nonconvex quadratic program with one quadratic and box constraints, which is NP-hard.

We next reformulate problem (10) as an equivalent D.C. program and discuss how to recover a global solution to problem (10) from the solution of the reformulated problem. Denote by \( \nu_i \), \( i = 1, \ldots, m \), the eigenvalues of the matrix \( \hat{\Lambda}^{-\frac{1}{2}} \hat{\Gamma} \), where \( \nu_i < 0 \) for \( i = 1, \ldots, s \) and \( \nu_i \geq 0 \) for \( i = s + 1, \ldots, m \), and by \( \eta_i \), \( i = 1, \ldots, m \), the corresponding orthogonal unit eigenvectors. We also denote the eigenvalues of the matrix \( \hat{\Lambda}^{-\frac{1}{2}} \hat{\Gamma} \) by \( \mu_i \), \( i = 1, \ldots, m \), with \( \mu_i < 0 \) for \( i = 1, \ldots, q \), and \( \mu_i \geq 0 \) for \( i = q + 1, \ldots, m \), and the corresponding orthogonal unit eigenvectors by \( \zeta_i \), \( i = 1, \ldots, m \). By the spectral decomposition, we can obtain

\[
\begin{aligned}
\hat{\Lambda}^{-\frac{1}{2}} \hat{\Gamma} &= B^+ - B^-, \quad B^+ := \sum_{i=s+1}^{m} \nu_i \eta_i \eta_i^T, \\
\hat{\Lambda}^{-\frac{1}{2}} \hat{\Gamma} &= A^+ - A^-, \quad A^+ := \sum_{i=q+1}^{m} \mu_i \zeta_i \zeta_i^T.
\end{aligned}
\]

(11)

Recall a well-known result regarding the simultaneous diagonalizability of two semi-definite matrices from Newcomb (1961).

**Lemma 2.10.** Let \( A \) and \( B \) be two \( n \times n \) real symmetric positive semi-definite matrices. Then \( A \) and \( B \) are simultaneously diagonalizable, that is, there exists a nonsingular matrix \( D \) such that both \( D^T AD \) and \( D^T BD \) are diagonal.

According to Lemma 2.10, because \( B^- \) and \( A^- \) are symmetric positive semi-definite matrices, \( B^- \) and \( A^- \) are simultaneously diagonalizable. In particular, we can identify a nonsingular matrix \( D \) to simultaneously diagonalize these two matrices as

\[
D^T B^- D = \text{diag}(\delta_1, \ldots, \delta_s, 0, \ldots, 0), \quad D^T A^- D = \text{diag}(\vartheta_1, \ldots, \vartheta_r, 0, \ldots, 0),
\]

(12)

where \( r = \text{rank}(B^- + A^-) \) with \( \max\{s, q\} \leq r \leq s + q, 0 \leq \delta_i \leq 1 \) for \( i = 1, \ldots, s, \vartheta_i = 1 - \delta_i \) for \( i = 1, \ldots, s \) and \( \vartheta_i = 1 \) for \( i = s + 1, \ldots, r \), and there are \( q \) nonzero numbers in \( \vartheta_1, \ldots, \vartheta_r \) (see the proof of Lemma 2.10 in Appendix A). Using the transformation \( y = Dz \), we then obtain the following proposition.

**Proposition 2.11.** Problem (10) has the same optimal value as the following D.C. program with a D.C. constraint

\[
\begin{aligned}
\min_{z \in \mathbb{R}^m} \hat{f}(z) &:= z^T (D^T B^+ D) z - x_0^T \Gamma D z - \sum_{i=1}^{s} \delta_i z_i^2 \\
\text{s. t.} \quad \hat{g}(z) &:= \psi(z) - \sum_{i=1}^{r} \vartheta_i z_i^2 - \rho_1 \sum_{i=1}^{s} \delta_i z_i^2 \leq 0,
\end{aligned}
\]

(13)

\( z \in \mathcal{Z} := \{ z \in \mathbb{R}^m \mid -x_0 \leq Dz \leq 0 \} \).
where $\psi(z)$ is a quadratic convex function given by

$$
\psi(z) := z^T D^T (A^+ + \rho_1 B^+) Dz + (p_0 - \rho_1 \Gamma^T x_0)^T Dz + l_0 - \rho_1 e_0.
$$

(14)

Furthermore, if $z^*$ is a global optimal solution to problem (13), then $y^* = Dz^*$ is a global optimal solution to problem (10).

It is worth mentioning that problem (13) is still nonconvex and its global optimal solution is hard to obtain. However, the objective function and the constraint of problem (13) have simple separable concave quadratic terms $-z_i^2$. Based on this special structure, we will, in the subsequent sections, develop an effective algorithm to find a global optimal solution to problem (13).

3 | THE SCO METHOD AND ITS CONVERGENCE

In this section, with a strict feasible solution shown in Propositions 2.5 and 2.11, we propose an SCO algorithm for problem (13). Our algorithm falls into the framework of the CCP, without any penalty parameter updates. As mentioned in the introduction part, the CCP algorithm is proposed by Lanckriet and Sriperumbudur (2009) for a general D.C. program with D.C. constraints that includes problem (13) as a special case. They analyze the global convergence of CCP (Theorem 10) by relying on Zangwill’s *global convergence* theory of iterative algorithms (Theorem 2). However, the convergence proof is incomplete without the proof of the closedness of a point-to-set map associated with the CCP algorithm (denoted by $B_{ccp}$). Our Lemma 3.4 below proves the closedness of $B_{ccp}$. Thereupon, we prove the convergence of our SCO algorithm to a KKT point of problem (13).

3.1 | Quadratic convex approximation

We first present a quadratic convex approximation for problem (13) by linearizing the concave quadratic terms in the objective function and the constraint. This idea of linearizing the concave terms in the D.C. program has been used in the literature (cf. Lanckriet and Sriperumbudur (2009); Hong et al. (2011); Pham Dinh and Le Thi (2014); Luo et al. (2019, 2021)).

Let $z_l, z_u \in \mathbb{R}^m$ denote, respectively, the lower and upper bounds of $z = D^{-1} y$ for $y \in [-x_0, 0]$ given by

$$
z_l = -\sum_{j:D_{ij} > 0} \hat{D}_{ij} x_{0,j}, \quad z_u = -\sum_{j:D_{ij} < 0} \hat{D}_{ij} x_{0,j}, \quad i = 1, \ldots, m,
$$

(15)

where $\hat{D}_{ij}$ denotes the element of the matrix $D^{-1}$. For an arbitrary given $\xi_i \in [z_l^i, z_u^i]$, one has

$$
-z_i^2 \leq -\xi_i^2 - 2\xi_i (z_i - \xi_i) = -2\xi_i z_i + \xi_i^2, \quad \forall z_i \in [z_l^i, z_u^i].
$$

(16)

Using the linear term $(-2\xi_i z_i + \xi_i^2)$ to approximate the concave quadratic terms $-z_i^2$ in the objective function and the constraint of problem (13), we have the following quadratic convex
approximation:

\[
\min_{z \in \mathbb{R}^m} \tilde{f}_\xi(z) := z^T (D^T B^+ D) z - x_0^T \Gamma D z + \sum_{i=1}^s \delta_i (-2 \xi_i z_i + \xi_i^2) \\
\text{s.t.} \quad \tilde{g}_\xi(z) := \psi(z) + \sum_{i=1}^r \theta_i (-2 \xi_i z_i + \xi_i^2) + \sum_{i=1}^s \delta_i (-2 \xi_i z_i + \xi_i^2) \leq 0,
\]

\[z \in \mathcal{Z},\]

where \(\xi = (\xi_1, \ldots, \xi_r)^T\) with \(\xi_i \in [z^i_l, z^i_u], i = 1, \ldots, r\), and \(\psi(z)\) is a convex function defined in Equation (14). Note that inequality (16) implies that \(\tilde{g}_\xi(z) \geq g(z)\), thus the optimal solution to problem (17) is also feasible to problem (13). Hence, the objective function value \(\tilde{f}(z)\) at the optimal solution to problem (17) provides an upper bound for problem (13).

We denote the feasible sets of problems (13) and (17) and the interior of problem (17) by

\[\mathcal{F} = \{z \in \mathcal{Z} | \tilde{g}(z) \leq 0\}, \quad \mathcal{F}_\xi = \{z \in \mathcal{Z} | \tilde{g}_\xi(z) \leq 0\}, \quad \text{int} \mathcal{F}_\xi = \{z \in \mathcal{Z} | \tilde{g}_\xi(z) < 0\}\]

We next present a simple property of the sets \(\mathcal{F}_\xi\) and \(\text{int} \mathcal{F}_\xi\).

**Lemma 3.1.** Let \(\bar{z} \in \mathcal{F}_\xi\) and \(\bar{\xi}_i = \bar{z}_i\) for \(i = 1, \ldots, r\). We have the following.

(i) \(\mathcal{F}_\xi\) is a nonempty closed convex set in \(\mathcal{F}\).

(ii) \(\text{int} \mathcal{F}_\xi \neq \emptyset\).

From Lemma 3.1, we see that problem (17) is feasible and well-defined and that the Slater constraint qualification holds for problem (17). The following proposition follows immediately.

**Proposition 3.2.** Let \(\bar{z} \in \mathcal{F}_\xi\) and \(\bar{\xi}_i = \bar{z}_i\) for \(i = 1, \ldots, r\). If \(\bar{z}\) is an optimal solution to problem (17) with \(\xi = \bar{\xi}\), then \(\bar{z}\) is a KKT point of problem (13).

### 3.2 The SCO algorithm

Next, we describe the SCO algorithm for problem (13) based on the convex approximation (17).

It should be pointed out that the SCO algorithm needs an initial solution \(z^0 \in \mathcal{F}\). By Proposition 2.5, \(-x_0\) is a feasible solution of problem (8). Note that problems (8) and (10) have the same feasible set, and that \(y = Dz\) with \(D\) being nonsingular. Thus, \(z^0 = D^{-1}(-x_0)\) is also a feasible solution to problem (13).

We remark that at the \(k\)th iteration of the SCO algorithm, from Step 1, we obtain \(z^{k+1}\) by solving problem (17) with \(\xi_i = z^k_i\) \((i = 1, \ldots, r)\), that is,

\[z^{k+1} \in \arg\min_{z} z^T (D^T B^+ D) z - x_0^T \Gamma D z - \sum_{i=1}^s \delta_i \psi_i (z_i; z^k_i)\]
Algorithm 1 [SCO algorithm (SCO($z^0, \varepsilon$))]

Input: Initial feasible point $z^0 \in \hat{\mathcal{F}}$ and stopping criterion $\varepsilon > 0$.

Step 0 Set $\xi^0 = z^0$ for $i = 1, \ldots, r$ and $\xi^0 = (\xi_1^0, \ldots, \xi_r^0)^T$. Set $k = 0$.

Step 1 Solve problem (17) with $\xi = \xi^k$ to get the optimal solution $z^{k+1}$. Set $\xi^{k+1} = (\xi_1^{k+1}, \ldots, \xi_r^{k+1})^T$ for $i = 1, \ldots, r$.

Step 2 If $\|\xi^{k+1} - \xi^k\| > \varepsilon$, then set $k = k + 1$ and go back to step 1; Otherwise, stop and output $z^{k+1}$ as the final output.

\[
\begin{aligned}
\text{s. t. } & \psi(z) - \sum_{i=1}^r \theta_i \hat{v}_i (z_i; z^k) - \rho_1 \sum_{i=1}^s \delta_i \hat{v}_i (z_i; z^k) \leq 0, \\
& z \in \mathcal{Z},
\end{aligned}
\]

where $\hat{v}_i (z_i; z^k) := (z^k_i)^2 + 2z^k_i (z_i - z^k_i)$. It should be pointed out that the same procedure is used in the CCP algorithm (cf. Lanckriet and Sriperumbudur (2009)) for solving problem (13). In other words, the SCO can be viewed as an adapted CCP for solving problem (13).

We show in the following lemma that the sequence $\{z^k\}$ generated by Algorithm 1 always lie inside the feasible region of problem (13), with the corresponding objective value improved by every iteration.

**Lemma 3.3.** The sequences $\{z^k\}$ and $\{\xi^k\}$ generated by Algorithm 1 satisfy that

(i) $\{z^k\} \subseteq \hat{\mathcal{F}}$.

(ii) $\hat{f}(z^k) - \hat{f}(z^{k+1}) \geq \sum_{i=1}^s \delta_i (\xi_i^{k+1} - \xi_i^k)^2$.

Note that, because $\{Dz^k\} \subseteq [-x_0, 0]$ with the nonsingular matrix $D$, the sequence $\{z^k\}$ is bounded, and thus has at least one accumulation point. Based on Lemma 3.3, we obtain the following lemma.

**Lemma 3.4.** Let the sequence $\{(z^k, \xi^k)\}$ be generated by Algorithm 1 with an accumulation point $(\hat{z}, \hat{\xi})$. Then, (i) $\hat{z} \in T_{\hat{\xi}}$ and $\hat{\xi}_i = \hat{z}_i$ for $i = 1, \ldots, r$, and (ii) $f_\xi(z) \geq \hat{f}_\xi(\hat{z})$ for any $z \in T_{\hat{\xi}}$.

Let $B_{\text{SCO}}$ denote a point-to-set map associated with the SCO algorithm, that is, $z^{k+1} \in B_{\text{SCO}}(\xi^k)$. Lemma 3.4 indicates that the accumulation point $(\hat{z}, \hat{\xi})$ of the sequence $\{(z^k, \xi^k)\}$ generated by Algorithm 1 satisfies that $\hat{z}$ is the optimal solution to problem (17) with $\xi = \hat{\xi}$. This proves the closedness of $B_{\text{SCO}}$, that is, $\hat{z} \in B_{\text{SCO}}(\hat{\xi})$. It is worth mentioning that the proof can also be easily generalized to the closedness proof of a point-to-set map associated with the CCP algorithm for a D.C. program with multiple D.C. constraints in the general case.

Combining Lemma 3.4 with Proposition 3.2, we immediately obtain the following convergence result for Algorithm 1.

**Theorem 3.5.** Let $\hat{z}$ be an accumulation point of the sequence $\{z^k\}$ generated by Algorithm 1 with $\varepsilon = 0$. Then $\hat{z}$ is a KKT point of problem (13).
Theorem 3.5 shows that the SCO algorithm converges to a KKT point of problem (13). In the next section, we would utilize this property of the SCO algorithm to design a global algorithm for problem (13). In particular, the solution derived by the SCO algorithm can be used as an upper bound in our branch-and-bound approach to identify the global optimizer. In addition, we will see in our numerical experiments in Section 6 that the KKT point identified by the SCO algorithm for problem (13) always provides a good approximation for the global optimal solution.

4 | A GLOBAL OPTIMIZATION ALGORITHM

As we mentioned above, the SCO algorithm only provides a KKT point, but not a global solution to problem (13). In this section, we develop an algorithm that identifies the global optimal solution to problem (13) within a prespecified \( \varepsilon \)-tolerance by integrating the SCO algorithm with a simple convex relaxation and branch-and-bound framework. We also establish the convergence of the algorithm and estimate its complexity.

4.1 | The quadratic convex relaxation

In this subsection, we present a simple quadratic convex relaxation for problem (13) via the convex envelope technique, and then estimate the gap between it and the original problem.

To start, we consider a restricted version of problem (13), where the variables \( z_i \) \( (i = 1, \ldots, r) \) are in a subrectangle \([l, u] \subseteq \mathbb{R}^r\):

\[
\min_{z \in \mathbb{R}^m} \hat{f}(z) := z^T(D^T B^T D)z - x_0^T \Gamma Dz - \sum_{i=1}^{s} \delta_i z_i^2 \\
\text{s.t.} \quad \hat{g}(z) := \psi(z) - \sum_{i=1}^{r} \theta_i z_i^2 - \rho_1 \sum_{i=1}^{s} \delta_i z_i^2 \leq 0, \\
-\varepsilon \leq Dz \leq 0, \quad z_i \in [l_i, u_i], \quad i = 1, \ldots, r,
\]

where \([l_i, u_i] \subseteq [z_i^l, z_i^u] \) and \(z_i^l, z_i^u\) are given in Equation (15). Let \( t_i = z_i^2 \) for \( i = 1, \ldots, r \). According to McCormick (1976), the convex envelope of \( t_i = z_i^2 \) on \([l_i, u_i]\) is \( \{(t_i, z_i) : z_i^2 \leq t_i, \ t_i \leq (l_i + u_i)z_i - l_iu_i\} \). Replacing the negative quadratic term \(-z_i^2\) in the objective function and the constraint of problem (19) with its convex envelope, we can derive the following quadratic convex relaxation for problem (19):

\[
\min_{z \in \mathbb{R}^m, t \in \mathbb{R}^r} z^T(D^T B^T D)z - x_0^T \Gamma Dz - \sum_{i=1}^{s} \delta_i t_i, \\
\text{s.t.} \quad \psi(z) - \sum_{i=1}^{r} \theta_i t_i - \rho_1 \sum_{i=1}^{s} \delta_i t_i \leq 0, \\
-\varepsilon \leq Dz \leq 0, \quad z_i \in [l_i, u_i], \quad i = 1, \ldots, r, \\
z_i^2 \leq t_i, \quad t_i \leq (l_i + u_i)z_i - l_iu_i, \quad i = 1, \ldots, r.
\]
It is worth mentioning that the last constraints on \((t_i, z_i)\) in problem (20) are so-called secant cuts originally introduced in Saxena et al. (2011), and that the above convex envelope technique was also used in Luo et al. (2019, 2021). Note that the constraints \(z_i \in [l_i, u_i] (i = 1, \ldots, r)\) are redundant and hence removed.

As pointed out in Luo et al. (2019), although there exist many other strong relaxation models for QCQP, they usually involve more intensive computation. In this work, we combine the relaxation model (20) with other simple optimization techniques to develop a global algorithm for problem (13). Our choice is based on the special structure of the relaxation model (20) that has the separability of the constraints on \((t_i, z_i)\). Such a structure allows us to adopt the adaptive branch-and-cut technique based on partition on the variables \(z_1, \ldots, z_r\) in the design of the global algorithm. Moreover, the relaxation (20) has the good approximation behavior as shown in our next theorem, which compares the optimal values of problem (19) and its relaxation problem (20).

**Theorem 4.1.** Let \(\hat{f}^*_{[l,u]}\) and \(\hat{v}^*_{[l,u]}\) be the optimal value of problem (19) and its relaxation (20), respectively. Let \((\bar{z}, \bar{t})\) be the optimal solution to problem (20). Then,

\[
\hat{f}(\bar{z}) - \hat{f}^*_{[l,u]} \leq \hat{f}(\bar{z}) - \hat{v}^*_{[l,u]} \leq \frac{s}{4} \|u - l\|_\infty^2,
\]

\[
\hat{g}(\bar{z}) \leq (r + \rho_1 s) \max_{i=1,\ldots,r} \{\bar{t}_i - \bar{z}_i^2\} \leq \frac{r + \rho_1 s}{4} \|u - l\|_\infty^2,
\]

where \(\| \cdot \|_\infty\) denotes the \(\ell_\infty\)-norm on \(\mathbb{R}^r\) defined by \(\|\xi\|_\infty = \max_{i=1,\ldots,r} |\xi_i|\).

To define how good an approximate solution is, we extend the \(\epsilon\)-solution introduced in Loridan (1982) to the following definition.

**Definition 4.2.** Let \(\epsilon > 0\) and \(\hat{f}^*\) be the optimal value of problem (13). The point \(\bar{z} \in \mathcal{Z}\) is said to be an \(\epsilon\)-optimal solution to problem (13) if \(\hat{g}(\bar{z}) \leq \epsilon\) and \(\hat{f}(\bar{z}) - \hat{f}^* \leq \epsilon\).

From Theorem 4.1, we have the following corollary immediately.

**Corollary 4.3.** Let \((\bar{z}, \bar{t})\) be the optimal solution to problem (20) and \(\epsilon > 0\). If \(\max_{i=1,\ldots,r} \{\bar{t}_i - \bar{z}_i^2\} \leq \frac{\epsilon}{r + \rho_1 s}\),

then \(\bar{z}\) is an \(\epsilon\)-optimal solution of problem (19).

Theorem 4.1 indicates that when the length of the longest edge of rectangle \([l, u]\) is sufficiently short, the relaxed model (20) can provide a good approximation to problem (19). Moreover, from Theorem 4.1, we see that if \(\|u - l\|_\infty \leq \frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}\), then \(\hat{g}(\bar{z}) \leq \epsilon\) and \(\hat{f}(\bar{z}) - \hat{f}^*_{[l,u]} \leq \epsilon\). Thus, \(\bar{z}\) can be viewed as an \(\epsilon\)-optimal solution to problem (19).

## 4.2 The SCOBB algorithm

In this subsection, we present a global algorithm (called SCOBB) for problem (13) that integrates the SCO approach with the B&B framework based on the quadratic convex relaxation (20) and
Algorithm 2 (The SCOBB algorithm)

Input: $\Lambda, \Gamma, p_0, x_0, \rho_1$, and stopping criteria $\epsilon > 0$.

Output: $\epsilon$-optimal solution $z^\star$.

Step 0 (Initialization)

(i) Compute positive semi-definite matrices $B^-$ and $A^-$ by Equation (9).
(ii) Compute a nonsingular matrix $D$ such that $D^T B^- D$ and $D^T A^- D$ are diagonal. Solve the equation $Dz = -x_0$ to obtain a solution $\tilde{z}$.
(iii) Let $l^0 = (z^0_1, \ldots, z^0_i)^T$, $u^0 = (z^u_1, \ldots, z^u_i)^T$, where $z^0_i$ and $z^u_i$ are computed by Equation (12).

Step 1 Find a KKT point $z^\star$ of problem (13) by running SCO($z^0, \epsilon$) with $z^0 = \tilde{z}$. Set the first upper bound $v^\star = \hat{f}(z^\star)$.

Step 2 Solve problem (20) over $[l = l^0, u = u^0]$ to obtain the optimal solution $(z^0, v^0)$ and the first lower bound $v^0$. If $\hat{g}(z^0) \leq \epsilon$ and $\hat{f}(z^0) < v^\star$, then update the upper bound $v^\star = \hat{f}(z^0)$ and solution $z^\star = z^0$. Set $k = 0$, $\Delta^k := [l^k, u^k]$, $\Omega := \{[\Delta^k, v^k, (z^k, t^k)]\}$.

Step 3 While $\Omega \neq \emptyset$ Do (the main loop)

(S3.1) (Node selection) Choose a node $[\Delta^k, v^k, (z^k, t^k)]$ from $\Omega$ with the smallest lower bound $v^k$ and delete it from $\Omega$.

(S3.2) (Termination) If $v^k \geq v^\star - \epsilon$, then $z^k$ is an $\epsilon$-optimal solution to problem (13), stop.

(S3.3) (Partition) Choose $i^\star$ maximizing $t^k_i - (z^k_i)^2$ for $i = 1, \ldots, r$. Set $w_{i^\star} = \frac{l^k_{i^\star} + u^k_{i^\star}}{2}$,

\[
\Phi_k(w_{i^\star}) = \left\{ \left( t_{i^\star}, z_{i^\star} \right) \left| \begin{array}{c} t_{i^\star} > (l^k_{i^\star} + w_{i^\star})z_{i^\star} - l^k_{i^\star}w_{i^\star} \\ t_{i^\star} > (u^k_{i^\star} + w_{i^\star})z_{i^\star} - w_{i^\star}u_{i^\star} \end{array} \right. \right\}.
\]

If $(t^k_{i^\star}, z^k_{i^\star}) \in \Phi_k(w_{i^\star})$, then set the branching point $\beta_{i^\star} = w_{i^\star}$; else set $\beta_{i^\star} = z^k_{i^\star}$. Partition $\Delta^k$ into two sub-rectangles $\Delta^{k_1}$ and $\Delta^{k_2}$ along the edge $[l^k_{i^\star}, u^k_{i^\star}]$ at point $\beta_{i^\star}$.

(S3.4) For $j = 1, 2$, if problem (20) over $\Delta^{k_j}$ is feasible, then solve problem (20) over $\Delta^{k_j}$ to get an optimal solution $(z^{k_j}, t^{k_j})$ and optimal value $v^{k_j}$. Set $\Omega = \Omega \cup \{[\Delta^{k_1}, v^{k_1}, (z^{k_1}, t^{k_1})]\} \cup \{[\Delta^{k_2}, v^{k_2}, (z^{k_2}, t^{k_2})]\}$.

(S3.5) (Restart SCO) Set $\tilde{z} = \arg \min \{\hat{f}(z^k), \hat{f}(z^k) \}$. If $\hat{g}(\tilde{z}) \leq \epsilon$ and $\hat{f}(\tilde{z}) \leq v^\star - \epsilon$, then find a KKT point $\tilde{z}^\star$ of problem (13) by running SCO($z^0, \epsilon$) with $z^0 = \tilde{z}$, update solution $z^\star = \arg \min \{\hat{f}(\tilde{z}), \hat{f}(\tilde{z}^k) \}$ and upper bound $v^\star = \hat{f}(z^\star)$.

(S3.6) (Node deletion) Delete from $\Omega$ all the nodes $[\Delta^j, v^j, (z^j, t^j)]$ with $v^j \geq v^\star - \epsilon$. Set $k = k + 1$.

End while

the adaptive branch-and-cut rule from Luo et al. (2019). The SCOBB algorithm for problem (13) is described in Algorithm 2.

We remark that there are two main differences between the SCOBB algorithm and the existing global algorithms for QP with nonconvex constraints (cf. Audet et al. (2000); Linderoth (2005)):

(i) The SCOBB algorithm utilizes SCO to compute upper bounds for problem (13) to accelerate the convergence, and improves the upper bound by restarting SCO under certain circumstance;

(ii) Based on the special structure of the relaxation (20), which is the separability of the constraints on $(t_i, z_i)$, we adopt the adaptive branch-and-cut rule from Luo et al. (2019) to cut off the optimal solution to the relaxation problem corresponding to the node with the smallest
lower bound after each iteration, so that the lower bound will be improved (see Figure 1 for an illustration in Luo et al. (2019)).

We next present a technical result for the sequence \(\{(z^k, t^k)\} \) generated by Algorithm 2.

**Lemma 4.4.**

(i) \( t^k_i - (z^k_i)^2 \leq \frac{1}{4} (u^k_i - l^k_i)^2 \), \( i = 1, \ldots, r \) for all \( k \).

(ii) At the \( k \)th iteration, if \( \max_{i=1,\ldots,r} \{t^k_i - (z^k_i)^2\} \leq \frac{\epsilon}{r+\rho_1 s} \), then Algorithm 2 stops and \( z^* \) is an \( \epsilon \)-optimal solution to problem (13).

We can now establish the convergence of Algorithm 2 based on Lemma 4.4.

**Theorem 4.5.** Algorithm 2 can identify an \( \epsilon \)-optimal solution to problem (13) via solving at most

\[
\prod_{i=1}^{r} \left[ \frac{\sqrt{r+\rho_1 s (z^*_i - z^i_i)}}{2\sqrt{\epsilon}} \right] \text{ relaxed subproblem (20)}.
\]

Theorem 4.5 indicates that Algorithm 2 enjoys a worst-case complexity bound

\[
\Theta\left( \prod_{i=1}^{r} \left[ \frac{\sqrt{r+\rho_1 s (z^*_i - z^i_i)}}{2\sqrt{\epsilon}} \right] N \right),
\]

where \( N \) is the complexity to solve problem (20). Note that problem (20) is a convex QP. This result, thus, indicates that the complexity of Algorithm 2 mainly depends on the algorithm tolerance \( \epsilon \), the difference between the upper and lower bounds of \( z \) defined in Equation (15), and the numbers of negative eigenvalues of the matrices in the objective function and the constraint, \( r \) and \( s \), defined following Equation (12).

More specifically, the computational time of our SCOBB algorithm would grow exponentially in \( r \), which is the total number of negative eigenvalues of the matrices in the objective function and
the constraint. Recall that if we admit the convexity assumption by Brown et al. (2010), the number of negative eigenvalues would be \( r = 0 \). As stated in Brown et al. (2010), the convexity restriction actually prevents the trader from trading infinite size to obtain arbitrarily large equity, and thus ensures that the deleveraging problem is well posed. The number \( r \) corresponds to the opportunity that the trader can increase the equity by selling one or several assets in the deleveraging problem, which implies the existence of arbitrage opportunity in the market. As we estimated from the real data in Table 8, the number \( r \) is relatively small, compared with the portfolio size \( m \). This finding is also consistent with our conjecture that the theoretical arbitrage opportunities in the real market, although may exist when we overlook the bid–ask spread, are limited. As a result, our SCOBB algorithm works efficiently in practice despite the above theoretical exponential complexity, as reported in Table 8.

5 | DELIVERAGING WITH RISK OF FUTURE LIQUIDITY SHOCK

In this section, we consider a two-period model under the situation where the investor may face further shocks in the second period. With the same spirit as Brown et al. (2010), but the consideration of cross impacts, we discuss the trading priority and the preemptive deleveraging in this two-period model.

In particular, the uncertainty may arise because of unforeseen equity withdrawals or higher cash requirements to fund other areas of the business, potentially in conjunction with tighter margin constraints. Similar as Brown et al. (2010), we set the amount withdrawn as a Bernoulli random variable \( \Delta \) such that

\[
\Delta = \begin{cases} 
\delta, & \text{with probability } \pi, \\
0, & \text{with probability } 1 - \pi.
\end{cases}
\]

The required leverage ratio in the second period is \( \tilde{\rho}_2 \). If the shock does not occur, that is, \( \Delta = 0 \), the required ratio remains the same as before, that is, \( \tilde{\rho}_2 = \rho_1 \). Otherwise, the required ratio might be lower with \( \tilde{\rho}_2 = \rho_2 \in (0, \rho_1] \).

To simplify the problem, we consider in this section symmetric and non-negative cross impact matrices (similar as the assumption in Proposition 2.6) with the constant trading rate assumption. In particular, we admit the following assumptions throughout this section.

Assumption 5.1. The following conditions hold:

(i) Both price matrices \( \Lambda \) and \( \Gamma \) are non-negative and symmetric;
(ii) The investor takes constant trading rates \( y_1 \) and \( y_2 \) during time periods \((0,1]\) and \((1,2]\) respectively.

Following the notation of previous sections, the price after the second period is

\[
\tilde{p}_2 = \tilde{p}_1 + \Gamma y_2 = p_0 + \Gamma (y_1 + y_2).
\]
and the asset value at the end of the second period is

$$a_2 = p_2^T x_2 = (p_0 + \Gamma(y_1 + y_2))^T(x_0 + y_1 + y_2)$$

$$= a_0 + \begin{pmatrix} p_0 + \Gamma x_0 \\ p_0 + \Gamma x_0 \end{pmatrix}^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} \Gamma & \Gamma \\ \Gamma & \Gamma \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}.$$ 

After the two periods’ trades, the investor is left with the liability

$$l_2 = l_1 + \Delta + \int_1^2 \left( \bar{p}_1 + \Gamma \int_1^t y_2 ds + \Lambda y_2 \right)^T y_2 dt = l_1 + \Delta + \bar{p}_1^T y_2 + y_2^T \left( \Lambda + \frac{1}{2} \Gamma \right)y_2$$

$$= l_0 + \Delta + \begin{pmatrix} p_0 \\ p_0 \end{pmatrix}^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} \Lambda + \frac{1}{2} \Gamma & \frac{1}{2} \Gamma \\ \frac{1}{2} \Gamma & \Lambda + \frac{1}{2} \Gamma \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}.$$

and the second-period equity,

$$e_2 = a_2 - l_2 = e_0 - \Delta + \begin{pmatrix} \Gamma x_0 \\ \Gamma x_0 \end{pmatrix}^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} \Lambda - \frac{1}{2} \Gamma & -\frac{1}{2} \Gamma \\ -\frac{1}{2} \Gamma & \Lambda - \frac{1}{2} \Gamma \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}.$$ 

The second period leverage constraint

$$\frac{l_2}{e_2} \leq \hat{p}_2$$

can be written as a quadratic constraint on the vectors of first- and second-period trades,

$$\hat{p}_2 e_0 - l_0 - (1 + \hat{p}_2) \Delta + \begin{pmatrix} \hat{p}_2 \Gamma x_0 - p_0 \\ \hat{p}_2 \Gamma x_0 - p_0 \end{pmatrix}^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} \hat{p}_2 (\Lambda - \frac{1}{2} \Gamma) + \Lambda + \frac{1}{2} \Gamma & \frac{1}{2} (1 - \hat{p}_2) \Gamma \\ \frac{1}{2} (1 - \hat{p}_2) \Gamma & \hat{p}_2 (\Lambda - \frac{1}{2} \Gamma) + \Lambda + \frac{1}{2} \Gamma \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \geq 0. \tag{21}$$

To avoid the possibility of bankruptcy, with the same spirit as Brown et al. (2010), the investor must choose their trades in such a manner that the leverage requirement can be met no matter what the realization of $\Delta$ is. The two-period problem of the expected equity maximizing investor is then

$$\max \mathbb{E}_\Delta e_2$$

s. t. $l_1 \leq \rho_1 e_1$, $l_2 \leq \hat{p}_2 e_2$, $\forall \Delta,$

$$-x_0 \leq y_1 \leq 0, \ -x_1 \leq y_2 \leq 0, \forall \Delta,$$

where the optimization is over $y_1$ and $y_2$, where $y_1$ is in $\mathbb{R}^m$ and $y_2$ is $\{0, \delta\} \rightarrow \mathbb{R}^m$, and $x_1 = x_0 + y_1$. Because the only uncertainty comes from the Bernoulli random variable, $y_2$ in fact cor-
responds to two decision variables, $y_{2,0}$ and $y_{2,\delta}$, the trading rates during the second time period when $\Delta = 0$ and when $\Delta = \delta$. Under Assumption 5.1, when the shock does not occur ($\Delta = 0$), the optimal second-period trade is $y_{2,0} = 0$. This result can be shown by computing the gradient of the objective with respect to $y_2$ as

$$
\Gamma x_0 + \Gamma y_1 - (2\Lambda - \Gamma)y_2 = \Gamma(x_1 + y_2) - 2\Lambda y_2.
$$

With Assumption 5.1 and the box constraint $-x_1 \leq y_2 \leq 0$, all the entries of the gradient are non-negative for $y_2 \in [-x_1, 0]$. As a result, if the second-period leverage constraint is met, the optimum is achieved at $y_2 = 0$. Note that when $\Delta = 0$, the second-period leverage constraint is satisfied if $y_2 = 0$ and the first-period leverage constraint is met, that is, $l_1 \leq \rho_1 e_1$. Intuitively, if there is no shock, (1) the portfolio still meets the margin obligation; and (2) without any hedge property inside the portfolio (all the impacts are non-negative), selling assets is costly, there is no need to further rebalance with any nonzero $y_2$. In the following part, we use $y_2 = y_{2,\delta}$ to denote the second-period trades associated with the realization $\Delta = \delta$, and use $l_2$ and $e_2$ as the liabilities and equity when $\Delta = \delta$. Noting that when $\Delta = 0$, the optimal second-period equity is $e_2 = e_1$. The two-period problem is, thus, simplified as an optimization problem over $y_1, y_2 \in \mathbb{R}^m$ that maximizes the expected equity and satisfies the leverage requirements:

$$
\max_{y_1, y_2 \in \mathbb{R}^m} \mathbb{E}_{\Delta} e_2 = (1 - \pi)e_1 + \pi e_2
$$

s.t. $l_1 \leq \rho_1 e_1$, $l_2 \leq \hat{\rho}_2 e_2$,

$$
-x_0 \leq y_1 \leq 0, \quad y_2 \leq 0, \quad y_1 + y_2 \geq -x_0.
$$

We observe that the objective of problem (22) is a quadratic function in $\mathbb{R}^{2m}$,

$$
\mathbb{E}_{\Delta} e_2 = (1 - \pi)e_1 + \pi e_2
$$

$$
= e_0 - \pi \delta + x_0^T \Gamma^T y_1 + \pi x_0^T \Gamma^T y_2 - y_1^T \left( \Lambda - \frac{1}{2} \Gamma \right) y_1 + \pi y_2^T \Gamma y_1 - \pi y_2^T \left( \Lambda - \frac{1}{2} \Gamma \right) y_2
$$

$$
= e_0 - \pi \delta + \left( \frac{\Gamma x_0}{\pi \Gamma x_0} \right)^T \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right) - \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right)^T \left( \begin{array}{cc} \Lambda - \frac{1}{2} \Gamma & -\pi \frac{1}{2} \Gamma \\ -\pi \frac{1}{2} \Gamma & \pi (\Lambda - \frac{1}{2} \Gamma) \end{array} \right) \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right).
$$

With Equation (21) and the first-period leverage requirement derived in Section 2, we can formulate problem (22) as the following QP problem:

$$
\max_{y_1, y_2 \in \mathbb{R}^m} e_0 - \pi \delta + \left( \frac{\Gamma x_0}{\pi \Gamma x_0} \right)^T \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right) - \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right)^T \left( \begin{array}{cc} \Lambda - \frac{1}{2} \Gamma & -\pi \frac{1}{2} \Gamma \\ -\pi \frac{1}{2} \Gamma & \pi (\Lambda - \frac{1}{2} \Gamma) \end{array} \right) \left( \begin{array}{c} y_1 \\ y_2 \end{array} \right)
$$
\begin{align*}
\text{s. t.} & \quad \rho_1 e_0 - l_0 - (p_0 - \rho_1 \Gamma x_0)^T y_1 - y_1^T \left( \Lambda + \frac{1}{2} \Gamma + \rho_1 \left( \Lambda - \frac{1}{2} \Gamma \right) \right) y_1 \geq 0, \\
& \quad \rho_2 e_0 - l_0 - (1 + \rho_2) \delta + \left( \rho_2 \Gamma x_0 - p_0 \right)^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \\
& \quad \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} \rho_2 (\Lambda - \frac{1}{2} \Gamma) + \Lambda + \frac{1}{2} \Gamma & \frac{1}{2} (1 - \rho_2) \Gamma \\ \frac{1}{2} (1 - \rho_2) \Gamma & \rho_2 (\Lambda - \frac{1}{2} \Gamma) + \Lambda + \frac{1}{2} \Gamma \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \geq 0, \\
& \quad y_1 \leq 0, \quad y_2 \leq 0, \quad y_1 + y_2 \geq -x_0. 
\end{align*}

We first state in Proposition 5.3 the same result on preemptive deleveraging under the same convexity assumption as Brown et al. (2010):

**Assumption 5.2.** The following conditions hold:

(a) The price impact matrices satisfy

\[ \Lambda > \max \left( \frac{1 + \sqrt{\pi}}{2}, \frac{\rho_2 - 1}{\rho_2 + 1} \right) \Gamma, \quad ((1 + \rho_2) \Lambda + \Gamma) x_0 \leq p_0; \]

(b) The trade \( y_1 = -x_0/2 \) generates sufficient cash to meet the first-period margin constraint.

**Proposition 5.3.** Under Assumption 5.2, there exists a threshold shock level \( \hat{\delta} \geq 0 \) such that the optimal two-period solution satisfies \( l_1/e_1 = \rho_1 \) for all \( \delta \in [0, \hat{\delta}) \) and \( l_1/e_1 < \rho_1 \) for all feasible \( \delta > \hat{\delta} \).

The proof is the same as that of result 3 in Brown et al. (2010) and thus is omitted here. As shown in the proposition, the consideration of cross impacts does not affect the decision on whether to sell more than is required. The trader would still choose to sell more than is required in the first period when the potential shock is large.

On the contrary, we find that the effects of cross impact on which asset is prioritized for selling is non-negligible even in the two-period model. In particular, we derive the result on trading priority in the first time period without any convexity assumption as follows.

**Proposition 5.4.** Suppose assets \( i \) and \( j \) have the same initial price and holding: \( p_{0,i} = p_{0,j}, x_{0,i} = x_{0,j} \), and the \( i \)th asset is more liquid and less positively correlated with the other assets, than \( j \)th asset with the following two conditions satisfied.

(i) \( \lambda_{ii} \leq \lambda_{ij} = \lambda_{ji} < \lambda_{jj}, \gamma_{ii} \leq \gamma_{ij} = \gamma_{ji} < \gamma_{jj} \).

(ii) For any \( k \neq i, j, \lambda_{ik} \leq \lambda_{jk} \) and \( \gamma_{ik} \leq \gamma_{jk} \).

Let \( (y_{1}^*, y_{2}^*) \) be the optimal solution of problem (22). If, (a) the asset \( i \) would not be sold out during the two-period deleveraging, that is, \( y_{1,i}^* + y_{2,i}^* < x_{0,i} \); or (b) although the \( i \)th asset would be sold out, the investor sell more than half of all the assets in the first period, that is, \( y_{1,i}^* \leq -\frac{1}{2} x_{0} \), then for any \( \pi \in (0, 1), 0 < \rho_2 \leq \rho_1 \), regardless of \( \delta \), asset \( i \) is prioritized for selling. \( y_{1,i}^* \leq y_{1,j}^* \).
We notice that the conditions (i) and (ii) in Proposition 5.4 are the same as those four conditions in Proposition 2.7 under the assumption that both the impact matrices $\Gamma$ and $\Lambda$ are symmetric. That is to say, the $i$th asset that is more liquid and less positively correlated with the other assets, than the $j$th asset, is still prioritized for selling in the first period of this two-period deleveraging problem. Different from the result in Brown et al. (2010), the effects of the shock $\delta$ and the ratio of the temporary to permanent price impacts are not obvious in our model. However, we want to emphasize the existence of such an effect because the shock and the ratio would affect whether conditions (a) and (b) are satisfied. Due to the complexity introduced by the interaction among all the assets (instead of only $i$th and $j$th assets), it is difficult to show the effects explicitly. We next present the effects of $\delta$ on the trading priority and preemptive deleveraging with an example.

Example 5.5. Consider an instance of problem (22) with the impact matrices symmetrized from those in Example 2.8:

$$\Lambda = \begin{pmatrix} 0.0052 & 0.0060 & 0.0062 \\ 0.0060 & 0.0085 & 0.0076 \\ 0.0062 & 0.0076 & 0.0093 \end{pmatrix}, \Gamma = \begin{pmatrix} 0.0041 & 0.0054 & 0.0074 \\ 0.0054 & 0.0067 & 0.0079 \\ 0.0074 & 0.0079 & 0.0048 \end{pmatrix}.$$

The initial holdings and prices are $x_0 = (1, 1, 1)^T$ and $p_0 = (8, 8, 9)^T$. Suppose that the initial liability equity ratio is $l_0/e_0 = 22$ and the required leverage ratio is $\rho_1 = 18$. One can easily calculate that the initial liability is $l_0 = 23.913$, and the initial equity is $e_0 = 1.0869$, satisfying Assumption 2.4. Take $\rho_2 = 16$. We vary the second-period shock magnitude $\delta$ from zero to one with the step size being 0.05.

It is easy to verify that the conditions (i) and (ii) in Proposition 5.4 are satisfied for assets 1 and 2. In particular, the first asset is more liquid and less positively correlated with the third asset than the second one. As a result, according to Proposition 5.4., the first asset would be prioritized for selling than the second asset when the first asset is not sold out or when the investor would sell more than half in the first period.

Figure 1 shows the optimal first-period trades $y_1^*$ and the total trades $(y_1^* + y_2^*)$ versus the second-period shock magnitude ($\delta$) for the case when the future distress probability is $\pi = 0.25$. We can observe from Figure 1b that when the shock magnitude is small ($\delta < 0.15$), the first asset is not sold out in this two-period problem. According to Proposition 5.4, the first asset is prioritized for selling than the second asset, as shown in Figure 1a. In addition, Figure 1b shows that the increase of the shock magnitude $\delta$ would incur higher sales of all the assets by the investor. This is also consistent with the common sense that the investor would become more eager to sell when he/she has to withdraw a larger amount of money. As a result, the first asset could be sold out when the shock size is large ($\delta \geq 0.15$). Because Proposition 5.4 states only sufficient, but not necessary conditions for the prioritized selling, the investor could still choose to sell more of the first asset than the second one even the conditions in Proposition 5.4 are not satisfied. When the shock magnitude is large enough ($\delta \geq 0.9$), the investor start to sell more of the second asset than the first asset.

We also change the probability $\pi$ to find similar results. Figure 2a presents the optimal trades for the case when the future distress is relatively unlikely to happen with $\pi = 0.01$. We also document the leverage ratios of the optimal liquidator’s position following first-period trade as $\delta$ grows and $\pi$ changes in Figure 2b. According to Proposition 5.3, the first-period leverage constraint is active when the future shock magnitude is not too large, and could be inactive for sufficiently large
shock size. We can observe this result from Figure 2b. In addition, when the future shock is more likely to happen (i.e., with a larger probability \( \pi \)), the investor would be more likely to sell more than is required in the first period.

6 | NUMERICAL EXPERIMENTS

In this section, we present computational results of both the SCO algorithm (Algorithm 1) and the SCOBB algorithm (Algorithm 2) for the OPD problem (8). The algorithms are coded in Matlab R2013b and run on a PC (3GHz, 16GB RAM). All the linear and convex quadratic subproblems in the algorithms are solved by the QP solver in Gurobi Optimizer 9.0.2 with Matlab interface (cf. Gurobi Optimizer (2020)).

We apply our SCO and SCOBB algorithms to the OPD problem (8) with both synthetic data and historical stock data from NASDAQ\(^1\). We compare the SCOBB algorithm with the benchmark commercial software BARON (version 1.89) (cf. Sahinidis (1996)) and Gurobi Optimizer. BARON, the acronym of Branch-And-Reduce Optimization Navigator, is a global optimization package for nonconvex optimization problems. Gurobi Optimizer is a commercial optimization solver that can find a globally optimal solution of nonconvex QCQPs. We also compare the SCO algorithm with the open source software package IPOPT. IPOPT, short for “Interior Point Optimizer,” is a local optimization package for large-scale nonlinear optimization (cf. Wächter and Biegler (2006)). Source code for IPOPT can be downloaded from: https://github.com/coin-or/Ipopt. To measure the effectiveness of the SCO algorithm, the relative gap is computed by using the following formula:

\[
gap := \frac{(\text{Opt. val} - \text{Obj. val})}{\max\{1, |\text{Opt. val}|\}},
\]

where “\( \text{Opt. val} \)” denotes the global optimal value, “\( \text{Obj. val} \)” denotes the objective value at the solution found by the SCO algorithm or IPOPT.
In our numerical experiments, the stopping parameter $\epsilon$ is set at $\epsilon = 10^{-5}$. The maximum computational time is set as 3600 s. We document the notations in our computational results in Table 1.

### Table 1  Notations in our numerical experiments.

| $m$ | The number of variables |
|-----|-------------------------|
| $s, q, r$ | The rank of matrices $B^-, A^-$, and $B^- + A^-$, respectively, where $B^-$ and $A^-$ are given in Equation (11). |
| Opt.val | The average global optimal value obtained by the algorithm for 10 test instances. |
| Time | The average CPU time of the algorithm (unit: seconds) for 10 test instances. |
| Iter | The average number of iterations in the main loop of the algorithm for 10 test instances. |
| Gap | The average relative gap obtained by SCO for 10 test instances. |

### Table 2  Descriptive statistics of the six stocks in our portfolio on Aug. 1st, 2018

| stock symbol | aver. bid price (in dollar) | aver. hourly trade vol. (in million) | aver. trade size (in share) |
|--------------|----------------------------|-------------------------------------|-----------------------------|
| GM           | $37.2036                   | 0.1676                              | 111.2406                    |
| KO           | $46.3190                   | 0.1157                              | 124.9356                    |
| PEP          | $113.5936                  | 0.1251                              | 71.7464                     |
| WMT          | $88.3519                   | 0.1539                              | 89.7876                     |
| AAPL         | $200.2391                  | 1.4356                              | 108.8619                    |
| GE           | $13.3137                   | 0.3808                              | 645.9435                    |

In our numerical experiments, the stopping parameter $\epsilon$ is set at $\epsilon = 10^{-5}$. The maximum computational time is set as 3600 s. We document the notations in our computational results in Table 1.

### 6.1  Numerical experiment with real data

In this subsection, we conduct numerical experiments using historical data from NASDAQ to illustrate the application of our algorithm. As discussed before, the most important parameters in an OPD problem are the permanent and temporary price impact matrices. We, thus, first apply a linear regression model on historical stock data from NASDAQ to estimate the permanent and temporary price impact matrices, and then apply our algorithm to solve the OPD problem on these stocks.

We first describe the data set used in this numerical experiment. We use NASDAQ TotalViewITCH data, which contain message data of order events. The database documents all the order activities resulting in an update of the volumes at the best ask and bid prices, and these activities include visible orders' submissions, cancellations, and executions. For each activity, the data set documents the price, trade sign (buy or sell), volume, and order identity number. The timestamp of these activities is measured in seconds with decimal precision of at least milliseconds. We consider the optimal deleveraging problem on a portfolio consisting of six representative stocks from S&P 100 components and five different sectors: Apple Inc. (AAPL), General Electric Company (GE), General Motors Company (GM), Coca-Cola Company (KO), PepsiCo, Inc. (PEP), and Walmart Inc. (WMT).

In particular, we first estimate the temporary and permanent price impact matrices of this portfolio with the data from 10:00 a.m. to 4:00 p.m. on August 1st, 2018. We then apply our algorithm to solve the OPD problem with the estimated matrices. See Table 2 for basic descriptive statistics.
of these six stocks on August 1st, 2018.

**Multiple linear regression model.** We apply the following multiple linear regression model to estimate the price impact matrices:

$$p_{it} = p_{i0} + c_i + \sum_{j=1}^{n} \gamma_{ji}(x_{jt} - x_{j0}) + \sum_{j=1}^{n} \lambda_{ji}y_{jt}, \ i = 1, \ldots, 6,$$

where \((x_{jt} - x_{j0})\) is the cumulative signed trade volumes of the \(j\)th stock during the time period \([0, t)\) and \(y_{jt}\) is the signed trade volume of the \(j\)th stock during a small time period \([t - 1, t), t \leq T\). The constant \(c_i\) and impact parameters \(\gamma_{ji}\) and \(\lambda_{ji}\) are estimated by the linear regression model. Note that although \(y_{jt}\) is the trading rate of a continuous trading in our model, it is impossible to trade continuously in practice. We thus consider the trade volume during a relatively small time period \([t - 1, t)\) as \(y_{jt}\) in our experiment. And the instantaneous price impact is approximated by the impacts of the trades \(y_{jt}\) on the price \(p_{it}\) within the time period \([t - 1, t)\). The small time period is set to be 10 s in this estimation, while the whole time horizon \(T\) is set to be 20 min. In other words, we take the impact of trades in last 10 s on the price as an estimation of the temporary price impact, while the impact of trades in up to last 20 min on the price as an estimation of the permanent one. For each short time horizon \([t - 1, t)\), the trading volume \(y_{jt}\) would affect the price temporarily by changing \(p_{it}\), but not \(p_{i,t+1}\).

As a result, in our regression model, each data point is corresponding to one small time period, which is 10 s. That is to say, we calculate the price change \((p_{it} - p_{i0})\) and signed trade volumes every 10 s. With the data from 10:00 a.m. to 4:00 p.m. on August 1st, 2018, there are 2160 data points for each linear regression, and 18 time horizons, each of which corresponds to 20 min. More specifically, each data point includes (1) the difference between the bid price vector at the end of the small time period and that at the beginning of the time horizon containing this small time period, (2) the cumulative signed trade volume vector during the time horizon, and (3) the signed trade volume vector during the small time period. We then document the temporary and permanent price impact matrices in Table 3, where all the temporary and permanent impact coefficients have been multiplied by \(10^4\). The statistics of this linear regression are presented in Appendix C (see Table C.1). In particular, for each of the six linear regressions, we document the R-square and \(p\)-value of the F-test, which show a significant linear regression relationship between the response and the predictor variables.

As shown in Table 3, the cross impacts between stocks in the same sector (PEP and KO) are positive, while those cross-sector impacts could be negative. This finding is consistent with that by Pasquariello and Vega (2015). In fact, the stock prices of Coca-Cola Company (KO) is much more positively correlated with PepsiCo, Inc. (PEP), which is from the same sector Consumer-Non-Durable as KO, than the other assets: \(\hat{\lambda}_{23} > \hat{\lambda}_{j3}, \ \hat{\gamma}_{23} > \hat{\gamma}_{j3}, \forall j \neq 2, 3\), where \(\hat{\lambda}_{ij}\) and \(\hat{\gamma}_{ij}\) are the elements of the symmetrized price impact matrices, as discussed in Proposition 2.7. As for two assets from an identical sector, one could also be more positively correlated with the cross-sector assets than the other one. For example, KO belongs to the same sector as PEP, but has a larger market capitalization than PEP and is more positively correlated with the other assets: \(\hat{\lambda}_{2j} > \hat{\lambda}_{3j}, \hat{\gamma}_{2j} > \hat{\gamma}_{3j}, \forall j \neq 2, 3\).

In addition, the impact of the trades of one stock on the price of one another stock, although much smaller than the impact of the trades on its own price, is nonignorable. This finding is consistent with our intuition: due to possible portfolio rebalancing trades, strategical trading activities of sophisticated speculators and some correlated liquidity shocks, the prices and trades of dif-
The temporary price impact matrix

|       | GM    | KO    | PEP   | WMT   | AAPL  | GE    |
|-------|-------|-------|-------|-------|-------|-------|
| GM    | 0.081784 | -0.0039955 | -0.0082433 | -0.027238 | 0.018445 | 0.0017027 |
| KO    | -0.0076704 | 0.049933 | 0.079503 | 0.058901 | -0.020779 | -0.0010544 |
| PEP   | -0.0092156 | 0.028484 | 0.074409 | 0.011151 | -0.052514 | -0.0071483 |
| WMT   | -0.0085645 | 0.013953 | -0.006123 | 0.10158 | 0.1043 | -0.008582 |
| AAPL  | 0.00043804 | -5.49E-05 | 2.41E-05 | 0.0017344 | 0.017789 | 0.00017018 |
| GE    | 0.0045241 | 0.00086434 | 0.0021353 | 0.0095049 | -0.0049313 | 0.0032635 |

The permanent price impact matrix

|       | GM    | KO    | PEP   | WMT   | AAPL  | GE    |
|-------|-------|-------|-------|-------|-------|-------|
| GM    | 0.10236 | -0.0061319 | -0.019132 | -0.0073272 | 0.22656 | -2.42E-05 |
| KO    | -0.004774 | 0.039061 | 0.091118 | 0.050312 | 0.058154 | -0.00145 |
| PEP   | -0.0084035 | 0.059625 | 0.085173 | -0.0082462 | -0.15316 | -0.0067892 |
| WMT   | 0.008701 | 0.0084812 | 0.0034464 | 0.066367 | 0.041291 | -0.00388 |
| AAPL  | -0.0018108 | -0.00051277 | -0.001625 | -0.0044423 | 0.014685 | -0.0020722 |
| GE    | 0.0060147 | -0.0024202 | 3.46E-05 | 0.0062422 | -0.0052475 | 0.0039042 |

The $i,j$ element of the matrix denotes the temporary price impact parameter of the asset $i$ on the asset $j$, multiplied by $10^4$.

3 The different stocks may have co-movements. The fact that these estimated temporary and permanent price impact matrices are not diagonal also demonstrates the meaningfulness of our algorithm considering nondiagonal price impact matrices. Moreover, the impact matrices are not only non-diagonal, but also asymmetric, which is also found in the empirical results documented in tab. V by Pasquariello and Vega (2015). The intuition is that the relationship between these fundamentally uncorrelated stocks could be complicated and that stocks may vary in the liquidity level: a more liquid stock could be less significantly affected by trades.

Example 6.1. We consider the optimal deleveraging problem (8) with the price impact matrices $\Lambda$ and $\Gamma$ estimated and shown in Table 3. Based on the descriptive statistics in Table 2, we set the initial holding $x_0$ and the initial price $p_0$ in problem (8) as

$$x_0 = (2000, 2000, 2000, 2000, 8600, 5000)^T, \quad p_0 = (37.39, 46.42, 113.91, 88.94, 198.76, 13.4)^T.$$  

Suppose that the initial liability is $l_0 = 2262,000$ and the required leverage ratio is $\rho_1 = 18$. One can easily calculate that the initial equity is $e_0 = 87656$, and the initial liability equity ratio is $l_0/e_0 = 25.8054$, and testify that Assumption 2.4 holds.

We consider the relaxed problem by removing the leverage constraint in problem (8):

$$\max_{y \in \mathbb{R}^m} \{ e_1(y) : -x_0 \leq y \leq 0\}.  \quad (23)$$

We solve problem (23) by the ADMBB algorithm in Luo et al. (2019) to identify the global optimal solution $\hat{y}$. Denote $\rho_{\text{max}} := l_1(\hat{y})/e_1(\hat{y})$. The required leverage ratio should be less than $\rho_{\text{max}}$ to be “active” in the problem. Solving problem (23) associated with the above data yields $\rho_{\text{max}} = 25.42 > \rho_1$. In other words, our problem setting guarantees that the trader in this example is capable to meet his liability obligation by liquidating and is forced to liquidate by the leverage constraint.
According to the discussion in Section 2.3, we first reformulate the OPD problem in Example 6.1 as problem (13) with \( s = 1, r = 3 \). We then apply our SCO and SCOBB algorithms and BARON\(^5\) to problem (13). From Proposition 2.11, the optimal solution to the OPD problem can be obtained from the solution to problem (13).

Numerical results of SCO, SCOBB, and BARON are reported in Table 4, where “Opt.solution” denotes the optimal solution \( y^* \) to problem (8) derived from the solution by the algorithms. In particular, as shown in Table 4, our SCOBB algorithm can find the global optimal solution within 0.1 s. The optimal value identified by our SCOBB algorithm is the same as that by BARON. In addition, our SCO algorithm, which is proved to converge to a KKT point, also finds the global optimal solution. The result also shows that the leverage ratio constraint \( g(y^*) \leq 0 \) could be active, even in this case where the price impact matrices are asymmetric with some negative elements.

To furthermore illustrate the loss of optimality induced by admitting a constant trading rate, we also solve the optimal control for unconstrained problem (3) with the data in Example 6.1. With the impact matrices estimated in Table 3, the matrix \((\Lambda + \Lambda^T)\) has a full rank. According to Proposition 2.1 (ii), we solve problem (4) to find the optimal solution \( y^*(0) = 0 \). The optimal solution to the unconstrained control problem (3) is then \( e^C \cdot y^*(0) = 0 \), which corresponds to no trade. The corresponding optimal value is thus \( e_0 = 87656 \). It is consistent with the common sense: For the unconstrained problem where the trader is not forced to trade to meet any leverage requirement, he/she would choose no trade, unless there is any arbitrage opportunity in the market. The relative gap between the Opt.val after admitting Assumption 2.3 and the global optimal value of the unconstrained optimal control problem is thus \( (87656 - 87523.2240)/87656 = 0.1515\% \), which is reasonably small. Our strategy serves as a good approximation to the optimal strategy in the general case.

You may notice that we solve the OPD problem with continuous trading rates, while in reality, both the trades and the parameter estimation are based on discrete trades, as what we did in the multiple linear regression model. To illustrate the performance of our strategy in the reality, we simulate the market with price dynamics being

\[
p_{it} = p_{i,t-1} + \sum_{j=1}^{n} y_{jt}(x_{jt} - x_{j,t-1}) + \sum_{j=1}^{n} \lambda_{ij} y_{jt} + \epsilon_{it}, \quad i = 1, \ldots, 6, \quad t = 1, \ldots, N,
\]

where the noise \( \epsilon_{it} \sim N(0, \sigma_i^2 dt) \), the time span \( dt = 0.001 \), and the trading time \( N = T/dt = 1000 \). The volatility is set as \( \sigma_i = 1\% \cdot p_{i0}, \) for \( i = 1, \ldots, 6 \). The initial holdings and prices \( x_0 \) and \( p_0 \) are shown in Example 6.1. First, we consider the optimal strategy by solving problem (8) with a correct estimation of the price matrices shown in Table 3, which is denoted by Strategy I and shown in Table 4.

Second, as the estimation of cross impacts could be difficult in practice, we consider Strategy II, which is optimal to problem (8) with the price impact matrices, whose nondiagonal elements could deviate from the correct ones within 5% disturbance range. In particular, we randomly gen-

---

**TABLE 4** Numerical results of our algorithms and BARON for problem (8) in Example 6.1.

| Algorithm | Opt.val | Time | Iter | Opt.solution \( y^* \) | \( g(y^*) \) |
|-----------|---------|------|------|-------------------------|-------------|
| SCO       | 87523.223953 | 0.03 | 8    | \((-1478.8137, -446.7456, 0, 0, -2754.7015, -5000)^T\) | -0.000520  |
| SCOBB     | 87523.223953 | 0.07 | 4    | \((-1478.8137, -446.7456, 0, 0, -2754.7015, -5000)^T\) | -0.000520  |
| BARON     | 87523.223953 | 0.37 | 91   | \((-1478.8137, -446.7456, 0, 0, -2754.7015, -5000)^T\) | 0.000000   |
TABLE 5  Strategy III and the 10 sets of Strategy II applied in the simulated market.

| Strategy II       | (−1605.5875, −336.4272, −0.0000, −0.0000, −2758.8249, −5000.0000)$^T$ |
|-------------------|------------------------------------------------------------------------|
|                   | (−1761.1581, −433.5135, −0.0000, −0.0000, −2711.7662, −5000.0000)$^T$ |
|                   | (−1891.5165, −524.0433, −0.0000, −0.0000, −2670.6217, −4999.9999)$^T$ |
|                   | (−1791.9586, −634.4550, −0.0000, 0.0000, −2661.5768, −5000.0000)$^T$   |
|                   | (−1772.2153, −666.3233, −0.0000, −0.0000, −2657.5529, −4999.9994)$^T$ |
|                   | (−1913.1723, −690.4817, −0.0000, −0.0000, −2629.3849, −5000.0000)$^T$ |
|                   | (−2000.0000, −635.2022, −0.0000, −0.0000, −2628.9262, −4999.9998)$^T$ |
|                   | (−2000.0000, −720.7033, −0.0000, −0.0000, −2610.5023, −5000.0000)$^T$ |
|                   | (−2000.0000, −769.8370, −0.0000, −0.0000, −2598.2124, −4999.9999)$^T$ |
|                   | (−1999.9998, −852.1729, −0.0000, −0.0000, −2579.5830, −5000.0000)$^T$ |

| Strategy III      | (0.0000, −0.0004, 0.0000, −0.0002, −3447.5196, −0.0022)$^T$            |

TABLE 6  Performance of deleveraging strategies.

| Strategy     | I        | II       | III      |
|--------------|----------|----------|----------|
| Num. of cases meeting the leverage requirement | 738      | 843.3    | 375      |
|              | (42.9549)|          |          |
| Mean of qualified equity                      | 87722.3337 | 87675.3755 | 87850.8664 |
| Std. of qualified equity                      | 278.5787  | 314.1269 | 197.9435 |
|              | (14.5661)|          |          |

Remark: “Num. of cases meeting the leverage requirement” denotes the number of realizations where the investor could meet the leverage requirement after trading in the simulation. The terminal equity of the investor in the realization where the leverage requirement is met is documented as a qualified equity. “Mean of qualified equity” and “Std. of qualified equity,” respectively, denote the average and the standard deviation of all the qualified equities. For Strategy II, all the numbers are the average of the performances of the 10 different strategies, while the number in parentheses is the standard deviation among these 10 performances.

We then simulate the market 1000 times where the investor sells $y^*/N$ at time $t = 1, \ldots, N$, where $y^*$ is Strategy I, II, or III. We document the number of realizations where the investor would meet the leverage requirement and the terminal equities if he/she meet the requirement after adopting Strategies I, II, and III, respectively, in Table 6. In particular, for each realization,
we check whether the investor would satisfy the leverage requirement by taking Strategy I, II, or III and document the terminal equity if the leverage requirement is met as a qualified equity. Afterwards, we output the total number, mean, and standard deviation of the qualified equities of every strategy as its performance indices. For Strategy II, we document the average of the 10 different sets’ performances as its performance.

We can observe that the investor may not meet the leverage ratio in the end if he/she overlooks the nondiagonal elements to choose Strategy III. And the existence of an estimation error of the cross impact matrices could decrease the expected equity after trading. However, with an estimation error within 5%, the solution to problem (8) with nondiagonal elements, Strategy II, would be better than that without nondiagonal elements, in terms of the percentage of meeting the leverage requirement. Because the investor could be forced to liquidate by the exchange if the leverage requirement is unmet, the cross impacts are non-negligible in the OPD problem, even the estimation might be difficult. In addition, the discrete trades and noise would affect the performance of our continuous strategies. In fact, we find that with a more volatile noise, the investor following these deterministic strategies would become more likely to be unable to meet the leverage requirement. However, the comparison results among the performances of Strategies I, II, and III remain similar. To construct and solve a stochastic model for OPD problem is thus both challenging and meaningful, and remains open for future research.

To illustrate the effects of cross impacts on selling priority discussed in Proposition 2.7, we furthermore consider an example where the initial price $p_0$ is set as $p_0 = (40, 40, 100, 100, 200, 20)^T$. The price impact matrices, initial holding $x_0$, and initial liability $l_0$ are the same as Example 6.1. Similarly, one can easily calculate that the initial leverage ratio in such an example is $l_0/e_0 = 19.17$, the leverage ratio yielding by problem (23) is $\rho_{\text{max}} = 18.75$ and that Assumption 2.4 holds. We, thus, set the required leverage ratio $\rho_1$ varying from 16 to 8, to show the selling priority in this example and to make sure that the trader is forced to liquidate and is capable to meet the leverage requirement. The numerical results of SCOBB are presented in Table 7. One can observe that the first asset is prioritized for selling, compared with the second asset, while the initial price and holding of these two assets are the same. Although the four conditions in Proposition 2.7, which provide a sufficient, but not necessary condition for prioritized sales, do not hold in this case. Inspired by conditions (ii) and (iv), we can easily testify that the first asset (GM) is less positively correlated with the other assets than the second one (KO) by

$$\sum_{k \neq 1,2} \hat{\lambda}_{1k} - \sum_{k \neq 1,2} \hat{\lambda}_{2k} = -0.0294 < 0, \quad \sum_{k \neq 1,2} \hat{\gamma}_{1k} - \sum_{k \neq 1,2} \hat{\gamma}_{2k} = -0.0940 < 0,$$

$$\sum_{k=1}^{6} (\gamma_{k1} - \gamma_{1k}) = -0.1942 < 0, \quad \sum_{k=1}^{6} (\gamma_{k2} - \gamma_{2k}) = -0.1343.$$
As a result, this illustrative example demonstrates that the cross impacts are important in the deleveraging problem and the selling priority inside the portfolio.

**Example 6.2.** We consider the OPD problems with portfolios consisting of up to 20 assets from NASDAQ during one sample month and with the historical data from the same database as Example 6.1. The describe statistics of the assets are documented in Appendix C (see Table C.2). We estimate the impact parameters in a similar way as described in Example 6.1. The price impact matrices and related statistics are available upon request. In particular, we consider 10 different sets of data, and obtain 10 instances of OPD problem for each portfolio size $m = 10, 15, 20$.

We assume that the initial liability equity ratio is $l_0/e_0 = 25$ and the required leverage ratio is $\rho_1 = 18$, and testify that Assumption 2.4 holds for all instances. As in Example 6.1, we transform the OPD problem into problem (10) and then problem (13). We then apply our SCOBB and SCO algorithms separately to problem (13), and three commercial optimization solvers to problem (10). The optimal solution to the original problem can be obtained from the solution to the transformed problems by using Proposition 2.11. According to the identified optimal solutions, the leverage ratio constraint $g(y^*) \leq 0$ could be inactive for some instances. That is to say, traders might sell more than the leverage requirement in the OPD problem, leading to $g(y^*) < 0$. As we discussed in Section 2.2, the intuition is that there might be some hedge properties inside the portfolio, such that the trader could utilize nonzero cross impact among different assets.

In Table 8, we summarize the average performances of SCOBB, SCO and the commercial optimization solvers, BARON, Gurobi, and IPOPT, on these small-scaled instances of OPD problem with the historical data on these 20 assets from NASDAQ. As shown in Table 8, our SCOBB algorithm can identify the global optimal solution for all test instances within around 20 s, while BARON (resp. Gurobi) can only solve 22 (resp. 28) out of 30 instances within its default computational time (500 s). Moreover, it takes much longer time for BARON and Gurobi than SCOBB to identify the solution to these solved instances with $m = 15, 20$. We also observe that the KKT point found by the SCO algorithm provides a good approximation for the global optimizer in terms of the gap. Meanwhile, the corresponding objective value at the local solution identified by IPOPT would have a relative gap larger than 1% to the global optimal value. Thereupon, our SCO algorithm yields a better local solution than the IPOPT in terms of the relative gap.

In addition, the computational time of our SCOBB algorithm grows exponentially in terms of the number of branches in the B&B framework, and thus the number of negative eigenvalues of the matrices $\hat{\Lambda} - \frac{1}{2} \hat{\Gamma}$ and $\hat{\Lambda} + \frac{1}{2} \hat{\Gamma}$ in the objective function and the constraint. As a result, we document the average numbers of negative eigenvalues in these instances with historical data in Table 8. The numerical results demonstrate that the total number of negative eigenvalues of the two matrices, $r$ defined in Equation (12), is usually not too large in the real problem. This fact guarantees that our SCOBB algorithm can solve OPD problem within short computational time, and thus serve a good purpose as a practical algorithm for the investors.

**6.2 Numerical results for randomly generated test problems**

Due to the limitation of our dataset, we only consider problem with up to 20 assets in the experiments with real data. To illustrate the efficiency of our algorithm for such a NP-hard problem, in this subsection, we furthermore randomly generate large-scale instances where the parameters
| Size | Rank | SCOB | BARON | Gurobi | SCO | IPOPT |
|------|------|------|-------|--------|-----|-------|
| m    | s    | q    | r     | Opt.val | Opt.val | Opt.val | Opt.val | Opt.val | Opt.val | Opt.val | Opt.val | Opt.val |
| 10   | 1.7  | 0.8  | 2.5   | 21.90   | 1578851.10 | 1701524.76 | 2859387.05 | 3074218.44 | 3799765.81 | 3889242.28 | 4009776.06 |
| 15   | 3.1  | 2.0  | 5.1   | 19.80   | 20.27   | 191.9   | 13.90   | 2934135.19 | 1299.2   | 112.25   | 13.94   |
| 20   | 4.8  | 3.4  | 8.2   | 20.54   | 20.27   | 1299.2  | 20.27   | 4009776.06 | 13.94    | 0.37     | 10e−8   |

**Remark:** The number in parentheses in the column of “Opt.val” for BARON and Gurobi stands for the number of the instances for which BARON or Gurobi can verify the global optimality of the solution within 500 s. “Time” and “Opt.val” for BARON and Gurobi denote the average CPU time and optimal value for the instances that are globally solved by BARON or Gurobi in the 10 instances, respectively.
are generated following the patterns observed from the real data in the last subsection. We then show the efficiency of our SCOBB algorithm and the accuracy of our SCO algorithm on the OPD problem (8) with these randomly generated instances and compare the performances of these two algorithms.

As shown in Example 6.2, the number of negative eigenvalues is limited in the real problem. Our experiments are thus restricted to instances with a few negative eigenvalues, that is, a small $r$ defined in Equation (12).

**Example 6.3.** Inspired by Example 6.1 with real data, we randomly generate the parameters $(\Lambda, \Gamma, p_0, x_0)$ in our test instances as follows. The entries of $p_0$ are drawn from $U[10, 100]$, that is, uniformly distributed within interval $[10, 100]$, and entries of $x_0$ are drawn from $U[500, 1000]$. The matrices $\hat{\Lambda} - \frac{1}{2} \hat{\Gamma}$ and $\hat{\Lambda} + \frac{1}{2} \hat{\Gamma}$ are randomly generated\(^6\) to have exactly $s$ and $q$ negative eigenvalues, respectively, with elements lying inside $[10^{-6}, 10^{-5}]$. We consider the optimal deleveraging problem with an initial liability equity ratio $l_0/e_0 = 25$ and the required leverage ratio $\rho_1 = 18$. Assumption 2.4 is satisfied for each random test instance. We then apply our SCO and SCOBB algorithms and BARON to randomly generated ten test instances for varied fixed sizes and numbers of negative eigenvalues.

We first compare the performances of SCOBB, SCO, and the commercial optimization solvers BARON and Gurobi for small-scale instances of problem (8), that is, a small-scale portfolio with the number of assets being $m = 20$ or $m = 30$, with $r = 5, 8, 10, 15$. The results documented in Table 9 show that the SCOBB algorithm is able to find the global optimal solution within 100 s for all the test instances, while BARON (resp. Gurobi) can only identify the global optimal solution in 13 (resp. 33) out of 80 test instances within 1000 s. Also, both BARON and Gurobi require more CPU time than SCOBB for the solved instances. In addition, for most of test instances, BARON and Gurobi only reported the best solution obtained within 1000 s and failed to verify the global optimality of the obtained solution. The computational time of both BARON and Gurobi grows rapidly in terms of the number of assets. As shown in Table 9, BARON fails to identify the global optimal solution for all the instances with $m = 30$. Meanwhile, the CPU time of our SCOBB algorithm grows rapidly in terms of $r$, which results from the branch-and-bound framework. We also observe that our SCO algorithm always provides the global optimal solutions for all the instances in our numerical experiments. Meanwhile, the relative gap of the IPOPT could be larger than 10 percentage.

Table 10 summarizes the average numerical results of our algorithms SCOBB and SCO, and the commercial software IPOPT for medium- and large-scale instances of problem (8) with $r = 6, 8, 10$. As one can see from Table 10 that SCOBB can effectively find the global optimal solution for all the test instances of problem (8) within 1600 s, with the number of assets up to $m = 500$. In addition, the SCO algorithm can always obtain the global optimal solutions in less CPU time for all the medium- and large-scale instances in our experiments. Meanwhile, the SCO algorithm has much smaller relative gap than the IPOPT.

To conclude, we compare the performances of these two algorithms: SCO and SCOBB in our numerical experiments, and summarize the differences as follows. First, from the aspect of optimality, our SCOBB algorithm is proved to identify the global optimal solution, while the SCO algorithm is proved to converge to a KKT point. However, from our numerical results reported in Sections 6.1 and 6.2, our SCO algorithm always successfully find the global optimal solution for all tested instances. Second, for the computational complexity, we theoretically show that the com-
| Size | m | s | q | r | $p_{max}$ | SCOBB Opt.val | Time | Iter | BARON Opt.val | Time | Iter | $t_{1}(\mathbf{y}^{*})$ | Gurobi Opt.val | Time | Iter | SCO Gap Time | IPOPT Gap Time |
|------|---|---|---|---|------|-------------|------|-----|--------------|------|-----|----------------|-------------|------|-----|---------|--------------|
| 20   | 2 | 3 | 5 | 23.10 | 32101.20 | 0.37 | 24.8 | 18 | 29804.77(1) | 180.09 | 20 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30600.25(5) | 320.86 | 20 | 0 | 0.08 | 14.5 | 18 | 7.20e−2 | 0.04 |
| 20   | 4 | 4 | 8 | 22.93 | 31192.75 | 1.53 | 129.4 | 18 | 27422.59(2) | 245.43 | 20 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30025.98(6) | 397.78 | 20 | 0 | 0.11 | 21.8 | 18 | 9.16e−2 | 0.04 |
| 20   | 5 | 5 | 10 | 22.93 | 31329.06 | 2.83 | 235.8 | 18 | 3025.98(6) | 397.78 | 20 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 31391.68(9) | 217.83 | 20 | 0 | 0.11 | 22.0 | 18 | 1.04e−1 | 0.08 |
| 20   | 7 | 8 | 15 | 22.58 | 33949.11 | 47.83 | 30635 | 18 | 33233.47(4) | 420.41 | 20 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 33347.88(8) | 121.04 | 20 | 0 | 0.11 | 22.0 | 18 | 1.04e−1 | 0.08 |
| 30   | 2 | 3 | 5 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 33233.47(4) | 420.41 | 30 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30600.25(5) | 320.86 | 30 | 0 | 0.09 | 12.8 | 18 | 7.07e−2 | 0.04 |
| 30   | 4 | 4 | 8 | 23.53 | 50733.41 | 3.84 | 193.3 | 18 | 49192.19(2) | 385.50 | 30 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30600.25(5) | 320.86 | 30 | 0 | 0.13 | 16.4 | 18 | 1.04e−1 | 0.03 |
| 30   | 5 | 5 | 10 | 23.34 | 46827.93 | 4.62 | 243.6 | 18 | 49192.19(2) | 385.50 | 30 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30600.25(5) | 320.86 | 30 | 0 | 0.16 | 22.3 | 18 | 1.16e−1 | 0.06 |
| 30   | 7 | 8 | 15 | 23.26 | 48078.36 | 53.74 | 2498.4 | 18 | 49192.19(2) | 385.50 | 30 | 23.46 | 47919.23 | 0.40 | 19.1 | 18 | 30600.25(5) | 320.86 | 30 | 0 | 0.16 | 22.3 | 18 | 1.16e−1 | 0.06 |

**Remark:** The number in parentheses in the column of “Opt.val” and “Gurobi” for BARON and Gurobi stands for the number of the instances for which BARON or Gurobi can verify the global optimality of the solution within 1000 s. “Time” and “Opt.val” for BARON and Gurobi denote the average CPU time and optimal value for the instances that are globally solved by BARON or Gurobi in the 10 instances, respectively. The sign “−” stands for the situations where the method fails to identify the global solution within 1000 s in all the 10 instances.
putational time of SCOBB algorithm grows exponentially in the number of eigenvalues $r$, while the time complexity of SCO algorithm is left for future research. In our numerical experiments, our SCOBB algorithm can solve the problem with up to 500 assets within 1600 s, while the SCO algorithm can be solved in several minutes for all the cases. As a result, we believe that our SCO algorithm provides a good approximation for the global optimal solution for these investors who care about the computational time, and that our SCOBB algorithm provides an efficient global optimal solution for the investors who value the solution accuracy.

7 | CONCLUSIONS

Market impact is crucial for traders, especially large traders, to deleverage a portfolio during a short time period. Cross impact among the assets are nonignorable, due to financial constraints or the portfolio rebalancing trades from sophisticated speculators. In this paper, we investigate an optimal deleveraging problem with cross impact. The objective is to maximize equity while meeting a prescribed leveraged ratio requirement. We obtain some analytical insights regarding the optimal deleveraging strategy. Especially, we find that with the cross price impact, traders may sell more, instead of precisely satisfy the leverage requirement, if there are hedge properties inside the portfolio. In addition, among all the assets inside the portfolio, the trader should sell more of the asset which is more liquid and less correlated with the other assets. These analytical properties are verified by our empirical and numerical experiments.

Our optimization model is a nonconvex quadratic program with nonhomogeneous quadratic and box constraints, which is NP-hard. We reformulate the nonconvex quadratic program as a D.C. program via spectral decomposition and SD, and then propose two efficient algorithms for it: the SCO algorithm and the SCOBB algorithm. We show that our SCO algorithm converges
to a KKT point of the transformed problem, while the SCOBBI algorithm can efficiently identify the global $\varepsilon$-optimal solution. We establish the global convergence of the SCOBBI algorithm and estimate its complexity.

We estimate the price impact matrices with the historical data from NASDAQ in our numerical experiments, to demonstrate the application of our algorithms. Randomly generated instances are also considered. According to our numerical experiments, our SCOBBI algorithm can effectively identify the global optimal solution to medium- and large-scale instances of the optimal deleveraging problem with limited number of negative eigenvalues of the matrices in the quadratic terms in our optimization problem. A future research topic is to investigate whether we can develop effective global algorithms for general optimal deleveraging problem without this restriction. Meanwhile, although we cannot prove the global optimality, our SCO algorithm always provides the global optimal solution in our numerical experiments for all the instances within short computational time, and thus also serve the purpose as an efficient algorithm for the optimal deleveraging strategy.

We assume in this paper that the price impact is linear. Empirical results (e.g., Loeb (1983); Lillo et al. (2003)), however, show that the temporary price impact function could be nonlinear. It, thus, remains open for future research to investigate whether we can develop an effective global algorithm for the optimal deleveraging problem with both cross impacts and nonlinear price impact functions. In addition, we only consider small-scale instances in the two-period model. Another future research topic is to develop an efficient algorithm for the large-scale two-period model with cross impact.

ACKNOWLEDGMENTS
The authors would like to thank the Associate Editor and the two anonymous referees for the detailed comments and valuable suggestions, which have improved the final presentation of the paper. The research of H. Luo and H. Wu is supported by NSFC Grants 12271485 and 11871433 and the Zhejiang Provincial NSFC Grants LZ21A010003 and LY18A010011. The research of Y. Chen is supported by NSFC Grants 72001105 and 72171109. The research of D. Li is supported by Hong Kong Research Grants Council under Grants 14213716 and 14202017.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are available at https://github.com/hezhiluo/OPD. These data were derived from the data from LOBSTER (https://lobsterdata.com/). Restrictions apply to the availability of these source data, which were used under license for this study. Data are available from the authors with the permission of LOBSTER.

ORCID
Yuanyuan Chen https://orcid.org/0000-0002-0023-3045

ENDNOTES
1 All the data used in Section 6 can be downloaded on https://github.com/hezhiluo/OPD.
2 Data are provided by LOBSTER (https://lobsterdata.com/).
3 We also consider different dates and up to twenty different stocks. The results are similar.
4 We want to point out that, although in the theoretical models without the consideration of bid–ask spread, the price impact matrices are usually assumed to be symmetric and positive semi-definite to avoid arbitrage possibilities, an asymmetric price impact matrix with limited number of negative eigenvalues does not necessarily bring in arbitrage opportunities in the real market with a positive bid–ask spread.
In this example, BARON cannot solve problem (10) within its default time limit, 500 s. We thus apply BARON to problem (13), instead of problem (10) in this example.

The way we generate matrices with an arbitrary fixed number of negative eigenvalues is documented in Appendix B.

REFERENCES

Almgren, R., & Chriss, N. (2000). Optimal execution of portfolio transactions. *Journal of Risk*, 3(2), 5–39.

Andrade, S. C., Chang, C., & Seasholes, M. S. (2008). Trading imbalances, predictable reversals, and cross-stock price pressure. *Journal of Financial Economics*, 88(2), 406–423.

Audet, C., Hansen, P., Jaumard, B., & Savard, G. (2000). A branch and cut algorithm for nonconvex quadratically constrained quadratic programming. *Mathematical Programming*, 87, 131–152.

Ben-Tal, A., & den Hertog, D. (2014). Hidden conic quadratic representation of some nonconvex quadratic optimization problems. *Mathematical Programming*, 143, 1–29.

Ben-Tal, A., & Teboulle, M. (1995). Hidden convexity in some nonconvex quadratically constrained quadratic programming. *Mathematical Programming*, 72(1), 51–63.

Benzaquen, M., Mastromatteo, I., Eisler, Z., & Bouchaud, J. P. (2017). Dissecting cross-impact on stock markets: An empirical analysis. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(2), 023406. 10.1088/1742-5468/aa53f7

Bertsimas, D., Hummel, P., & Lo, A. W. (1999). Optimal control of execution costs for portfolios. *Computing in Science & Engineering*, 1(6), 40–53.

Brown, D. B., Carlin, B., & Lobo, M. S. (2010). Optimal portfolio liquidation with distress risk. *Management Science*, 56(11), 1997–2014.

Burer, S., & Vandenbussche, D. (2008). A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations. *Mathematical Programming*, 113(2), 259–282.

Burer, S., & Vandenbussche, D. (2009). Globally solving box-constrained nonconvex quadratic programs with semidefinite-based finite branch-and-bound. *Computational Optimization and Applications*, 43(2), 181–195.

Carlin, B. I., Lobo, M. S., & Viswanathan, S. (2007). Episodic liquidity crises: Cooperative and predatory trading. *The Journal of Finance*, 62(5), 2235–2274.

Cartea, A., Gan, L., & Jaimungal, S. (2019). Trading co-integrated assets with price impact. *Mathematical Finance*, 29(2), 542–567.

Chen, J., & Burer, S. (2012). Globally solving nonconvex quadratic programming problems via completely positive programming. *Mathematical Programming Computation*, 4, 33–52.

Chen, J., Feng, L., & Peng, J. (2015). Optimal deleveraging with nonlinear temporary price impact. *European Journal of Operational Research*, 244, 240–247.

Chen, J., Feng, L., Peng, J., & Ye, Y. (2014). Analytical results and efficient algorithm for optimal portfolio deleveraging with market impact. *Operation Research*, 62(1), 195–206.

Cont, R., & Waglath, L. (2016). Fire sales forensics: Measuring endogenous risk. *Mathematical Finance*, 26(4), 835–866.

D’Hulster, K. (2009). *The leverage ratio: A new binding limit on banks*. http://www.worldbank.org/financialcrisis/pdf/leverage-ratio-web.pdf

Ding, X. D., Luo, H. Z., Wu, H. X., & Liu, J. Z. (2021). An efficient global algorithm for worst-case linear optimization under uncertainties based on nonlinear semidefinite relaxation. *Computational Optimization and Applications*, 80(1), 89–120.

Fleming, J., Kirby, C., & Ostdiek, B. (1998). Information and volatility linkages in the stock, bond, and money markets. *Journal of Financial Economics*, 49(1), 111–137.

Floudas, C. A., & Visweswaran, V. (1994). Quadratic optimization. In R. Horst, & P. M. Pardalos (Eds.), *Handbook of global optimization* (pp. 217–270). Kluwer Academic Publishers.

Gould, N. I. M., & Toint, P. L. (2002). Numerical methods for large-scale non-convex quadratic programming. In *Trends in industrial and applied mathematics* (Amrtaaar, 2001). *Applied optimization* (Vol. 72, pp. 149–179).

Gurobi Optimizer. (2020). *Gurobi Interactive Shell (win64), Version 9.0.2 Copyright (c). Gurobi Optimization, LLC.*

Holthausen, R. W., Leftwich, R. W., & Mayers, D. (1990). Large-block transactions, the speed of response, and temporary and permanent stock-price effects. *Journal of Financial Economics*, 26(1), 71–95.
Hong, L. J., Yang, Y., & Zhang, L. (2011). Sequential convex approximations to joint chance constrained programs: A Monte Carlo approach. *Operations Research, 59*(3), 617–630.

Jiang, R. J., Li, D., & Wu, B. Y. (2018). SOCP reformulation for the generalized trust region subproblem via a canonical form of two symmetric matrices. *Mathematical Programming, 169*, 531–563.

Kyle, A. S., & Xiong, W. (2001). Contagion as a wealth effect. *The Journal of Finance, 56*(4), 1401–1440.

Lanckriet, G., & Sriperumbudur, B. K. (2009). On the convergence of the concave-convex procedure. In *Advances in neural information processing systems* (Vol. 22, pp. 1759–1767).

Le Thi, H. A., & Pham Dinh, T. (1997). Solving a class of linearly constrained indefinite quadratic problems by D.C. algorithms. *Journal of Global Optimization, 11*, 253–285.

Le Thi, H. A., Huynh, V. N., & Pham Dinh, T. (2014). DC programming and DCA for general DC programs. In *Advanced computational methods for knowledge engineering* (pp. 15–35). Springer International Publishing.

Li, Y., Guo, J., Lai, K. K., & Shi, J. (2020). Optimal portfolio liquidation with cross-price impacts on trading. *Operations Research, 22*, 1–20.

Lillo, F., Farmer, J., & Mantegna, R. (2003). Master curve for price-impact function. *Nature, 421*, 129–130.

Linderoth, J. (2005). A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Mathematical Programming, 103*, 251–282.

Lipp, T., & Boyd, S. (2016). Variations and extension of the convex–concave procedure. *Optimization and Engineering, 17*(2), 263–287.

Loeb, T. F. (1983). Trading cost: The critical link between investment information and results. *Financial Analysts, 39*(3), 39–44.

Loridan, P. (1982). Necessary conditions for $\varepsilon$-optimality. *Mathematical Programming Studies, 19*, 140–152.

Luo, H. Z., Bai, X. D., Lim, G., & Peng, J. (2019). New global algorithms for quadratic programming with a few negative eigenvalues based on alternative direction method and convex relaxation. *Mathematical Programming Computation, 11*(1), 119–171.

Luo, H. Z., Ding, X. D., Peng, J. M., Jiang, R. J., & Li, D. (2021). Complexity results and effective algorithms for worst-case linear optimization under uncertainties. *INFORMS Journal on Computing, 33*(1), 180–197.

Madhavan, A. (2000). Market microstructure: A survey. *Journal of Financial Markets, 3*(3), 205–258.

McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs: Part I-convex underestimating problems. *Mathematical Programming, 10*(1), 147–175.

Moré, J. J. (1993). Generalization of the trust region problem. *Optimization Methods and Software, 2*(1), 189–209.

Nesterov, Y. (1998). Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software, 9*(1-3), 141–160.

Newcomb, R. W. (1961). On the simultaneous diagonalization of two semi-definite matrices. *Quarterly of Applied Mathematics, 19*(2), 144–146.

Palacios-Gomez, F., Lasdon, L., & Enquist, M. (1982). Nonlinear optimization by successive linear programming. *Management Science, 28*(10), 1106–1120.

Pasquariello, P., & Vega, C. (2015). Strategic cross-trading in the U.S. stock market. *Review of Finance, 19*(1), 229–282.

Pham Dinh, T., & Le Thi, H. A. (2014). Recent advances in DC programming and DCA. In *Transactions on computational intelligence XIII* (pp. 1–37). Springer.

Polik, I., & Terlaky, T. (2007). A survey of the S-lemma. *SIAM Review, 49*(3), 371–418.

Sahinidis, N. V. (1996). BARON: A general purpose global optimization software package. *Journal of Global Optimization, 8*, 201–205.

Saxena, A., Bonami, P., & Lee, J. (2011). Convex relaxation of nonconvex mixed integer quadratically constrained programs: Projected formulations. *Mathematical Programming, 130*, 359–413.

Schöneborn, T., & Schied, A. (2009). *Liquidation in the face of adversity: Stealth versus sunshine trading*. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1007014

Sias, R. W., Starks, L. T., & Titman, S. (2001). The price impact of institutional trading. *Available at SSRN 283779.*

Strekalovsky, A. S., & Minarchenko, I. M. (2018). A local search method for optimization problem with d.c. inequality constraints. *Applied Mathematical Modelling, 58*, 229–244.

Tsoukalas, G., Wang, J., & Giesecke, K. (2019). Dynamic portfolio execution. *Management Science, 65*(5), 2015–2040.

Wächter, A., & Biegler, L. T. (2006). On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming, 106*, 25–57.
APPENDIX A: PROOFS OF STATEMENTS
In this appendix, we provide the proofs of all the theorems, lemmas and propositions in the main body of the paper.

A.1 | Proofs in Section 2
Proof of Proposition 2.1. According to the dynamics and problem shown in Equations (1)–(3), we can derive the terminal equity as

\[ e(1) = \tilde{p}(1)^T x(1) - l_0 - \int_0^1 (\tilde{p}(t) + \Lambda y(t))^T y(t) dt. \]

With the initial holdings \( x_0 \) and initial prices \( p_0 \), we have

\[
e(1) = p_0^T x_0 + \int_0^1 (\tilde{p}(t)^T dx(t) + x(t)^T d\tilde{p}(t)) - l_0 - \int_0^1 (\tilde{p}(t) + \Lambda y(t))^T y(t) dt \]

\[ = p_0^T x_0 - l_0 + \int_0^1 (\tilde{p}(t)^T y(t) + x(t)^T \Lambda y(t) - (\tilde{p}(t) + \Lambda y(t))^T y(t)) dt \]

\[ = p_0^T x_0 - l_0 + \int_0^1 (x(t)^T \Gamma y(t) - y(t)^T \Lambda y(t)) dt, \]  

(A.1)

and thus reformulate problem (3) as

\[
\max p_0^T x_0 - l_0 + \int_0^1 f_i(\tilde{p}(t), x(t), y(t)) dt \\
\text{s. t. } dx(t) = y(t) dt, \; d\tilde{p}(t) = \Gamma y(t) dt,
\]

where \( f_i(p(t), x(t), y(t)) = x(t)^T \Gamma y(t) - y(t)^T \Lambda y(t) \). It is a deterministic problem with Hamiltonian function being

\[
H(x(t), \tilde{p}(t), y(t), \lambda_1(t), \lambda_2(t)) = x(t)^T \Gamma y(t) - y(t)^T \Lambda y(t) + \lambda_1(t)^T y(t) + \lambda_2(t)^T \Gamma y(t),
\]
and the adjoint equation being

\[
\begin{bmatrix}
\dot{\lambda}_1(t) \\
\dot{\lambda}_2(t)
\end{bmatrix} = - \begin{bmatrix}
\nabla x H \\
\nabla p H
\end{bmatrix} = \begin{bmatrix}
-\Gamma y(t) \\
0
\end{bmatrix}.
\] (A.2)

The optimal trading rate satisfies

\[
\Gamma^T x(t) - (\Lambda + \Lambda^T) y(t) + \lambda_1(t) + \Gamma^T \lambda_2(t) = 0 \Rightarrow \Gamma^T d x(t) = (\Lambda + \Lambda^T) dy(t) - d\lambda_1(t) - \Gamma^T d\lambda_2(t).
\]

Together with Equations (2) and (A.2), we have

\[
\Gamma^T y(t) dt = \Gamma^T d x(t) = (\Lambda + \Lambda^T) dy(t) \Leftrightarrow (\Lambda + \Lambda^T) dy(t) = (\Gamma^T - \Gamma) y(t) dt. \quad \text{(A.3)}
\]

(i) If the matrix \((\Lambda + \Lambda^T)\) has a full rank and the permanent price impact matrix \(\Gamma\) is symmetric, then from Equation (A.3), the optimal trading rate satisfies \(dy(t) = 0\) and thus is constant. This proves the Statement (i).

(ii) If the matrix \((\Lambda + \Lambda^T)\) has a full rank and the matrix \(C\) is defined as \((\Lambda + \Lambda^T)^{-1}(\Gamma^T - \Gamma)\), then from Equation (A.3), the optimal trading rate satisfies \(dy(t) = C \cdot y(t) dt\) and thus takes the form of \(y(t) = e^{tC} \cdot y(0)\). We observe from Equation (2) that the holding at time \(t\) is \(x(t) = x_0 + \int_0^t e^{sC} y(0) ds\).

According to Equation (A.1), the terminal equity is

\[
e_1 = p_0^T x_0 - l_0 + \int_0^1 \left( x(t)^T \Gamma y(t) - y(t)^T \Lambda y(t) \right) dt
\]

\[
= e_0 + \int_0^1 \left( x_0 + \int_0^t e^{sC} y(0) ds \right)^T \Gamma e^{tC} y(0) - \left( e^{tC} y(0) \right)^T \Lambda e^{tC} y(0) \right) dt
\]

\[
= e_0 + \left( x_0^T \Gamma \left( \int_0^1 e^{tC} dt \right) y(0) + y(0)^T \left( C^{-1} \right)^T \left( \int_0^1 (e^{tC} - I)^T \Gamma e^{tC} dt \right) y(0) \right) - y(0)^T \left( \int_0^1 (e^{tC})^T \Lambda e^{tC} dt \right) y(0)
\]

\[
= e_0 + \left( x_0^T \bar{\gamma} y(0) - y(0)^T (\bar{\Lambda} - B) y(0) \right),
\]

where

\[
\bar{\gamma} = \Gamma \left( \int_0^1 e^{tC} dt \right) = \Gamma (e^{C} - I) C^{-1}, \quad \bar{\Lambda} = \left( \int_0^1 (e^{tC})^T \Lambda e^{tC} dt \right), \quad B = (C^{-1})^T \left( \int_0^1 (e^{tC} - I)^T \Gamma e^{tC} dt \right).
\]

Together with the boundary condition \(-x_0 \leq y(0) \leq 0\), we reformulate problem (3) as problem (4) to solve the optimal initial trading rate \(y^*(0)\). □
Proof of Proposition 2.5. By Assumption 2.4, \( l_1(-x_0) < 0 \) and hence \( x_0^T(\Lambda + \frac{1}{2} \Gamma)x_0 - e_0 < 0 \). Via a simple calculation, we derive the following:

\[
e_1(-x_0) = -x_0^T\left(\Lambda + \frac{1}{2} \Gamma\right)x_0 + e_0 > 0,
\]

\[
\rho_1 e_1(-x_0) - l_1(-x_0) = -(1 + \rho_1)\left[x_0^T\left(\Lambda + \frac{1}{2} \Gamma\right)x_0 - e_0\right] > 0.
\]

The proof is completed. \( \square \)

Proof of Proposition 2.6. Since the feasible set of problem (8) is compact, there exists an optimal solution \( y^* = (y_1^*, \ldots, y_m^*)^T \). Let us denote \( g_0(y) = \rho_1 e_1(y) - l_1(y) \). Assume to the contrary that the leverage constraint is not active at the optimum solution. Then \( g_0(y^*) < 0 \). Denote

\[
\left\{ \begin{array}{l}
g_i(y) = y_i - \beta_i, \quad g_{m+i}(y) = -y_i + \alpha_i, \quad i = 1, \ldots, m, \\
I(y) = \{ i : g_i(y) = 0, \ i = 0, 1, \ldots, 2m \}.
\end{array} \right.
\]

Clearly, \( 0 \notin I(y^*) \). It is thus easy to see that the gradients \( \nabla g_i(y^*) \), \( i \in I(y^*) \) are linear independent. According to the first-order optimality condition, there exists \( \mu^* = (\mu_0^*, \mu_1^*, \ldots, \mu_m^*, \mu_{m+1}^*, \ldots, \mu_{2m}^*)^T \geq 0 \) satisfying the following conditions:

\[
-\nabla e_1(y^*) + \mu_0^* \nabla g_0(y^*) + \sum_{i=1}^{m} \left( \mu_i^* \nabla g_i(y^*) + \mu_{m+i}^* \nabla g_{m+i}(y^*) \right) = 0, \tag{A.4}
\]

\[
\mu_0^* g_0(y^*) = 0, \tag{A.5}
\]

\[
\mu_i^* g_i(y^*) = 0, \quad i = 1, \ldots, m, \tag{A.6}
\]

\[
\mu_{m+i}^* g_{m+i}(y^*) = 0, \quad i = 1, \ldots, m, \tag{A.7}
\]

where

\[
g_i(y) = y_i, \quad g_{m+i}(y) = -y_i - x_{0,i}, \quad i = 1, \ldots, m.
\]

Since \( g_0(y^*) \neq 0 \), we have \( \mu_0^* = 0 \). Thus, equality (A.4) becomes

\[
\sum_{j=1}^{m} \left[ (\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) y_j^* - \gamma_{ji} x_{0,j} \right] + \mu_i^* - \mu_{m+i}^* = 0, \quad i = 1, \ldots, m, \tag{A.8}
\]

where \( \hat{\lambda}_{ij} = \lambda_{ij} + \lambda_{ji} \) and \( \hat{\gamma}_{ij} = \gamma_{ij} + \gamma_{ji} \). Since matrices \( \Gamma \) and \( \Lambda \) are non-negative, we have \( \hat{\lambda}_{ij} \geq 0 \) and \( \hat{\gamma}_{ij} \geq 0 \) for all \( i, j \). Since \( x_0 > 0 \), it is easy to see from Equations (A.6) and (A.7) that \( \mu_i^* \) and \( \mu_{m+i}^* \), \( 1 \leq i \leq m \), cannot be positive simultaneously. We consider the following cases:
Case (i): $\mu^*_i = 0$, $\mu^*_m+i \neq 0$. From Equation (A.7), $y^*_i = -x_{0,i}$. Then, from $\mu^* \geq 0$ and Equation (A.8), we obtain

$$0 < \mu^*_m+i = \sum_{j=1}^{m} \left( \hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij} \right) y^*_j - y_j x_{0,j} \leq \sum_{j=1}^{m} \hat{\lambda}_{ij} y^*_j + \frac{1}{2} \sum_{j=1}^{m} (y_{ij} - y_{ji}) x_{0,j} \leq 0,$$  \hspace{1cm} (A.9)

where the second inequality is due to $y^* \geq -x_0$ and $\hat{\gamma}_{ij} \geq 0$ for all $i, j$, while the last inequality is due to $y^* \leq 0$, $\hat{\lambda}_{ij} \geq 0$ and the assumption that $\Gamma$ is symmetric. This is a contradiction.

Case (ii): $\mu^*_i = \mu^*_m+i = 0$. Since $x_0 > 0$ and $\Gamma$ is non-negative, it follows from Equation (A.8) that

$$\sum_{j=1}^{m} (\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) y^*_j = \sum_{j=1}^{m} x_{0,j} y^*_j > 0.$$  \hspace{1cm} (A.10)

Let us denote $J = \{ j : \hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij} < 0, j = 1, \ldots, m \}$. Since $y^* \leq 0$ and by Equation (A.10), we must have $J \neq \emptyset$. Let us define

$$j_0 \in \arg \max_{j \in J} \{ \hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij} \}. \hspace{1cm} (A.11)$$

Note that $-x_0 \leq y^* \leq 0$ implies that $(\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) y^*_j \leq 0$ for $j \notin J$ and $(\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) y^*_j \leq -(\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) x_{0,j}$ for $j \in J$. It then follows from Equation (A.10) that

$$\sum_{j=1}^{m} x_{0,j} y^*_j \leq \sum_{j \in J} (\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) y^*_j \leq (\hat{\lambda}_{ij_0} - 0.5 \hat{\gamma}_{ij_0}) y^*_j - \sum_{j \in J \setminus \{j_0\}} (\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) x_{0,j},$$

which in turn implies that

$$(\hat{\lambda}_{ij_0} - 0.5 \hat{\gamma}_{ij_0}) y^*_j \geq \sum_{j=1}^{m} y_{ij} x_{0,j} + \sum_{j \in J \setminus \{j_0\}} (\hat{\lambda}_{ij} - 0.5 \hat{\gamma}_{ij}) x_{0,j}$$

$$= \sum_{j \notin J} y_{ij} x_{0,j} + \sum_{j \in J \setminus \{j_0\}} \left[ \hat{\lambda}_{ij} x_{0,j} + \frac{1}{2} (y_{ij} - y_{ji}) x_{0,j} \right] + x_{0,j_0} y_{j_0,i}$$

$$= \sum_{j \notin J} y_{ij} x_{0,j} + \sum_{j \in J \setminus \{j_0\}} \hat{\lambda}_{ij} x_{0,j} + x_{0,j_0} y_{j_0,i},$$

due to the assumption that $\Gamma$ is symmetric. Since $\hat{\lambda}_{ij_0} - 0.5 \hat{\gamma}_{ij_0} < 0$ by $j_0 \in J$, the above inequality yields

$$y^*_j \leq \frac{\sum_{j \notin J} y_{ij} x_{0,j} + \sum_{j \in J \setminus \{j_0\}} \hat{\lambda}_{ij} x_{0,j} + x_{0,j_0} y_{j_0,i}}{\hat{\lambda}_{ij_0} - 0.5 \hat{\gamma}_{ij_0}}.$$  \hspace{1cm} (A.12)

Combining Equation (A.12) with $y^*_j \geq -x_{0,j_0}$ and $y_{j_0,i} = y_{i,j_0}$, we derive

$$0 \leq y^*_j + x_{0,j_0} \leq \frac{\sum_{j \in J} x_{0,j} y_{ij} + \sum_{j \notin J} \hat{\lambda}_{ij} x_{0,j}}{\hat{\lambda}_{ij_0} - 0.5 \hat{\gamma}_{ij_0}} < 0,$$
which gives a contradiction.

Since for any \(1 \leq i \leq m\), the above cases are not possible, we must have \(\mu_i^* \neq 0\), \(\mu_{m+i}^* = 0\) for all \(1 \leq i \leq m\). Then from Equation (A.6), \(y_i^* = 0\), \(i = 1, \ldots, m\). Thus \(g_0(y^*) \leq 0\) becomes \(l_0 - \rho_1 e_0 \leq 0\), which contradicts the assumption that \(l_0 - \rho_1 e_0 > 0\). This completes the proof of the proposition.

**Proof of Proposition 2.7.** Assume to the contrary that \(y^*\) is the optimal solution and \(-x_{0,j} \leq y_j^* < y_i^* \leq 0\). We will find a direction \(d \in \mathbb{R}^m\) along which the liability \(l_1\) is strictly decreasing but the equity \(e_1\) is strictly increasing. Let \(d = (d_1, \ldots, d_m)^T\), where \(d_i = -1\), \(d_j = 1\), and \(d_k = 0\), \(\forall k \neq i, j\).

By the expressions of \(l_1(y)\) and \(e_1(y)\), we have

\[
\nabla l_1(y) = (2\Lambda + \Gamma)y + p_0, \quad \nabla e_1(y) = -(2\Lambda - \Gamma)y + \Gamma^T x_0. \tag{A.13}
\]

Because \(p_{0,i} = p_{0,j}\) and \(x_{0,j} = x_{0,j}\), by conditions (i) and (ii) and the fact that \(y_j^* - y_i^* < 0\), we can derive from Equation (A.13) that

\[
\nabla l_1^T(y^*)d = \sum_{k=1}^m \left(2\lambda_{jk} + \hat{\gamma}_{jk} - 2\hat{\lambda}_{ik} - \hat{\gamma}_{ik}\right)y_k^* \\
\leq \left[2(\hat{\lambda}_{jj} - \hat{\lambda}_{ij}) + \hat{\gamma}_{jj} - \hat{\gamma}_{ij}\right]y_j^* + \left[2(\hat{\lambda}_{ji} - \hat{\lambda}_{ii}) + \hat{\gamma}_{ji} - \hat{\gamma}_{ii}\right]y_i^* \\
\leq \left[2(\hat{\lambda}_{jj} - \hat{\lambda}_{ij}) + \hat{\gamma}_{jj} - \hat{\gamma}_{ij}\right](y_j^* - y_i^*) < 0.
\]

Meanwhile, Equation (A.13) implies that

\[
\nabla e_1^T(y^*)d = \sum_{k=1}^m \left[(2\lambda_{jk} + \hat{\gamma}_{jk} - 2\hat{\lambda}_{ik} - \hat{\gamma}_{ik})y_k^* + (\gamma_{kj} - \gamma_{ki})x_{0,k}\right]. \tag{A.14}
\]

Because \(-x_{0,k} \leq y_k^* \leq 0\) holds for all \(k\), we follow from condition (i), (ii), and (iv) that

\[
(2\hat{\lambda}_{jk} + \hat{\gamma}_{jk} - 2\hat{\lambda}_{ik} - \hat{\gamma}_{ik})y_k^* + (\gamma_{kj} - \gamma_{ki})x_{0,k} \\
= 2(\hat{\lambda}_{ik} - \hat{\lambda}_{jk})y_j^* + (\gamma_{kj} - \gamma_{ki})x_{0,k} \\
\geq (\gamma_{kj} - \gamma_{ki})x_{0,k} \\
= \frac{1}{2}(\gamma_{kj} - \gamma_{ki})x_{0,k} \geq 0, \quad \forall k \neq i, j. \tag{A.15}
\]

Thus, according to the inequality (A.15), we can relax the gradient (A.14) to

\[
\nabla e_1^T(y^*)d \geq 2(\hat{\lambda}_{ij} - \hat{\lambda}_{jj})y_j^* + (\gamma_{jj} - \gamma_{ji})x_{0,j} \\
+ (\gamma_{ij} - \gamma_{ji})x_{0,i} \\
= 2(\hat{\lambda}_{ij} - \hat{\lambda}_{jj})y_j^* + (\gamma_{jj} - \gamma_{ji})y_j^* + 2(\hat{\lambda}_{ii} - \hat{\lambda}_{jj})y_j^* \\
+ (\gamma_{jj} - \gamma_{ii})x_{0,i} \\
\geq 2(\hat{\lambda}_{ij} - \hat{\lambda}_{jj})(y_j^* - y_i^*) + (\gamma_{jj} - \gamma_{jj})y_j^* + (\gamma_{jj} - \gamma_{ii})x_{0,i}
\]

This completes the proof of the proposition.
\[= 2(\hat{\lambda}_{ij} - \hat{\lambda}_{jj})(y_j^* - y_i^*) > 0,\]

where the second inequality follows from conditions (i) and (iii) and the fact \(-x_{0,i} = -x_{0,j} \leq y_j^* < y_i^* \leq 0\), the second equality is due to \(\hat{y}_{jj} = y_{jj}, \hat{y}_{ji} = \hat{y}_{ij}, \hat{y}_{ii} = y_{ii}\), while the last inequality is due to condition (i) and the fact that \(y_j^* < y_i^*\).

Therefore, there exist a sufficiently small \(\delta > 0\) such that \(-x_{0,i} \leq y^* + \delta d \leq 0\) and the trading strategy \(y^* + \delta d\) leads to strictly larger equity and smaller liability after trading. This contradicts the optimality of \(y^*\). \(\square\)

The proof of Lemma 2.10. In order to find a nonsingular matrix \(D\) in the simultaneous diagonalizability method, we give the proof of Lemma 2.10 slightly different from that in Newcomb (1961) as follows.

Let \(A\) and \(B\) be two \(n \times n\) real symmetric positive semidefinite matrices, and let \(r = \text{rank}(A + B)\) and \(s = \text{rank}(B)\). Then there exists a nonsingular matrix \(Q \in \mathbb{R}^{n \times n}\) such that \(Q^T (A + B)Q = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix}\), where \(I_r\) denotes the unit matrix of order \(r\). Let \(Q^T BQ = \begin{pmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{pmatrix}\) with \(G_{11} \in \mathbb{R}^{r \times r}\). Since \(A\) is positive semidefinite, we have \(Q^T (A + B)Q \succeq Q^T BQ\) and hence \(\begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix} \succeq \begin{pmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{pmatrix}\), from which we can infer that \(Q^T BQ = \begin{pmatrix} G_{11} & 0 \\ 0 & 0 \end{pmatrix}\) and \(I_r \succeq G_{11} \succeq 0\). Note that \(\text{rank}(G_{11}) = \text{rank}(B) = s\). Let \(\delta_1, \ldots, \delta_s\) be the positive eigenvalues of matrix \(G_{11}\). Since \(I_r \succeq G_{11} \succeq 0\), we obtain \(s < r\) and \(0 < \delta_i \leq 1\) for \(i = 1, \ldots, s\). Since \(G_{11}\) is a real symmetric matrix of order \(r\), there exists an orthogonal matrix \(D_r \in \mathbb{R}^{r \times r}\) such that \(D_r^T G_{11} D_r = \text{diag}(\delta_1, \ldots, \delta_s, 0, \ldots, 0)\). Let us choose \(D = Q \begin{pmatrix} D_r & 0 \\ 0 & I_{n-r} \end{pmatrix}\). It is then easy to check that \(D\) is nonsingular, \(D^T BD = \text{diag}(\delta_1, \ldots, \delta_s, 0, \ldots, 0)\) and \(D^T AD = \text{diag}(\delta_1, \ldots, \delta_r, 0, \ldots, 0)\), where \(\delta_i = 1 - \delta_i\) for \(i = 1, \ldots, s\) and \(\delta_i = 1\) for \(i = s + 1, \ldots, r\). Furthermore, since \(D\) is nonsingular and \(q = \text{rank}(A)\), there are \(q\) nonzero numbers in \(\delta_1, \ldots, \delta_r\).

\(\square\)

A.2 Proofs in Section 3.2

Proof of Lemma 3.1. (i) We first observe that

\[\hat{g}_\xi(z) = \hat{g}(z) + \sum_{i=1}^{r} \hat{\theta}_i (z_i - \xi_i)^2 + \rho_1 \sum_{i=1}^{s} \delta_i (z_i - \xi_i)^2. \tag{A.16}\]

Since \(z \in \hat{F}_\xi\) and \(\hat{\xi}_i = \hat{z}_i\) for \(i = 1, \ldots, r\), we can infer from Equation (A.16) that \(\hat{y} \in \hat{F}_\xi\) and hence \(\hat{F}_\xi \neq \emptyset\). It is also easy to see that \(\hat{F}_\xi\) is a closed convex set. In addition, we can follow from Equation (A.16) that \(\hat{g}_\xi(z) \geq \hat{g}(z)\) and hence \(\hat{F}_\xi \subseteq \hat{F}\).

(ii) By contradiction, suppose that \(\text{int}\hat{F}_\xi = \emptyset\). This means that

\[\hat{g}_\xi(z) \geq 0, \quad \forall z \in Z.\]
Since \( \bar{z} \in \hat{P} \) and \( \bar{\xi}_i = \bar{z}_i \) for \( i = 1, \ldots, r \), we can deduce from Equation (A.16) that \( 0 \leq \hat{g}_\xi(\bar{z}) = \hat{g}(\bar{z}) \leq 0 \), which in turn implies that \( \hat{g}(\bar{z}) = 0 \) and

\[
\bar{z} \in \arg\min_{z \in \tilde{Z}} \hat{g}_\xi(z).
\]

From the necessary optimality condition, there exist \( \eta \) and \( \mu \in \mathbb{R}_+^m \) such that

\[
\nabla \hat{g}_\xi(\bar{z}) + D(\eta - \mu) = 0, \quad \eta^T D \bar{z} = 0, \quad \mu^T (D \bar{z} + x_0) = 0. \tag{A.17}
\]

Since \( \bar{\xi}_i = \bar{z}_i \) for \( i = 1, \ldots, r \), from Equation (A.16), we have \( \nabla \hat{g}_\xi(\bar{z}) = \nabla \hat{g}(\bar{z}) \). Clearly, \(-x_0 \leq \bar{y} \leq 0\) and \( g(\bar{y}) = \hat{g}(\bar{z}) = 0 \), so \( \bar{y} \) is a feasible solution to problem (10). We also follow that \( \nabla \hat{g}(\bar{z}) = P \nabla g(\bar{z}) \). Note that \( P \) is nonsingular. It then follows from Equation (A.17) that

\[
\nabla g(\bar{z}) + (\eta - \mu) = 0, \quad \eta^T \bar{y} = 0, \quad \mu^T (\bar{y} + x_0) = 0. \tag{A.18}
\]

Let us define \( I_1 = \{ i \mid \bar{y}_i = 0, \ i = 1, \ldots, m \} \), and \( I_2 = \{ i \mid \bar{y}_i = -x_{0,i}, \ i = 1, \ldots, m \} \). It then follows from Equation (A.18) that

\[
\nabla g(\bar{y}) + \sum_{i \in I_1} \eta_i e_i - \sum_{i \in I_2} \mu_i e_i = 0,
\]

where \( e_i \) is the \( i \)-th unit vector of \( \mathbb{R}^m \). This, together with the feasibility of \( \bar{y} \), contradicts the fact that the gradients \( \nabla g(\bar{y}) \), \( e_i \), \( i \in I_1 \cup I_2 \) are linear independent.

\[ \square \]

**Proof of Lemma 3.3.** (i) Since \( z^0 \in \hat{P} \) and \( \xi^0 = (\xi_1^0, \ldots, \xi_r^0)^T \) with \( \xi_i^0 = z_i^0 \) for \( i = 1, \ldots, r \), we obtain from Equation (A.16) that \( z^0 \in \hat{P}_{\xi^0} \) and hence \( \hat{P}_{\xi^0} \neq \emptyset \). We see from step 1 that \( z^1 \in \hat{P}_{\xi^0} \). It then follows from Lemma 3.1 that \( z^1 \in \hat{P} \). By induction, we conclude that \( \{z^k\} \subseteq \hat{P} \).

(ii) From claim (i), we have \( \{z^k\} \subseteq \hat{P} \). Note from step 1 that \( \xi^k = (\xi_1^k, \ldots, \xi_r^k)^T \) with \( \xi_i^k = z_i^k \) for \( i = 1, \ldots, r \). It is easy to check from Equation (A.16) that \( z^k \in \hat{P}_{\xi^k} \) for all \( k \). Note that

\[
f_\xi(z) = f(z) + \sum_{i=1}^s \delta_i (z_i - \xi_i)^2. \tag{A.19}
\]

From step 1, since \( z^{k+1} \) is the optimal solution of problem (17) with \( \xi = \xi^k \) and by \( z^k \in \hat{P}_{\xi^k} \), we can deduce that

\[
f(z^{k+1}) + \sum_{i=1}^s \delta_i \left( z_i^{k+1} - \xi_i^k \right)^2 \leq f(z^k) + \sum_{i=1}^s \delta_i \left( z_i^k - \xi_i^k \right)^2,
\]

which, together with the fact that \( \xi_i^k = z_i^k \) for \( i = 1, \ldots, r \) and \( s < r \), yields

\[
f(z^k) - f(z^{k+1}) \geq \sum_{i=1}^s \delta_i \left( z_i^{k+1} - \xi_i^k \right)^2, \quad \forall k \geq 1.
\]

The proof is finished. \[ \square \]
Proof of Lemma 3.4. By Lemma 3.3, we have that

\[ f(z^k) - f(z^{k+1}) \geq \sum_{i=1}^{s} \delta_i \left( \xi_{i}^{k+1} - \xi_{i}^{k} \right)^2 \]  
(A.20)

for all k. Thus, \( \{f(z^k)\} \) is a nonincreasing sequence. By Lemma 3.3, \( \{z^k\} \subseteq \hat{F} \). Since \( F \) is compact, \( \{f(z^k)\} \) is bounded. Hence \( \{f(z^k)\} \) converges and consequently

\[ f(z^k) - f(z^{k+1}) \rightarrow 0 \text{ as } k \rightarrow \infty. \]  
(A.21)

By the fact that \( \delta_i > 0 \) for \( i = 1, \ldots, s \), it then follows from Equation (A.20) that

\[ \lim_{k \rightarrow \infty} \left( \xi_{i}^{k+1} - \xi_{i}^{k} \right) = 0, \quad i = 1, \ldots, s. \]  
(A.22)

Now let \((\tilde{z}, \tilde{\xi})\) be an accumulation point of the sequence \( \{(z^k, \xi^k)\} \), and let \( K \subset \{1, 2, \ldots\} \) be such that \( \{(z^k, \xi^k)\}_K \rightarrow (\tilde{z}, \tilde{\xi}) \). The closedness of \( \hat{F} \) and \( \{z^k\} \subseteq \hat{F} \) imply \( \tilde{z} \in \hat{F} \). Note from step 1 that \( \tilde{\xi}^k_i = z_i^k \), \( i = 1, \ldots, r \) for all \( k \), so \( \tilde{\xi}^k_i = \tilde{z}_i^k \) for \( i = 1, \ldots, r \). We then follow from Equation (A.16) that \( \tilde{z} \in \hat{F}_{\tilde{\xi}} \). In the following, we will prove that for any \( z \in \hat{F}_{\tilde{\xi}} \), it holds that

\[ \hat{f}_{\tilde{\xi}}(z) \geq \hat{f}_{\tilde{\xi}}(\tilde{z}). \]  
(A.23)

Indeed, for any \( z \in \hat{F}_{\tilde{\xi}} \), we consider two cases as follows.

Case (i): \( z \in \hat{F}_{\tilde{\xi}} \) for a sufficiently large \( k \in K \). In this case, since \( z^{k+1} \) is an optimal solution of problem (17) with \( \xi = \xi^k \), we have

\[ \hat{f}_{\tilde{\xi}}(z) \geq \hat{f}_{\tilde{\xi}}(z^{k+1}). \]  
(A.24)

Since \( \xi_{i}^{k+1} = z_{i}^{k+1}, i = 1, \ldots, r \) for all \( k \) and \( s < r \), we follow from Equation (A.19) that

\[ \hat{f}_{\tilde{\xi}}(z^{k+1}) = \hat{f}(z^{k+1}) + \sum_{i=1}^{s} \delta_i \left( \xi_{i}^{k+1} - \xi_{i}^{k} \right)^2. \]  
(A.25)

Note from Equation (A.21) that

\[ \lim_{k \rightarrow \infty, k \in K} \hat{f}(z^{k+1}) = \hat{f}(\tilde{z}). \]  

Note from Equation (A.19) and \( \tilde{\xi}_i = \tilde{z}_i, i = 1, \ldots, r \) that \( \hat{f}(\tilde{z}) = \hat{f}_{\tilde{\xi}}(\tilde{z}) \). It then follows from Equations (A.25) and (A.22) that

\[ \lim_{k \rightarrow \infty, k \in K} \hat{f}_{\tilde{\xi}}(z^{k+1}) = \hat{f}_{\tilde{\xi}}(\tilde{z}). \]  
(A.26)

Also, \( \lim_{k \rightarrow \infty, k \in K} \hat{f}_{\tilde{\xi}}(z) = \hat{f}_{\tilde{\xi}}(z) \). Then taking the limit in Equation (A.24) as \( k \rightarrow \infty \) and \( k \in K \) yields Equation (A.23).

Case (ii): \( z \notin \hat{F}_{\tilde{\xi}} \) for a sufficiently large \( k \in K \). Since \( \tilde{z} \in \hat{F} \) and \( \tilde{\xi}_i = \tilde{z}_i, i = 1, \ldots, r \), by Lemma 3.1, \( \text{int} \hat{F}_{\tilde{\xi}} \neq \emptyset \). Then there exists \( \hat{z} \in \hat{F} \) such that \( \hat{g}_{\tilde{\xi}}(\hat{z}) < 0 \). Let \( \delta = -\hat{g}_{\tilde{\xi}}(\hat{z}) > 0 \). Since
\( \hat{g}_\xi(\hat{z}) \) is continuous in \( \xi \) and by \( \{\xi^k\}_{k} \rightarrow \xi \), we have that for sufficiently large \( k \in \mathcal{K} \), one has
\[
\hat{g}_{\xi^k}(\hat{z}) \leq -\delta/2 < 0.
\]

For given \( z \in \hat{F}_\xi \setminus \mathcal{T}_\xi \), let us define \( \rho_k = \max \{0, \hat{g}_{\xi^k}(z)\} \). Clearly, \( \rho_k > 0 \). Since \( \{\xi^k\}_{k} \rightarrow \xi \) and \( z \in \hat{F}_\xi \), it is easy to check that
\[
\lim_{k \rightarrow \infty, k \in \mathcal{K}} \rho_k = 0. \tag{A.27}
\]

Let us define
\[
\lambda_k = 2\rho_k/(2\rho_k + \delta). \tag{A.28}
\]

Clearly, \( 0 < \lambda_k < 1 \). Let us choose \( \hat{z}^k = (1 - \lambda_k)z + \lambda_k \hat{z} \). Since \( \hat{g}_{\xi^k}(z) \) is a convex function in \( z \) and by Equation (A.28), we have
\[
\hat{g}_{\xi^k}(\hat{z}^k) \leq (1 - \lambda_k)\hat{g}_{\xi^k}(z) + \lambda_k \hat{g}_{\xi^k}(\hat{z}) \leq (1 - \lambda_k)\rho_k + \lambda_k(-\delta/2) = 0,
\]
which in turn implies that \( \hat{z}^k \in \mathcal{T}_{\xi^k} \). Therefore, we have
\[
\text{dist}(z, \mathcal{T}_{\xi^k}) := \min \{||w - z|| : w \in \mathcal{T}_{\xi^k}\} \leq ||\hat{z}^k - z||.
\]

Note that \( \hat{z}^k - z = \lambda_k(\hat{z} - z) \). Since \( \mathcal{T}_{\xi^k} \) is a nonempty closed convex set, by the projection theorem, there exists a unique \( \hat{z}^k \in \mathcal{T}_{\xi^k} \) such that
\[
||\hat{z}^k - z|| = \text{dist}(z, \mathcal{T}_{\xi^k}) \leq ||\hat{z}^k - z|| = \lambda_k||\hat{z} - z|| \leq 2\rho_k\delta^{-1}||\hat{z} - z||.
\]

From the above relation and Equation (A.27), we obtain
\[
\lim_{k \in \mathcal{K} \rightarrow \infty} \hat{z}^k = z. \tag{A.29}
\]

Since \( \hat{z}^k \in \mathcal{T}_{\xi^k} \) and \( z^{k+1} \) is an optimal solution of problem (17) with \( \xi = \xi^k \), we have
\[
\hat{f}_{\xi^k}(z^{k+1}) \leq \hat{f}_{\xi^k}(\hat{z}^k). \tag{A.30}
\]

By Equation (A.26), \( \lim_{k \in \mathcal{K} \rightarrow \infty} \hat{f}_{\xi^k}(z^{k+1}) = \hat{f}_{\xi}(\hat{z}) \). By Equation (A.29) and \( \{\xi^k\}_{k} \rightarrow \xi \), we obtain
\[
\lim_{k \in \mathcal{K} \rightarrow \infty} \hat{f}_{\xi^k}(\hat{z}^k) = \hat{f}_{\xi}(\hat{z}). \tag{A.31}
\]
Taking the limit in Equation (A.30) as \( k \rightarrow \infty \) and \( k \in \mathcal{K} \) then gives rise to Equation (A.23). The proof of the theorem is completed.

\( \square \)

### A.3 Proofs in Section 4

**Proof of Theorem 4.1.** We first have \( \hat{f}^*_{[\ell, u]} \geq \hat{\delta}^*_{[\ell, u]}(\hat{y}) \). It follows that
\[
\hat{f}(\hat{z}) - \hat{f}^*_{[\ell, u]} \leq \hat{f}(\hat{z}) - \hat{\delta}^*_{[\ell, u]} = \sum_{i=1}^{s} \delta_i (\bar{t}_i - \bar{z}^2_i).
\]
\[
\begin{align*}
&\leq \sum_{i=1}^{s} (\bar{t}_i - \bar{z}_i^2) \\
&\leq \sum_{i=1}^{s} \left[ -\bar{z}_i^2 + (l_i + u_i)\bar{z}_i - l_i u_i \right] \\
&\leq \frac{1}{4} \sum_{i=1}^{s} (u_i - l_i)^2 \leq \frac{S}{4} \|u - l\|_\infty^2,
\end{align*}
\]

where the second and third inequalities follow from the fact that \(\delta_i \in (0, 1]\) for \(i = 1, \ldots, s\) and the constraints on \((t_i, z_i)\) in Equation (20).

Similarly, using the feasibility of \((\bar{z}, \bar{t})\) and the fact that \(0 < \theta_i \leq 1\) and \(0 < \delta_i \leq 1\) for each \(i\), we can derive that

\[
\hat{g}(\bar{z}) = \psi(\bar{z}) - \sum_{i=1}^{r} \theta_i \bar{t}_i - \rho_1 \sum_{i=1}^{s} \delta_i \bar{z}_i + \sum_{i=1}^{r} \theta_i (\bar{t}_i - \bar{z}_i^2) + \rho_1 \sum_{i=1}^{s} \delta_i (\bar{t}_i - \bar{z}_i^2)
\]

\[
\leq \sum_{i=1}^{r} (\bar{t}_i - \bar{z}_i^2) + \rho_1 \sum_{i=1}^{s} (\bar{t}_i - \bar{z}_i^2)
\]

\[
\leq (r + \rho_1 s) \max_{i=1,\ldots,r} \{\bar{t}_i - \bar{z}_i^2\} \leq \frac{r + \rho_1 s}{4} \|u - l\|_\infty^2.
\]

This completes the proof. \(\square\)

**Proof of Lemma 4.4.** Recall that \((z^k, t^k)\) is the optimal solution of problem (20) over \(\Delta^k\). From the proof of Theorem 4.1, we can obtain Statement (i). Now we prove Statement (ii). Since \((z^k, t^k)\) is the optimal solution of problem (20) over \(\Delta^k\) and \(v^k\) is its optimal value, by Theorem 4.1, we can infer that \(\hat{f}(z^k) - v^k \leq \epsilon\) and \(\hat{g}(z^k) \leq \epsilon\). Thus, \(z^k\) is \(\epsilon\)-feasible for problem (13). From steps 2 and (S3.5) of Algorithm 2, we see that \(\hat{f}(z^k) \geq v^* = \hat{f}(z^*)\) for all \(k\). Thus,

\[
v^k - v^* \geq v^k - \hat{f}(z^k) \geq -\epsilon,
\]

so the stopping criterion of the algorithm is satisfied and then the algorithm stops. Let \(\hat{f}^*\) denotes the optimal value of problem (13). By step (S3.1), \(v^k\) is the smallest lower bound. Thus \(\hat{f}^* \geq v^k\). It then follows from Equation (A.31) that

\[
\hat{f}(z^*) = v^* \leq v^k - \epsilon \leq \hat{f}^* + \epsilon.
\]

This, together with \(\hat{g}(z^*) \leq \epsilon\), implies that \(z^*\) is a global \(\epsilon\)-solution to problem (13). \(\square\)

**Proof of Theorem 4.5.** At the \(k\)th iteration, if the selected node \([\Delta^k, v^k, (z^k, t^k)]\) with the smallest lower bound \(v^k\) satisfies either \(u_i^k - l_i^k \leq \frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}\) for the chosen \(i^*\) in partition or \(\|u^k - l^k\|_\infty \leq \frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}\), then by Lemma 4.4, the algorithm stops and yields an \(\epsilon\)-optimal solution \(z^*\) to problem (13). At the \(k\)th iteration, if the algorithm does not stop, then by Lemma 4.4(ii), \(s_i^k - (t_i^k)^2 > \frac{\epsilon}{r + \rho_1 s}\)
for the chosen index $i^*$ in step (S3.3). Hence, by Lemma 4.4(i), $u_{i^*}^k - l_{i^*}^k > \frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}$, that is, the $i^*$th edge of the subrectangle $\Delta^k$ must be longer than $\frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}$. According to step (S3.3), it will be divided at either point $z_{i^*}^k$ or the midpoint $(u_{i^*}^k + l_{i^*}^k)/2$. Note that if $u_{i^*}^k - l_{i^*}^k \leq \frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}$, then the $i$th direction will never be chosen in step (S3.3) as a branching direction at the $k$-iteration. This means that all the edges of the subrectangle corresponding to a node with the smallest lower bound selected at every iteration will never be shorter than $\frac{2\sqrt{\epsilon}}{\sqrt{r + \rho_1 s}}$. Therefore, every edge of the initial rectangle will be divided into at most $\left\lceil \frac{\sqrt{r + \rho_1 s} (z_{i,u}^k - z_{i,l}^k)}{2\sqrt{\epsilon}} \right\rceil$ subintervals. In other words, to obtain an $\epsilon$-optimal solution to problem (13), the total number of the relaxed subproblem (20) required to be solved in all the runs of Algorithm 2 is bounded above by

\[ \prod_{i=1}^{r} \left[ \frac{\sqrt{r + \rho_1 s} (z_{i,u}^k - z_{i,l}^k)}{2\sqrt{\epsilon}} \right]. \]

This proves the conclusion of the theorem. \hfill \Box

A.4 | Proofs in Section 5

Proof of Proposition 5.4. Assume that $(y_1^*, y_2^*)$ is the optimal solution and $-x_{0,j} \leq y_{1,j}^* < y_{1,i}^* \leq 0$. We will show that such a solution can not be optimal. To this end, we will show that there exists a feasible direction from $(y_1^*, y_2^*)$ along which the liability $l_1$ and $l_2$ are strictly decreasing but the equity $e_1$ and $e_2$ are strictly increasing. This implies that the solution can not be optimal.

Let $\epsilon_1 = (\epsilon_{1,1}, \ldots, \epsilon_{1,m})^T$, where $\epsilon_{1,i} = -1$, $\epsilon_{1,j} = 1$, and $\epsilon_{1,k} = 0$, $\forall k \neq i, j$. We now examine the gradients of the liability and the equity in each period. For the first period, by the expressions of $l_1(y_1)$ and $e_1(y_1)$, we have

\[ \nabla l_1(y_1) = p_0 + (2\Lambda + \Gamma)y_1, \quad \nabla e_1(y_1) = \Gamma x_0 - (2\Lambda - \Gamma)y_1. \tag{A.32} \]

From conditions (i) and (ii) and the fact that $y_1^* \leq 0$ and $y_{1,i}^* < y_{1,i}^* \leq 0$, we deduce

\[
\epsilon_1^T \nabla l_1(y_1^*) = \sum_{k=1}^{m} (2\lambda_{jk} + \gamma_{jk} - 2\lambda_{ik} - \gamma_{ik})y_{1,k}^*
\leq [2(\lambda_{jj} - \lambda_{ij}) + (\gamma_{jj} - \gamma_{ij})]y_{1,j}^* + [2(\lambda_{ji} - \lambda_{ii}) + (\gamma_{ji} - \gamma_{ii})]y_{1,i}^*
\leq [2(\lambda_{jj} - \lambda_{ij}) + (\gamma_{jj} - \gamma_{ij})](y_{1,j}^* - y_{1,i}^*)
< 0.
\]

Moreover, it follows from Equation (A.32) that

\[
\epsilon_1^T \nabla e_1(y_1^*) = \sum_{k=1}^{m} \left[ (-2\lambda_{jk} + \gamma_{jk} + 2\lambda_{ik} - \gamma_{ik})y_{1,k}^* + (\gamma_{jk} - \gamma_{ik})x_{0,k} \right]. \tag{A.33}
\]
Because \(-x_{0,k} \leq y_{1,k}^* \leq 0\) holds for all \(k\) and by condition (ii), we can deduce

\[
(-2\lambda_{jk} + \gamma_{jk} + 2\lambda_{ik} - \gamma_{ik})y_{1,k}^* + (\gamma_{jk} - \gamma_{ik})x_{0,k}
\]

\[
= 2(\lambda_{ik} - \lambda_{jk})y_{1,k}^* + (\gamma_{jk} - \gamma_{ik})x_{0,k}
\]

\[
\geq -(\gamma_{jk} - \gamma_{ik})x_{0,k} + (\gamma_{jk} - \gamma_{ik})x_{0,k}
\]

\[
= 0, \quad \forall k \neq i, j.
\] (A.34)

Thus, by Equation (A.34) and \(x_{0,i} = x_{0,j}\), we have from Equation (A.33) that

\[
\epsilon_1^T \nabla e_1(y_1^*) \geq \left[ 2(\lambda_{ij} - \lambda_{jj}) + \gamma_{jj} - \gamma_{ij} \right] y_{1,j}^* + (\gamma_{jj} - \gamma_{ii})x_{0,j}
\]

\[
+ 2(\lambda_{ii} - \lambda_{ji})y_{1,i}^* + (\gamma_{jj} - \gamma_{ii})x_{0,i}
\]

\[
\geq 2(\lambda_{ij} - \lambda_{jj})y_{1,j}^* + (\gamma_{jj} - \gamma_{ii})x_{0,i}
\]

\[
> 0,
\]

where the second inequality follows from condition (i) and the fact \(-x_{0,j} = -x_{0,j} \leq y_{1,j}^* < y_{1,j}^* \leq 0\), while the last inequality is due to condition (i) and the fact that \(y_{1,j}^* < y_{1,j}^*\).

For the second period, from the expressions of \(l_2(y_1, y_2)\) and \(e_2(y_1, y_2)\), we observe that

\[
\nabla l_2(y_1, y_2) = \begin{bmatrix}
p_0 + (2\Lambda + \Gamma)y_1 + \Gamma y_2 \\
p_0 + \Gamma y_1 + (2\Lambda + \Gamma)y_2
\end{bmatrix},
\]

\[
\nabla e_2(y_1, y_2) = \begin{bmatrix}
x_0 - (2\Lambda - \Gamma)y_1 + \Gamma y_2 \\
x_0 + \Gamma y_1 - (2\Lambda - \Gamma)y_2
\end{bmatrix}.
\]

We now consider the following two cases, and show how we can an \(\epsilon_2\) in each case such that \((\epsilon_1^T, \epsilon_2^T)\nabla l_2(y_1^*, y_2^*) < 0\) and \((\epsilon_1^T, \epsilon_2^T)\nabla e_2(y_1^*, y_2^*) > 0\) in each case.

Case (i): \(y_{1,i}^* + y_{2,i}^* + x_{0,i} > 0\). Take \(\epsilon_2 = 0\). Then,

\[
\begin{bmatrix}
(\epsilon_1^T, \epsilon_2^T) \nabla l_2(y_1^*, y_2^*) = 2\epsilon_1^T \Lambda y_1^* + \epsilon_1^T \Gamma (y_1^* + y_2^*), \\
(\epsilon_1^T, \epsilon_2^T) \nabla e_2(y_1^*, y_2^*) = -2\epsilon_1^T \Lambda y_1^* + \epsilon_1^T \Gamma (x_0 + y_1^* + y_2^*).
\end{bmatrix}
\] (A.35)

Since \(y_{1,k}^* + y_{2,k}^* \leq 0\) and \(x_{0,k} + y_{1,k}^* + y_{2,k}^* \geq 0\) for all \(k\), and by condition (ii), we yield

\[
\epsilon_1^T \Gamma (y_1^* + y_2^*) = \sum_{k=1}^{m} (\gamma_{jk} - \gamma_{ik})(y_{1,k}^* + y_{2,k}^*) \leq 0,
\] (A.36)
\[
\epsilon_1^T \Gamma(x_0 + y_1^* + y_2^*) = \sum_{k=1}^{m} (y_{jk} - y_{ik}) (x_{0,k} + y_{1,k}^* + y_{2,k}^*) \geq 0. 
\]  

(A.37)

Moreover, from conditions (i) and (ii) and the fact that \( y_i^* \leq 0, 0 \geq y_{1,i}^* > y_{1,j}^* \), we obtain

\[
\epsilon_1^T \Lambda y_1^* = \sum_{k=1}^{m} (\lambda_{jk} - \lambda_{ik}) y_{1,k}^* \\
\leq (\lambda_{jj} - \lambda_{ij}) y_{1,j}^* + (\lambda_{ji} - \lambda_{ii}) y_{1,i}^* \\
\leq (\lambda_{jj} - \lambda_{ij}) (y_{1,j}^* - y_{1,i}^*) < 0. 
\]  

(A.38)

Using Equations (A.36)–(A.38), we can deduce from Equation (A.35) that

\[
(\epsilon_1^T, \epsilon_2^T) \nabla l_2 (y_1^*, y_2^*) < 0, \quad (\epsilon_1^T, \epsilon_2^T) \nabla e_2 (y_1^*, y_2^*) > 0. 
\]

In addition, note that \( y_{1,i}^* + y_{2,i}^* + x_{0,i} > 0 \) and \( y_{1,j}^* < 0 \), we take

\[
\hat{\delta} = \min \left\{ y_{1,i}^* + y_{2,i}^* + x_{0,i}, -y_{1,j}^* \right\} > 0. 
\]

Note that \( y_{1,i}^* + y_{2,i}^* + x_{0,i} \leq y_{1,i}^* + x_{0,i} \) due to \( y_{2,i}^* \leq 0 \). It is then easy to check that when \( \delta \in (0, \hat{\delta}) \), we have that \(-x_0 \leq y_{1,i}^* + \delta \epsilon_1 \leq 0, y_{1,i}^* + y_{2,i}^* + \delta \epsilon_1 \geq -x_0 \), and the trading strategy \( (y_{1,i}^* + \delta \epsilon_1, y_{2,i}^*) \) leads to strictly larger equity and smaller liability after trading. Since \( \pi \in (0, 1) \), this means that we have found a feasible direction \( (\epsilon_1, 0) \) from \( (y_{1,i}^*, y_{2,i}^*) \) that still satisfies all the constraints of problem (22) with a strictly larger objective value. This contradicts the optimality of \( (y_1^*, y_2^*) \).

Case (ii): \( y_{1,i}^* + y_{2,i}^* + x_{0,i} = 0 \). Let \( \epsilon_2 = -\epsilon_1 \) and note that

\[
(\epsilon_1^T, \epsilon_2^T) \nabla l_2 (y_1^*, y_2^*) = 2\epsilon_1^T \Lambda (y_1^* - y_2^*), \quad (\epsilon_1^T, \epsilon_2^T) \nabla e_2 (y_1^*, y_2^*) = -2\epsilon_1^T \Lambda (y_1^* - y_2^*). 
\]

We thus need to show that \( \epsilon_1^T \Lambda (y_1^* - y_2^*) < 0 \). We claim that this is always true under the given conditions. Since \( x_{0,i} = x_{0,j} \) and \( y_{1,i}^* + y_{2,j}^* \geq -x_{0,j} \), we obtain \( y_{1,j}^* + y_{2,j}^* \geq y_{1,i}^* + y_{2,j}^* \) and hence \( y_{2,j}^* - y_{2,i}^* \geq y_{1,i}^* - y_{1,j}^* \), which in turn by \( y_{1,i}^* > y_{1,j}^* \) implies \( 0 \geq y_{2,j}^* > y_{2,i}^* \). Since \( y_1^* + y_2^* \geq -x_0 \) and \( y_1^* \leq -\frac{1}{2} x_0 \), we have

\[
y_{1,k}^* - y_{2,k}^* = 2y_{1,k}^* - (y_{1,k}^* + y_{2,k}^*) \leq 2y_{1,k}^* + x_{0,k} \leq 0, \forall k \neq i, j. 
\]

From condition (ii), we then follow

\[
\epsilon_1^T \Lambda (y_1^* - y_2^*) = \sum_{k=1}^{m} (\lambda_{jk} - \lambda_{ik}) (y_{1,k}^* - y_{2,k}^*) \\
\leq (\lambda_{jj} - \lambda_{ij}) (y_{1,j}^* - y_{2,j}^*) + (\lambda_{ji} - \lambda_{ii}) (y_{1,i}^* - y_{2,i}^*) \\
< (\lambda_{jj} - \lambda_{ij}) (y_{1,j}^* - y_{2,j}^*) + (\lambda_{ji} - \lambda_{ii}) (y_{1,j}^* - y_{2,j}^*) \\
= (\lambda_{jj} - \lambda_{ij}) (y_{1,i}^* - y_{2,i}^*) 
\]
\[(\lambda_{jj} - \lambda_{ii})(2y^*_1,i + x_{0,i}) \leq 0,\]

where the second inequality is due to \(\lambda_{jj} > \lambda_{ii}, y^*_1,i > y^*_1,j\) and \(y^*_2,i > y^*_2,j\), the last equality is due to \(y^*_1,i + y^*_2,i + x_{0,i} = 0\), and the last inequality follows from \(\lambda_{jj} > \lambda_{ii}\) and \(y^*_1,i \leq -\frac{1}{2}x_{0,i}\).

In addition, note that \(y^*_2,i < 0\) and \(y^*_1,j < 0\). Note also that \(y^*_1,i + y^*_2,i + x_{0,i} = 0\) and \(y^*_2,i < 0\) imply \(y^*_1,i + x_{0,i} > 0\). We take

\[\delta = \min \left\{ y^*_1,i + x_{0,i}, -y^*_2,i, -y^*_1,j \right\} > 0.\]

Then, for any \(\delta \in (0, \delta)\), we have that \(-x_0 \leq y^*_1 + \delta \epsilon_1 \leq 0, y^*_1 + \delta \epsilon_1 + y^*_2 + \delta \epsilon_2 = y^*_1 + y^*_2 \geq -x_0, y^*_2 + \delta \epsilon_2 \leq 0\) and the trading strategy \((y^*_1 + \delta \epsilon_1, y^*_2 + \delta \epsilon_2)\) leads to strictly larger equity and smaller liability after trading. Since \(\pi \in (0, 1)\), this means that we have found a feasible direction \((\epsilon_1, \epsilon_2)\) from \((y^*_1, y^*_2)\) that still satisfies all the constraints of the problem with a strictly larger objective value, contradicting the optimality of \((y^*_1, y^*_2)\). \[\square\]

**APPENDIX B: THE WAY TO GENERATE MATRICES WITH FIXED NUMBER OF NEGATIVE EIGENVALUES IN EXAMPLE 6.3.**

To control the number of negative eigenvalues of the matrices \(\hat{\Lambda} - \frac{1}{2} \hat{\Gamma}\) and \(\hat{\Lambda} + \frac{1}{2} \hat{\Gamma}\), we generate the matrices \(\Lambda\) and \(\Gamma\) in the following way. First, we randomly generate two matrices \(C, F \in \mathbb{R}^{m \times m}\) whose entries are from \(U[10^{-6}, 10^{-5}]\). Let \(\tilde{C} = \frac{1}{2} (C + C^T)\) and \(\tilde{F} = \frac{1}{2} (F + F^T)\). Then we compute the eigenvalues \(\tilde{\lambda}_1 \leq \ldots \leq \tilde{\lambda}_m\) and \(\tilde{\mu}_1 \leq \ldots \leq \tilde{\mu}_m\) for matrices \(\tilde{C}\) and \(\tilde{F}\), respectively. For arbitrary fixed integers \(q\) and \(s\) with \(1 \leq q, s < m\), let

\[v = \frac{1}{2}(\lambda_q + \lambda_{q+1}), \quad w = \frac{1}{2}(\mu_s + \mu_{s+1}).\]

It is easy to check that matrices \(C - vI\) and \(F - wI\) have exactly \(q\) and \(s\) negative eigenvalues, respectively. Let us define

\[\Lambda = \frac{1}{2}(C - vI + F - wI), \quad \Gamma = C - vI - F + wI.\]

It is then easy to check that \(\hat{\Lambda} - \frac{1}{2} \hat{\Gamma} = \tilde{F} - wI\) and \(\hat{\Lambda} + \frac{1}{2} \hat{\Gamma} = \tilde{C} - vI\), where \(\hat{\Gamma}\) and \(\hat{\Lambda}\) are given in Equation (9).

**APPENDIX C: TABLES IN SECTION 6**

| Table C.1 | The statistics of the linear regression model for each stock in Example 6.1. |
|-----------|---------------------------------------------------------------|
| Stock symbol | GM | KO | PEP | WMT | AAPL | GE |
| \(p\)-value | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| \(R\)-square | 51.10\% | 63.89\% | 56.41\% | 40.09\% | 47.78\% | 48.09\% |
TABLE C.2  Descriptive statistics of the twenty stocks in the sample month considered in Example 6.2.

| Stock symbol | Aver. bid price (in dollar) | Aver. hourly trade vol. (in million) | Aver. trade size (in share) |
|--------------|-----------------------------|--------------------------------------|----------------------------|
| AMD          | $6.8335                     | 0.4651                               | 671.4899                   |
| AXP          | $64.4741                    | 0.0544                               | 101.2464                   |
| BAC          | $14.5604                    | 0.4974                               | 821.4500                   |
| BLK          | $366.0428                   | 0.0126                               | 61.3285                    |
| BRK.B        | $144.2829                   | 0.0307                               | 70.0429                    |
| COST         | $167.1873                   | 0.0421                               | 64.8231                    |
| CVX          | $101.7120                   | 0.1135                               | 93.4919                    |
| DIS          | $96.0200                    | 0.1006                               | 100.6033                   |
| F            | $12.7467                    | 0.2529                               | 440.9061                   |
| FB           | $124.9424                   | 0.6525                               | 126.9192                   |
| JNJ          | $125.0084                   | 0.0844                               | 86.7354                    |
| MCD          | $118.0119                   | 0.1252                               | 94.7140                    |
| MO           | $67.7486                    | 0.0791                               | 102.1315                   |
| MSFT         | $56.5364                    | 0.6957                               | 212.4948                   |
| V            | $78.4346                    | 0.1048                               | 101.3619                   |
| VZ           | $55.4707                    | 0.1976                               | 150.9227                   |
| WFC          | $48.0434                    | 0.2071                               | 175.8796                   |
| WMT          | $72.9371                    | 0.0704                               | 99.5293                    |
| XOM          | $88.1901                    | 0.2000                               | 94.1738                    |
| YHOO         | $38.2431                    | 0.3934                               | 199.8870                   |