J Optim Theory Appl (2007) 135: 563–581
DOI 10.1007/s10957-007-9252-7

An Algorithm for Portfolio Optimization with Variable Transaction Costs, Part 1: Theory

M.J. Best · J. Hlouskova

Published online: 20 July 2007

Abstract A portfolio optimization problem consists of maximizing an expected utility function of n assets. At the end of a typical time period, the portfolio will be modified by buying and selling assets in response to changing conditions. Associated with this buying and selling are variable transaction costs that depend on the size of the transaction. A straightforward way of incorporating these costs can be interpreted as the reduction of portfolios' expected returns by transaction costs if the utility function is the mean-variance or the power utility function. This results in a substantially higher-dimensional problem than the original n-dimensional one, namely $(2K+1)n$-dimensional optimization problem with $(4K+1)n$ additional constraints, where $2K$ is the number of different transaction costs functions. The higher-dimensional problem is computationally expensive to solve. This two-part paper presents a method for solving the $(2K+1)n$-dimensional problem by solving a sequence of n-dimensional optimization problems, which account for the transaction costs implicitly rather than explicitly. The key idea of the new method in Part 1 is to formulate the optimality conditions for the higher-dimensional problem and enforce them by solving a sequence of lower-dimensional problems under the nondegeneracy assumption. In Part 2, we propose a degeneracy resolving rule, address the efficiency of the new method and present the computational results comparing our method with the interior-point optimizer of Mosek.

Communicated by F. Giannessi.

This research was supported by the National Science and Engineering Research Council of Canada and the Austrian National Bank. The authors acknowledge the valuable assistance of Rob Grauer and Associate Editor Franco Giannessi for thoughtful comments and suggestions.

M.J. Best (✉)
Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Waterloo, Ontario, Canada
e-mail: mjbest@uwaterloo.ca

J. Hlouskova
Department of Economics and Finance, Institute for Advanced Studies, Vienna, Austria

Keywords Convex programming · Portfolio optimization · Variable transaction costs

1 Introduction

Investors typically make decisions about how much to buy or sell an asset according to a utility function. The mean-variance utility function [1] and the power utility function [2] are two well-known utility functions. The data defining the utility function, such as the conditional first or second moment of the asset returns, changes with time. In response to the changed data, the investor may wish to adjust his/her portfolio by selling or purchasing various quantities of his/her assets. There is a cost associated with this which is called the transaction cost.

Transaction costs have been studied and modeled in a variety of settings. Reference [3] gives a closed-form solution for the mean-variance utility function with a diagonal covariance matrix, budget constraint and upper bounds on all asset holdings. The transaction costs used in that model were linear for both the buy side and the sell side. In [4], a transaction cost algorithm was developed specifically for a mean-variance utility function having a single linear purchase and sales transaction cost. That method handled transaction costs explicitly within a general quadratic programming algorithm. In [5], an algorithm was developed for the solution of a general utility function with single convex buy-and-sell side transaction costs. The contribution of this paper is to formulate an efficient solution procedure for a problem where each asset's transaction cost is given by piecewise nonlinear convex functions and where the number of pieces is arbitrary. A straightforward way of formulating this as a smooth problem requires many additional variables (the number of pieces times number of original the problem variables). An overview of transaction costs in a variety of settings is given in [6].

Here, we present an algorithm to maximize an expected utility function of n asset holdings while accounting for variable and nonlinear transaction costs. The key feature of the model is that the transaction costs are piecewise convex with K pieces on the buy side and K pieces on the sell side, where K is an arbitrary integer.[1] This enables the investor to model more precisely transaction costs.

As we shall show, these variable transaction costs can be accounted for by solving an optimization problem having $2Kn$ additional variables and $(4K+1)n$ additional linear constraints. However, our new method works by solving a sequence of n-dimensional problems in which the asset's bounds and the linear part of the objective function are varied. The problem could also be formulated in the nonsmooth setting and analyzed in terms of subgradients[2] using the methods presented, for instance, in [8]. However, we have chosen not to do so because the higher-dimensional

[1] In general, K can differ on the buy and sell sides, respectively. We used the same quantity for ease of presentation and notation. The extension of the algorithm to account for different K in the buy and sell sides is straightforward.

[2] In the language of nonsmooth optimization, the presented algorithm could be interpreted as a subgradient method, in the way it moves from one subproblem to a neighboring subproblem. See [7] for a solution method applied to a nondifferentiable and convex problem in an n-dimensional setting.

formulation gives more insight into the underlying structure of the problem where the transaction costs are formulated directly.

Throughout this paper, a prime denotes transposition. All vectors are column vectors unless primed. The ith component of the vector x is denoted by x_i or $(x)_i$.

We consider a portfolio optimization problem of the form

$$\min\{f(x) \mid Ax \le b\}, \tag{1}$$

where A is an (m, n) matrix, b is an m-vector, x is an n-vector of asset holdings, and $-f(x)$ is an expected utility function. The constraint $Ax \le b$ represents general linear constraints on the asset holdings and may include explicit lower and upper bounds. We assume that a target n-vector $\hat{x}$ is given. $\hat{x}$ could represent the current holdings of assets so that purchase transaction costs are measured by the amount increased from $\hat{x}$ and sales transaction costs are measured by a decrease from $\hat{x}$. Alternatively, $\hat{x}$ could represent an index fund being followed, such as the S&P 500.

We incorporate transaction costs into (1) as follows. Let $1 \le i \le n$ and let x_i^{+k} denote the amount purchased according to the kth transaction cost function $p_i^k(x_i^{+k})$. Then, the total amount purchased of asset i is

$$x_i^{+1} + x_i^{+2} + \cdots + x_i^{+K}$$

and the total transaction cost for it is

$$p_i^1(x_i^{+1}) + p_i^2(x_i^{+2}) + \cdots + p_i^K(x_i^{+K}).$$

Each x_i^{+k} is restricted to lie in an interval $[0, e_i^k]$. Letting

$$x^{+k} = (x_1^{+k}, x_2^{+k}, \ldots, x_n^{+k})', \qquad p^k(x^{+k}) = \sum_{i=1}^{n} p_i^k(x_i^{+k}),$$

the total purchase transaction cost is $\sum_{k=1}^{K} p^k(x^{+k})$. In a similar way, for $1 \le i \le n$ define x_i^{-k} to be the amount sold according to the kth transaction cost $q_i^k(x_i^{-k})$. Then, the total amount of asset i to be sold is

$$x_i^{-1} + x_i^{-2} + \cdots + x_i^{-K}$$

and the total transaction cost for it is

$$q_i^1(x_i^{-1}) + q_i^2(x_i^{-2}) + \cdots + q_i^K(x_i^{-K}).$$

Each x_i^{-k} is restricted to lie in an interval $[0, d_i^k]$. Letting

$$x^{-k} = (x_1^{-k}, x_2^{-k}, \ldots, x_n^{-k})', \qquad q^k(x^{-k}) = \sum_{i=1}^{n} q_i^k(x_i^{-k}),$$

the total sales transaction cost is $\sum_{k=1}^{K} q^k(x^{-k})$.

With this notation, the problem to be solved is

$$\min\ f(x)+\sum_{k=1}^{K}p^k(x^{+k})+\sum_{k=1}^{K}q^k(x^{-k}), \tag{2a}$$

$$\text{s.t.}\quad x-\sum_{k=1}^{K}x^{+k}+\sum_{k=1}^{K}x^{-k}=\hat{x},\quad Ax\le b, \tag{2b}$$

$$0\le x^{+k}\le e^k,\quad 0\le x^{-k}\le d^k,\quad k=1,\dots,K, \tag{2c}$$

where e^k and d^k are n-vectors for $k=1,\dots,K$.[3] This way of incorporating transaction costs can be interpreted as the reduction of the portfolio's expected return by the transaction costs if the utility function is the mean-variance or power utility function (see [5, 9] for more detail). Unlike the mean-variance utility function, the power utility function allows us to model the portfolio optimization problem as a discrete-time dynamic investment model based on the reinvestment version of dynamic investment theory (see [10]). From this perspective, our approach of incorporating transaction costs can be implemented also in the multiperiod setting.

We model transaction costs functions as convex rather than concave functions[4] because, in addition to the explicit transaction cost, we account also for the implicit or market impact costs. These latter costs depend on the size of the trade and liquidity restrictions. If the trading volume is too high, the price of a share may rise (fall) between the investment decision and the complete trade execution if the share is to be bought (sold). This will be the case for a large investor whose trading activities affect market prices due to the impact of the trade on the net demand in the market. Throughout this paper, we assume the case of a large investor whose total transaction costs (explicit and implicit market impact costs) can be modeled as above via a convex function (see [11, 12]).

The paper proceeds as follows. In Sect. 2, we derive the optimality conditions in terms of n-dimensional quantities that are sufficient for optimality for the $(2K+1)n$-dimensional problem. Section 3 introduces the algorithm that solves $(2K+1)n$-dimensional convex problem (2) as a sequence of n-dimensional convex programming problems. Sect. 4 concludes.

2 Optimality Conditions

We use (2) as our model problem. The key result of this section is Theorem 2.1 which gives conditions in terms of n-dimensional quantities that are sufficient for optimality for the $(2K+1)n$-dimensional problem (2).

[3] Note that the formulation (2) implicitly imposes the following bounds on x: $\hat{x}-\sum_{j=1}^{K}d^j\le x\le \hat{x}+\sum_{j=1}^{K}e^j$.

[4] Modeling transaction costs as concave functions would be appropriate if the transaction costs are defined as charges for trading only, as in broker commissions, custodial fees, etc.

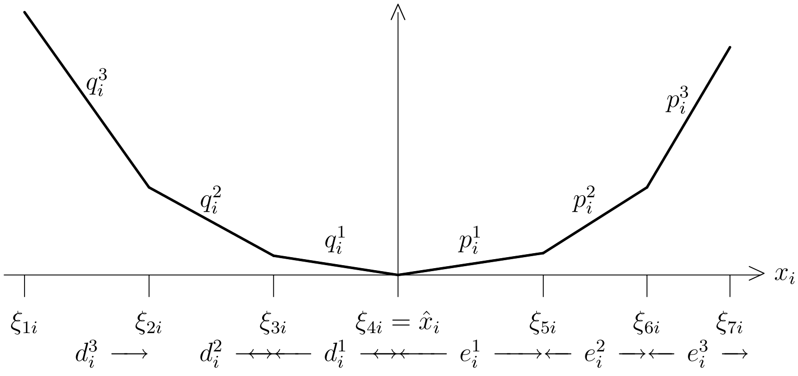

Fig. 1 Slope of transactions cost for asset i

Definition 2.1 *For* $k = 1, \ldots, 2K+1$, *define the vectors of breakpoints* $\xi_k = (\xi_{k1}, \ldots, \xi_{kn})'$ *for* (2) *according to*

$$\xi_{ki} = \begin{cases} \hat{x}_i - \sum_{j=1}^{K-k+1} d_i^j, & k = 1, \ldots, K, \\ \hat{x}_i, & k = K+1, \\ \hat{x}_i + \sum_{j=1}^{k-K-1} e_i^j, & k = K+2, \ldots, 2K+1, \end{cases}$$

with $i = 1, \ldots, n$.

The vector ξ_k provides information concerning where one transaction cost function ends and the next begins ($k = 1, \ldots, K$ for the sales transaction costs, $k = K+1$ for the target indicating the change from selling to buying, and $k = K+2, \ldots, 2K+1$ for the purchase transaction costs). For the case $K = 3$, the bounds d^k, e^k, the special case of linear transaction cost functions $p_i^k(x_i^{+k})$, $q_i^k(x_i^{-k})$, $k = 1, 2, 3$, and the breakpoints $\xi_{1i}, \ldots, \xi_{7i}$ for asset i are shown in Fig. 1. If the purchased amount of asset i does not exceed e_i^1, then the purchase transaction cost is $p_i^1(x_i - \hat{x}_i)$. If the purchased amount of asset i exceeds e_i^1 but is less than $e_i^1 + e_i^2$, then the corresponding transaction cost is $p_i^1 e_i^1 + p_i^2(x_i - \xi_{5i})$. Finally, if the amount purchased is larger than $e_i^1 + e_i^2$ but smaller than $e_i^1 + e_i^2 + e_i^3$ then the associated transaction cost is $p_i^1 e_i^1 + p_i^2 e_i^2 + p_i^3(x_i - \xi_{6i})$. A similar interpretation applies when an asset is sold.

The following assumption will be used to guarantee (among other things) that (2) will be a convex problem and that the solution method that we propose will terminate in a finite number of iterations.

Assumption 2.1 Let x, x^{+k} and $x^{-k} \in \Re^n$, $k = 1, \ldots, K$.

(i) $f(x)$ is a twice differentiable convex function;
(ii) $p^k(x^{+k})$ and $q^k(x^{-k})$ are separable functions; i.e., $p^k(x^{+k}) = \sum_{i=1}^n p_i^k(x_i^{+k})$ and $q^k(x^{-k}) = \sum_{i=1}^n q_i^k(x_i^{-k})$, $k = 1, \ldots, K$;
(iii) $p_i^k(x_i^{+k})$ and $q_i^k(x_i^{-k})$ are convex, twice differentiable functions of a single variable for $i = 1, \ldots, n$, $k = 1, \ldots, K$;

(iv) $\nabla p^1(0) + \nabla q^1(0) \geq 0$;

(v) $\frac{dq_i^k}{dx_i^{-k}}(d_i^k) \leq \frac{dq_i^{k+1}}{dx_i^{-,k+1}}(0)$ and $\frac{dp_i^k}{dx_i^{+k}}(e_i^k) \leq \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(0)$, for $k = 1, \ldots, K - 1$, $i = 1, \ldots, n$.

Remark 2.1 (i) Note that Assumption 2.1(ii) is not unduly restrictive in our context of portfolio optimization as the transaction cost for an asset depends solely on the amount traded of that asset.

(ii) In the context of transaction costs, a natural and stronger assumption instead of Assumption 2.1(iv) is $\nabla p^k(x^{+k}) \geq 0$ and $\nabla q^k(x^{-k}) \geq 0$; i.e., $p_i^k(x_i^{+k})$ and $q_i^k(x_i^{-k})$ are increasing functions for $i = 1, \ldots, n$, $k = 1, \ldots, K$; thus, higher transactions result in higher transaction costs. An additional assumption is $q_i^k(0) = p_i^k(0) = 0$ for $i = 1, \ldots, n$ and $k = 1, \ldots, K$; i.e., no purchases or no sales result in no transaction costs. However, the weaker Assumption 2.1 in addition to the other assumptions is all that is required for our algorithm to work. Furthermore, under Assumption 2.1, our algorithm can be applied also to maximizing a nonsmooth utility function with specified points of nondifferentiability.

For any interval of consecutive breakpoints, we need to know how much of a given component x_i lies in this interval. This gives rise to the following definition.

Definition 2.2 *Let* $x \in \Re^n$.

(i) $x^{+k} \in \Re^n$ *is the positive portion of* x *with respect to* ξ_{K+k}, $k = 1, \ldots, K$, *iff*

$$x_i^{+k} = \begin{cases} 0, & x_i \leq \xi_{K+k,i}, \\ x_i - \xi_{K+k,i}, & \xi_{K+k,i} \leq x_i \leq \xi_{K+k+1,i}, \\ e_i^k, & x_i \geq \xi_{K+k+1,i}. \end{cases}$$

(ii) $x^{-k} \in \Re^n$ *is the negative portion of* x *with respect to* ξ_{K-k+2}, $k = 1, \ldots, K$, *iff*

$$x_i^{-k} = \begin{cases} d_i^k, & x_i \leq \xi_{K-k+1,i}, \\ \xi_{K-k+2,i} - x_i, & \xi_{K-k+1,i} \leq x_i \leq \xi_{K-k+2,i}, \\ 0, & x_i \geq \xi_{K-k+2,i}. \end{cases}$$

(iii) *A feasible breakpoint for* (2) *is a breakpoint which, together with corresponding* x^{+k}, x^{-k}, $k = 1, \ldots, K$, *satisfies the constraints of* (2).

For example, in Fig. 1, if x_i lies between ξ_{5i} and ξ_{6i}, then

$$x_i^{+1} = e_i^1, \quad x_i^{+2} = x_i - \xi_{5i}, \quad x_i^{+3} = 0.$$

Remark 2.2 Let x^{+k} and x^{-k} be the positive and negative portions of x with respect to ξ_{K+k} and ξ_{K-k+2}, $k = 1, \ldots, K$, respectively. Then,

$$(x', (x^{+1})', \ldots, (x^{+K})', (x^{-1})', \ldots, (x^{-K})')'$$

satisfies the constraints

$$0 \le x^{+k} \le e^k, \quad 0 \le x^{-k} \le d^k, \quad k = 1, \dots, K,$$

and

$$x - \sum_{k=1}^{K} x^{+k} + \sum_{k=1}^{K} x^{-k} = \hat{x}.$$

In formulating an algorithm for the solution of (2), we will require the solution of a sequence of n-dimensional subproblems of the form

$$(\mathrm{SUB}(\tilde{d}, \tilde{e})) \quad \min\{f(x) + \tilde{c} \mid Ax \le b,\ \tilde{d} \le x \le \tilde{e}\}.$$

Each such subproblem depends on two n-vectors $\tilde{d}$ and $\tilde{e}$ that will be specified by the algorithm and on a scalar $\tilde{c}$, which is a function of them. The bounds $\tilde{d}$ and $\tilde{e}$ will always satisfy one of the following cases:

(i) $\tilde{d}_i = \xi_{K-k+1,i}$, $\tilde{e}_i = \xi_{K-k+2,i}$ for some k with $k = 1, \dots, K$;
(ii) $\tilde{d}_i = \xi_{K+k,i}$, $\tilde{e}_i = \xi_{K+k+1,i}$ for some k with $k = 1, \dots, K$;
(iii) $\tilde{d}_i = \tilde{e}_i = \xi_{ki}$ for some k with $k = 1, \dots, 2K+1$, with $1 \le i \le n$.

In addition, we require also $\tilde{c} = \sum_{i=1}^{n} \tilde{c}_i$, where

$$\tilde{c}_i = \begin{cases} q_i^k(\xi_{K-k+2,i} - x_i), & \text{case (i)}, \\ p_i^k(x_i - \xi_{K+k,i}), & \text{case (ii)}, \\ 0, & \text{case (iii) when } k = K+1, \end{cases} \tag{3}$$

for $i = 1, \dots, n$ and $k \in \{1, \dots, K\}$.

The subproblem $\mathrm{SUB}(\tilde{d}, \tilde{e})$ has a convex, twice differentiable nonlinear objective function. Furthermore it has only linear constraints. There are many algorithms with known rapid convergence rates which can be used to solve it. One such algorithm is that of [13]. Consecutive subproblems are closely related and an optimal solution for the present subproblem is feasible for the next, so that an active set solution method can be used efficiently to solve them.

Remark 2.3 (a) $\mathrm{SUB}(\tilde{d}, \tilde{e})$ possesses an optimal solution, because the feasible region for $\mathrm{SUB}(\tilde{d}, \tilde{e})$ is compact and, from Assumption 2.1(i), (iii), the objective function for $\mathrm{SUB}(\tilde{d}, \tilde{e})$ is continuous.

(b) $f(x)$ is bounded from below over the feasible region of (2). This is a consequence of (2) having upper and lower bounds on all of the components of x and $f(x)$ being continuous (see (2), the implicit upper and lower bounds on x, the continuity of $f(x)$ and Assumption 2.1(i)).

The linearity of the constraints of (2) and Assumption 2.1(i), (iii) imply that the Karush-Kuhn-Tucker (KKT) conditions for (2) are both necessary and sufficient for optimality (see [14]). The KKT conditions for (2) are

$$-\nabla f(x) = z + A'u, \quad u \ge 0, \quad u'(Ax - b) = 0, \tag{4a}$$

$$-\nabla p^k(x^{+k}) = -z + v_u^k - v_l^k, \quad v_l^k, v_u^k \geq 0, \tag{4b}$$

$$-\nabla q^k(x^{-k}) = z + w_u^k - w_l^k, \quad w_l^k, w_u^k \geq 0, \tag{4c}$$

$$(x^{+k})' v_l^k = 0, \quad (x^{+k} - e^k)' v_u^k = 0, \quad (x^{-k})' w_l^k = 0, \quad (x^{-k} - d^k)' w_u^k = 0, \tag{4d}$$

$$x - \sum_{k=1}^{K} x^{+k} + \sum_{k=1}^{K} x^{-k} = \hat{x}, \quad Ax \leq b, \quad 0 \leq x^{+k} \leq e^k, \quad 0 \leq x^{-k} \leq d^k, \tag{4e}$$

where $k = 1, \ldots, K$ and u, z, v_l^k, v_u^k, w_l^k, w_u^k are the vectors of multipliers for the constraints

$$Ax \leq b, \quad x - \sum_{k=1}^{K} x^{+k} + \sum_{k=1}^{K} x^{-k} = \hat{x},$$
$$x^{+k} \geq 0, \quad x^{+k} \leq e^k, \quad x^{-k} \geq 0, \quad x^{-k} \leq d^k,$$

respectively.

The following theorem shows that the structure of (2) allows the optimality conditions for it to be formulated in a special way.

Theorem 2.1 *Let Assumption 2.1 be satisfied and let x be any point such that $Ax \leq b$ and $\xi_1 \leq x \leq \xi_{2K+1}$. Let x^{+k} and x^{-k}, $k = 1, \ldots, K$ be the positive and negative portions of x with respect to ξ_{K+k}, ξ_{K-k+2}, respectively. Let $u \in \Re^m$ satisfy $u \geq 0$ and $u'(Ax - b) = 0$ and let $z = -\nabla f(x) - A'u$. Then, $X = (x', (x^{+1})', \ldots, (x^{+K})', (x^{-1})', \ldots, (x^{-K})')'$ is optimal for (2) if, for $i = 1, \ldots, n$, the following conditions hold*:

(i) *if $x_i = \xi_{K+k+1,i}$, then*

$$z_i - \frac{dp_i^k}{dx_i^{+k}}(e_i^k) \geq 0, \tag{5a}$$

$$-z_i + \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(0) \geq 0; \tag{5b}$$

(ii) *if $x_i = \hat{x}_i = \xi_{K+1,i}$, then*

$$-z_i + \frac{dp_i^1}{dx_i^{+1}}(0) \geq 0, \tag{6a}$$

$$z_i + \frac{dq_i^1}{dx_i^{-1}}(0) \geq 0; \tag{6b}$$

(iii) *if $x_i = \xi_{K-k+1,i}$, then*

$$-z_i - \frac{dq_i^k}{dx_i^{-k}}(d_i^k) \geq 0, \tag{7a}$$

$$z_i + \frac{dq_i^{k+1}}{dx_i^{-,k+1}}(0) \geq 0; \tag{7b}$$

where $k \in \{1, \ldots, K\}$.

Proof Suppose that X, z, u are as in the statement of the theorem and that cases (i)–(iii) are satisfied. In order to prove that X is optimal for (2), we must show that the KKT conditions for (2), namely (4), are satisfied. The first and last part of the KKT conditions are satisfied by construction. We will show that the remaining parts are satisfied in component form; i.e., we will construct multiplier vectors v_l^k, v_u^k, w_l^k, w_u^k, $k = 1, \ldots, K$, such that

$$-\frac{dp_i^k}{dx_i^{+k}}(x_i^{+k}) = -z_i + (v_u^k)_i - (v_l^k)_i, \quad (v_l^k)_i, (v_u^k)_i \geq 0, \tag{8a}$$

$$-\frac{dq_i^k}{dx_i^{-k}}(x_i^{-k}) = z_i + (w_u^k)_i - (w_l^k)_i, \quad (w_l^k)_i, (w_u^k)_i \geq 0, \tag{8b}$$

$$x_i^{+k}(v_l^k)_i = 0, \quad (x_i^{+k} - e_i^k)(v_u^k)_i = 0, \tag{8c}$$

$$x_i^{-k}(w_l^k)_i = 0, \quad (x_i^{-k} - d_i^k)(w_u^k)_i = 0, \tag{8d}$$

for $i = 1, \ldots, n$. The arguments of the proof depend on which breakpoints x_i lies between, whether x_i lies on a breakpoint, whether x_i lies on ξ_{1i} or $\xi_{2K+1,i}$ and which side of $\hat{x}_i$ (the ith target component) x_i lies on.

Suppose first that $x_i = \xi_{ji}$ for some j with $K + 1 < j < 2K + 1$. By definition of the positive and negative parts of x,

$$x_i^{+k} = e_i^k, \quad \text{for } k = 1, \ldots, j,$$

$$x_i^{+k} = 0, \quad \text{for } k = j + 1, \ldots, K,$$

$$x_i^{-k} = 0, \quad \text{for } k = 1, \ldots, K.$$

Define

$$(v_l^k)_i = 0, \quad k = 1, \ldots, j, \tag{9a}$$

$$(v_l^k)_i = -z_i + \frac{dp_i^k}{dx_i^{+k}}(0), \quad k = j + 1, \ldots, K, \tag{9b}$$

$$(v_u^k)_i = z_i - \frac{dp_i^k}{dx_i^{+k}}(e_i^k), \quad k = 1, \ldots, j, \tag{9c}$$

$$(v_u^k)_i = 0, \quad k = j + 1, \ldots, K, \tag{9d}$$

$$(w_l^k)_i = z_i + \frac{dq_i^k}{dx_i^{-k}}(0), \quad k = 1, \dots, K, \tag{9e}$$

$$(w_u^k)_i = 0, \quad k = 1, \dots, K. \tag{9f}$$

From (9) and case (i) in the statement of the theorem,

$$(v_l^{j+1})_i \geq 0 \quad \text{and} \quad (v_u^j)_i \geq 0. \tag{10}$$

Let k be such that $1 \leq k \leq j-1$. By definition of $(v_u^k)_i$ and Assumption 2.1(v),

$$(v_u^k)_i \geq z_i - \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(0). \tag{11}$$

From Assumption 2.1(i), (iii), $p_i^k(.)$ is a convex, twice differentiable function so that $\frac{dp_i^k}{dx_i^{+k}}(.)$ is an increasing function. Therefore, from (11),

$$(v_u^k)_i \geq z_i - \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(e_i^{k+1}). \tag{12}$$

Repeating the argument which gives (12) from (11) a suitable number of times gives

$$(v_u^k)_i \geq z_i - \frac{dp_i^j}{dx_i^{+j}}(e_i^j) = (v_u^j)_i \geq 0,$$

where the last equality and inequality follow from the definition of $(v_u^j)_i$ in (9) and (10). Thus,

$$(v_u^k)_i \geq 0, \quad \text{for } k = 1, \dots, j. \tag{13}$$

Let k be such that $j+2 \leq k \leq K$. From the definition of $(v_l^k)_i$ in (9), for $k = j+1, \dots, K$,

$$(v_l^k)_i = (v_l^{j+1})_i + \frac{dp_i^k}{dx_i^{+k}}(0) - \frac{dp_i^{j+1}}{dx_i^{+,j+1}}(0). \tag{14}$$

From Assumption 2.1(v) and the fact that $\frac{dp_i^k}{x_i^{+k}}(.)$, $\frac{dp_i^j}{x_i^{+j}}(.)$ are increasing functions, it follows from (14) that

$$\begin{aligned}(v_l^k)_i &\geq (v_l^{j+1})_i + \frac{dp_i^{k-1}}{x_i^{+,k-1}}(e_i^{k-1}) - \frac{dp_i^{j+1}}{dx_i^{+,j+1}}(0) \\ &\geq (v_l^{j+1})_i + \frac{dp_i^{k-1}}{x_i^{+,k-1}}(0) - \frac{dp_i^{j+1}}{dx_i^{+,j+1}}(0)\end{aligned}$$

$$\vdots$$

$$\geq (v_l^{j+1})_i + \frac{dp_i^{j+1}}{x_i^{+,j+1}}(0) - \frac{dp_i^{j+1}}{dx_i^{+,j+1}}(0) = (v_l^{j+1})_i \geq 0,$$

where the last inequality follows from (10). Thus,

$$(v_l^k)_i \geq 0, \quad \text{for } k = j+2, \ldots, K. \tag{15}$$

Finally, note that, from the definitions of $(w_l^k)_i$ and $(v_u^k)_i$ in (9),

$$(w_l^k)_i = (v_u^k)_i + \frac{dp_i^k}{dx_i^{+k}}(e_i^k) + \frac{dq_i^k}{dx_i^{-k}}(0), \quad \text{for } k = 1, \ldots, j$$

and from Assumption 2.1(iv), (v) and (13),

$$(w_l^k)_i \geq 0, \quad \text{for } k = 1, \ldots, j. \tag{16}$$

It follows from Assumption 2.1(iii), (v), the definition of $(w_k^k)_i$ in (9) and (16) that

$$(w_l^k)_i \geq (w_l^j)_i \geq 0, \quad \text{for } k = j+1, \ldots, K. \tag{17}$$

Thus, it follows by construction, namely from (9, 10) and (15–17) that, for $x_i = \xi_{ji}$, $K+1 < j < 2K+1$, the rest of the KKT conditions in (8) are satisfied.

In order to complete the proof of the theorem, it is necessary to consider six other cases for the position of x_i. The total of seven cases are shown in Table 1.

Our analysis above is for Case 3. Cases 1 to 3 apply for buying asset i and these are analogous to Cases 7 to 5, respectively, for selling asset i. The corresponding definitions of the multipliers z, v_l^k, v_u^k, w_l^k, w_u^k, $k = 1, \ldots, K$, are summarized below and the proof for each case is similar to the case considered here:

$$z_i = \begin{cases} \dfrac{dp_i^k}{dx_i^{+k}}(x_i - \xi_{K+k,i}), & x_i \in (\xi_{K+k,i}, \xi_{K+k+1,i}), \\ -(\nabla f(x))_i - (A'u)_i, & x_i \in I^k(x),\ k = 1, \ldots, 2K+1, \\ -\dfrac{dq_i^k}{dx_i^{-k}}(\xi_{K-k+2,i} - x_i), & x_i \in (\xi_{K-k+1,i}, \xi_{K-k+2,i}), \end{cases}$$

Table 1 Possible cases for x_i, $i = 1, \ldots, n$

Case	Position of x_i
1	$x_i = \xi_{2K+1,i}$
2	$\xi_{ji} < x_i < \xi_{j+1,i}, \quad K+1 \leq j \leq 2K$
3	$x_i = \xi_{ji}, \quad K+2 \leq j \leq 2K$
4	$x_i = \xi_{K+1,i}$
5	$x_i = \xi_{ji}, \quad 1 < j \leq K$
6	$\xi_{ji} < x_i < \xi_{j+1,i}, \quad 1 \leq j \leq K$
7	$x_i = \xi_{1i}$

$$(v_l^k)_i = \begin{cases} 0, & x_i \in (\xi_{K+j,i}, \xi_{K+j+1,i}),\ k \le j \text{ or} \\ & i \in I^{K+j+1}(x),\ k \le j, \\ \dfrac{dp_i^k}{dx_i^{+k}}(0) - z_i, & \text{otherwise,} \end{cases}$$

$$(v_u^k)_i = \begin{cases} z_i - \dfrac{dp_i^k}{dx_i^{+k}}(e_i^k), & x_i \in (\xi_{K+j,i}, \xi_{K+j+1,i}),\ k < j,\ i \in I^{K+j+1}(x), \\ 0, & \text{otherwise,} \end{cases}$$

$$(w_l^k)_i = \begin{cases} 0, & i \in I^{K-j+1}(x), \text{ or } x_i \in (\xi_{K-j+1,i}, \xi_{K-j+2,i}),\ k \le j, \\ \dfrac{dq_i^k}{dx_i^{-k}}(0) + z_i, & \text{otherwise,} \end{cases}$$

$$(w_u^k)_i = \begin{cases} -z_i - \dfrac{dq_i^k}{dx_i^{-k}}(d_i^k), & i \in I^{K-j+1}(x),\ k \le j, \\ & x_i \in (\xi_{K-j+1,i}, \xi_{K-j+2,i}),\ k < j, \\ 0, & \text{otherwise,} \end{cases}$$

where $I^k(x)$, $k = 1, \dots, K$, is defined by (23).

It follows then that (9) holds for $i = 1, \dots, n$; consequently, X satisfies the KKT conditions for (2) and is therefore optimal for (2). □

A relationship between an optimal solution for $\text{SUB}(\tilde{d}, \tilde{e})$ and an optimal solution for (2) is established in the following key result.

Lemma 2.1 *Let Assumption 2.1 be satisfied. Let x be an optimal solution for $\text{SUB}(\tilde{d}, \tilde{e})$, let u be the m-vector of multipliers for the constraints $Ax \le b$ of $\text{SUB}(\tilde{d}, \tilde{e})$, and let x^{+k} and x^{-k}, $k = 1, \dots, K$, be the positive and negative portions of x with respect to ξ_{K+k} and ξ_{K-k+2}, respectively. If (5–7) are satisfied, then $(x', (x^{+1})', \dots, (x^{+K})', (x^{-1})', \dots, (x^{-K})')'$ is an optimal solution for (2).*

Proof Let x be an optimal solution for $\text{SUB}(\tilde{d}, \tilde{e})$. It can be seen that x is also optimal for $\text{SUB}(\hat{d}, \hat{e})$ where, for $i = 1, \dots, n$,

$$\hat{d}_i = \begin{cases} \tilde{d}_i, & x_i \in (\xi_{ki}, \xi_{k+1,i}), \quad k = 1, \dots, 2K, \\ \xi_{ki}, & x_i = \xi_{ki}, \quad k = 1, \dots, 2K+1, \end{cases} \tag{18}$$

$$\hat{e}_i = \begin{cases} \tilde{e}_i, & x_i \in (\xi_{ki}, \xi_{k+1,i}), \quad k = 1, \dots, 2K, \\ \xi_{ki}, & x_i = \xi_{ki}, \quad k = 1, \dots, 2K+1. \end{cases} \tag{19}$$

The KKT conditions for $\text{SUB}(\hat{d}, \hat{e})$ can then be written as follows:

$$-(\nabla f(x))_i - \frac{dp_i^k}{dx_i^{+k}}(x_i - \xi_{K+k,i}) = (A'u)_i + (\hat{u}_1)_i - (\hat{u}_2)_i,$$

$$\text{for } \hat{d}_i = \xi_{K+k,i},\ \hat{e}_i = \xi_{K+k+1,i}. \tag{20a}$$

$$-(\nabla f(x))_i = (A'u)_i + (\hat{u}_1)_i - (\hat{u}_2)_i,$$
$$\text{for } \hat{d}_i = \hat{e}_i = \xi_{ji},\ j = 1,\dots,2K+1, \tag{20b}$$

$$-(\nabla f(x))_i + \frac{dq_i^k}{dx_i^{-k}}(\xi_{K-k+2,i} - x_i) = (A'u)_i + (\hat{u}_1)_i - (\hat{u}_2)_i,$$
$$\text{for } \hat{d}_i = \xi_{K-k+1,i},\ \hat{e}_i = \xi_{K-k+2,i}, \tag{20c}$$

$$Ax \le b, \quad \hat{d} \le x \le \hat{e}, \quad u \ge 0, \quad \hat{u}_1 \ge 0, \quad \hat{u}_2 \ge 0, \tag{20d}$$

$$u'(Ax - b) = 0, \quad \hat{u}_1'(\hat{e} - x) = 0, \quad \hat{u}_2'(x - \hat{d}) = 0, \tag{20e}$$

where $k = 1,\dots,K$, u is the m-vector of multipliers for the constraints $Ax \le b$, and $\hat{u}_1$, $\hat{u}_2$ are the n-vectors of multipliers for the constraints $x \le \hat{e}$ and $x \ge \hat{d}$, respectively.

We define the n-vector z as follows

$$z_i = \begin{cases} \dfrac{dp_i^k}{dx_i^{+k}}(x_i - \xi_{K+k,i}), & x_i \in (\xi_{K+k,i}, \xi_{K+k+1,i}), \quad k = 1,\dots,K, \\ (\hat{u}_1)_i - (\hat{u}_2)_i, & x_i = \xi_{ki}, \quad k = 1,\dots,2K+1, \\ -\dfrac{dq_i^k}{dx_i^{-k}}(\xi_{K-k+2,i} - x_i), & x_i \in (\xi_{K-k+1,i}, \xi_{K-k+2,i}), \quad k = 1,\dots,K. \end{cases} \tag{21}$$

The complementarity slackness parts of the KKT conditions (20) and (21) imply that the first three lines of the KKT conditions (20) for SUB($\hat{d}, \hat{e}$) can now be written as

$$-(\nabla f(x))_i = z_i + (A'u)_i, \quad i = 1,\dots,n.$$

This coincides with the first line of the KKT conditions for (4) where u is the nonnegative multiplier for $Ax \le b$ in SUB($\hat{d}, \hat{e}$).

If x^{+k} and x^{-k} are the positive and negative portions of x with respect to ξ_{K+k} and ξ_{K-k+2}, $k = 1,\dots,K$, and (5–7) are satisfied then it follows from Theorem 2.1 that $(x', (x^{+1})',\dots,(x^{+K})', (x^{-1})',\dots,(x^{-K})')'$ is an optimal solution for (2). □

3 Detailed Formulation of the Algorithm

Our solution method for (2) proceeds by solving a sequence of n-dimensional subproblems SUB($\tilde{d}, \tilde{e}$). The variables x^{+k} and x^{-k}, $k = 1,\dots,K$, the constraint

$$x - \sum_{k=1}^{K} x^{+k} + \sum_{k=1}^{K} x^{-k} = \hat{x},$$

as well as the bounds

$$0 \le x^{+k} \le e^k, \quad 0 \le x^{-k} \le d^k, \quad k = 1,\dots,K,$$

are accounted for implicitly rather than explicitly. At any iteration j, each variable x_i is forced to lie in the interval

$$\text{(i)} \quad \xi_{K-k+1,i} \le x_i \le \xi_{K-k+2,i}, \quad k = 1, \dots, K,$$

or

$$\text{(ii)} \quad \xi_{K+k,i} \le x_i \le \xi_{K+k+1,i}, \quad k = 1, \dots, K,$$

or is forced to lie at a single point

$$\text{(iii)} \quad x_i = \xi_{ki}, \quad \text{for all } k = 1, \dots, 2K+1.$$

The objective function for $\mathrm{SUB}(\tilde{d}, \tilde{e})$ is formed from that of (1) with the addition of transaction costs according to the applicable case (i)–(iii) above as follows. In case (i), we add the additional transaction cost for selling the amount $\xi_{K-k+2,i} - x_i$ of asset i, namely $q_i^k(\xi_{K-k+2,i} - x_i)$. In case (ii), we add the additional transaction cost for buying the amount $x_i - \xi_{K+k,i}$ of asset i, namely $p_i^k(x_i - \xi_{K+k,i})$. In case (iii), we add $q_i^k(d_i^k)$ if $k \in \{1, \dots, K\}$ or zero if $k = K+1$ or $p_i^{k-K-1}(e_i^{k-K-1})$ if $k \in \{K+2, \dots, 2K+1\}$.

Solving $\mathrm{SUB}(\tilde{d}, \tilde{e})$ at the jth iteration produces an optimal solution x^{j+1} and multipliers u^{j+1} for the constraints $Ax \le b$. In addition, we set

$$z_i = -(\nabla f(x^{j+1}))_i - (A' u^{j+1})_i$$

for i such that $x_i^{j+1} = \xi_{ki}$, $k \in \{1, \dots, 2K+1\}$ and, for each such i, calculate the two multipliers as follows:

(i) if $x_i = \xi_{K+k+1,i}$ then calculate

$$(v_u^k)_i = z_i - \frac{dp_i^k}{dx_i^{+k}}(e_i^k) \quad \text{and} \quad (v_l^{k+1})_i = -z_i + \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(0),$$

where $(v_u^k)_i$ is the multiplier for $x_i^{+k} \le e_i^k$ and $(v_l^{k+1})_i$ is the multiplier for $x_i^{+,k+1} \ge 0$;

(ii) if $x_i = \xi_{K-k+1,i}$ then calculate

$$(w_u^k)_i = -z_i - \frac{dq_i^k}{dx_i^{-k}}(d_i^k) \quad \text{and} \quad (w_l^{k+1})_i = z_i + \frac{dq_i^{k+1}}{dx_i^{-,k+1}}(0),$$

where $(w_u^k)_i$ is the multiplier for $x_i^{-k} \le d_i^k$ and $(w_l^{k+1})_i$ is the multiplier for $x_i^{-,k+1} \ge 0$;

(iii) if $x_i = \xi_{K+1,i}$ then calculate

$$(v_l^1)_i = -z_i + \frac{dp_i^1}{dx_i^{+1}}(0) \quad \text{and} \quad (w_l^1)_i = z_i + \frac{dq_i^1}{dx_i^{-1}}(0),$$

where $(v_l^1)_i$ is the multiplier for $x_i^{+1} \ge 0$ and $(w_l^1)_i$ is the multiplier for $x_i^{-1} \ge 0$.

If all such multipliers are nonnegative, then from Lemma 2.1, $((x^{j+1})', (x^{+1})', \ldots, (x^{+K})', (x^{-1})', \ldots, (x^{-K})')'$ is optimal for (2), where x^{+k} and x^{-k}, $k = 1, \ldots, K$ are the positive and negative portions of x^{j+1} with respect to ξ_{K+k} and ξ_{K-k+2}, respectively.

If at least one of these multipliers is strictly negative, we can drop the associated active inequality constraint. In active set methods, it is traditional to choose the smallest such multiplier and we do so here. Suppose that the multiplier $(v_l^{k+1})_i$ is the smallest over all the multipliers in cases (i)–(iii). In the next iteration, the upper bound on x_i is set to $\xi_{K+k+2,i}$ and the lower bound on it is set to $\xi_{K+k+1,i}$. If $(v_u^k)_i$ is the smallest multiplier, then for the next iteration, the lower bound on x_i is changed to $\xi_{K+k,i}$ and the upper bound is changed to $\xi_{K+k+1,i}$.

Suppose next that the multiplier $(v_l^1)_i$ is the smallest over all the multipliers in cases (i)–(iii). Then, in the next iteration, the upper bound on x_i is set to $\xi_{K+2,i}$ and the lower bound on it is set to $\xi_{K+1,i}$. If $(w_l^1)_i$ is the smallest multiplier, then for the next iteration, the lower bound on x_i is changed to ξ_{Ki} and the upper bound is changed to $\xi_{K+1,i}$.

Finally, if the multiplier $(w_l^{k+1})_i$ is the smallest over all the multipliers in cases (i)–(iii), then in the next iteration, the upper bound on x_i is set to $\xi_{K-k+1,i}$ and the lower bound on it is set to $\xi_{K-k,i}$. If $(w_u^k)_i$ is the smallest multiplier, then for the next iteration, the lower bound on x_i is changed to $\xi_{K-k+1,i}$ and the upper bound is changed to $\xi_{K-k+2,i}$.

We now give a concise formulation of the algorithm for the solution of (2) under Assumption 2.1 and $S \neq \emptyset$ where

$$S \equiv \{x \in \Re^n \mid Ax \leq b,\ \xi_1 \leq x \leq \xi_{2K+1}\} \tag{22}$$

and the index sets of x are defined as

$$I^k(x) = \{i \mid x_i = \xi_{ki},\ k = 1, \ldots, 2K+1\}. \tag{23}$$

Algorithm

Step 0. Initialization. Start with any $x^0 \in S$. Construct the initial bounds $\tilde{d}^0$, $\tilde{e}^0$ as follows. For each $i = 1, \ldots, n$, do the following:

Step 0a. If $\xi_{K+k,i} < x_i^0 \leq \xi_{K+k+1,i}$, define

$$\tilde{d}_i^0 = \xi_{K+k,i}, \quad \tilde{e}_i^0 = \xi_{K+k+1,i}, \quad k = 1, \ldots, K.$$

Step 0b. If $\xi_{K-k+1,i} \leq x_i^0 < \xi_{K-k+2,i}$, define

$$\tilde{d}_i^0 = \xi_{K-k+1,i}, \quad \tilde{e}_i^0 = \xi_{K-k+2,i}, \quad k = 1, \ldots, K.$$

Step 0c. Otherwise, define $\tilde{d}_i^0 = \tilde{e}_i^0 = \hat{x}_i$.

Step 0d. Then, define $j = 0$ and go to Step 1.

Step 1. Solution of the Subproblem. Solve $\mathrm{SUB}(\tilde{d}^j, \tilde{e}^j)$ to obtain an optimal solution x^{j+1} and the multiplier vector u^{j+1} for the constraints $Ax \le b$. Go to Step 2.

Step 2. Update and Optimality Test. Compute

$$z_i = -(\nabla f(x^{j+1}))_i - (A'u^{j+1})_i, \quad i \in I^k(x^{j+1}),\ \text{for } 1 \le k \le 2K+1,$$

and for each such i perform one of the following cases where $1 \le k \le K$.

Step 2a. If $x_i^{j+1} = \xi_{K+k+1,i}$, then calculate

$$(v_l^{k+1})_i = -z_i + \frac{dp_i^{k+1}}{dx_i^{+,k+1}}(0), \qquad (v_u^k)_i = z_i - \frac{dp_i^k}{dx_i^{+k}}(e_i^k).$$

Step 2b. If $x_i^{j+1} = \xi_{K+1,i}$, then calculate

$$(v_l^1)_i = -z_i + \frac{dp_i^1}{dx_i^{+1}}(0), \qquad (w_l^1)_i = z_i + \frac{dq_i^1}{dx_i^{-1}}(0).$$

Step 2c. If $x_i^{j+1} = \xi_{K-k+1,i}$, then calculate

$$(w_l^{k+1})_i = z_i + \frac{dq_i^{k+1}}{dx_i^{-,k+1}}(0), \qquad (w_u^k)_i = -z_i - \frac{dq_i^k}{dx_i^{-k}}(d_i^k).$$

Furthermore, compute m_1, m_2, k^* such that

$$y_i^k = \begin{cases} (v_l^{k+1})_i, & \text{if } x_i^{j+1} = \xi_{K+k+1,i}, \\ (v_l^1)_i, & \text{if } x_i^{j+1} = \xi_{K+1,i}, \\ (w_u^k)_i, & \text{if } x_i^{j+1} = \xi_{K-k+1,i}, \end{cases}$$

$$\hat{y}_i^k = \begin{cases} (v_u^k)_i, & \text{if } x_i^{j+1} = \xi_{K+k+1,i}, \\ (w_l^1)_i, & \text{if } x_i^{j+1} = \xi_{K+1,i}, \\ (w_l^{k+1})_i, & \text{if } x_i^{j+1} = \xi_{K-k+1,i}, \end{cases}$$

$$y_{m_1}^{k^*} \equiv \min\{y_i^k\}, \qquad y_{m_2}^{k^*} \equiv \min\{\hat{y}_i^k\},$$

where $m_1, m_2 \in \{1, \dots, n\}$ and $k^* \in \{1, \dots, 2K+1\}$.

Step 2d. If $y_{m_1}^{k^*} \ge 0$ and $y_{m_2}^{k^*} \ge 0$, or if no element of x^{j+1} coincides with a breakpoint, then stop with an optimal solution $((x^{j+1})', (x^{+1})', \dots, (x^{+K})', (x^{-1})', \dots, (x^{-K})')'$ for (2), where x^{+k} and x^{-k} are positive and negative portions of x^{j+1} with respect to ξ_{K+k} and ξ_{K-k+2}, respectively, $k = 1, \dots, K$.

Step 2e. If $y_{m_1}^{k^*} < y_{m_2}^{k^*}$, then define

$$\tilde{d}_i^{j+1} = \begin{cases} \tilde{d}_i^j, & i \in \{1, \dots, n\} - I^k(x^{j+1}), \\ \xi_{ki}, & i \in I^k(x^{j+1}), \end{cases}$$

$$\tilde{e}_i^{j+1} = \begin{cases} \tilde{e}_i^j, & i \in \{1, \dots, n\} - I^k(x^{j+1}), \\ \xi_{ki}, & i \in I^k(x^{j+1}) - \{m_1\}, \\ \xi_{k^*+1,m_1}, & i = m_1, \end{cases}$$

$k = 1, \dots, 2K + 1$. Replace j with $j + 1$ and go to Step 1.

Step 2f. If $y_{m_1}^{k^*} \geq y_{m_2}^{k^*}$, then define

$$\tilde{d}_i^{j+1} = \begin{cases} \tilde{d}_i^j, & i \in \{1, \dots, n\} - I^k(x^{j+1}), \\ \xi_{ki}, & i \in I^k(x^{j+1}) - \{m_2\}, \\ \xi_{k^*-1,m_2}, & i = m_2, \end{cases}$$

$$\tilde{e}_i^{j+1} = \begin{cases} \tilde{e}_i^j, & i \in \{1, \dots, n\} - I^k(x^{j+1}), \\ \xi_{ki}, & i \in I^k(x^{j+1}), \end{cases}$$

$k = 1, \dots, 2K + 1$. Replace j with $j + 1$ and go to Step 1.

We assume implicitly in Step 2 that each breakpoint has two multipliers associated with it. However, if $x_i^{j+1} = \xi_{1i}$ then only $(w_u^K)_i$ is calculated in Step 2c. On the other hand, if $x_i^{j+1} = \xi_{2K+1,i}$ then only $(v_u^K)_i$ is calculated in Step 2a.

The case of (2) being unbounded from below is excluded in Remark 2.3 and so does not need to be treated in the algorithm. Note also that, if $S \neq \emptyset$, then the feasible region of any subproblem solved by the algorithm is nonempty.

In order to ensure finite termination of the algorithm, we need a nondegeneracy assumption. Degenerate and nondegenerate points for (2) are defined as follows.

Definition 3.1 A point $X = (x', (x^{+1})', \dots, (x^{+K})', (x^{-1})', \dots, (x^{-K})')'$ is nondegenerate for (2) iff it is feasible for (2) and the gradients of those constraints active at X are linearly independent. X is degenerate otherwise.

We establish the finite termination property of the algorithm in the following theorem.

Theorem 3.1 *Let Assumption 2.1 be satisfied and let $S \neq \emptyset$, where S is given by (22). Beginning with any $x^0 \in S$, let the algorithm be applied to (2) and let $x^1, x^2, \dots, x^j, \dots$ be the points so obtained. Let $(x^{+k})^j$ and $(x^{-k})^j$ be the positive and negative portions of x^j with respect to ξ_{K+k} and ξ_{K-k+2}, respectively, for $k = 1, \dots, K$. Assume each $X^j = ((x^j)', ((x^{+1})^j)', \dots, ((x^{+K})^j)', ((x^{-1})^j)', \dots, ((x^{-K})^j)')'$ is nondegenerate. Then $G(X^j) < G(X^{j+1})$, for $j = 1, 2, \dots,$ where $G(X)$ is the objective function of (2). In addition, the algorithm will solve (2) in a* *finite number of steps*; *i.e.*, *there exist an iteration* j, *such that* X^j *is an optimal solution for* (2).

Proof First observe that, from Assumption 2.1, the hypothesis that $S \neq \emptyset$, and the fact that an optimal solution of $\mathrm{SUB}(\tilde{d}^{j-1}, \tilde{e}^{j-1})$ is feasible for $\mathrm{SUB}(\tilde{d}^j, \tilde{e}^j)$ imply that there exists an optimal solution for $\mathrm{SUB}(\tilde{d}^j, \tilde{e}^j)$ in Step 1 of the algorithm.

Suppose that $j-1$ iterations of the algorithm are performed, let x^j be optimal for $\mathrm{SUB}(\tilde{d}^{j-1}, \tilde{e}^{j-1})$, and let x^{+k}, x^{-k}, $k = 1, \dots, K$, be positive and negative portions of x^j with respect to ξ_{K+k} and ξ_{K-k+2}, respectively. Let

$$X^j = ((x^j)', (x^{+1})', \dots, (x^{+K})', (x^{-1})', \dots, (x^{-K})')'$$

and $y_{m_1}^{k^*}$ and $y_{m_2}^{k^*}$ be defined as in Step 2. If $y_{m_1}^{k^*} \geq 0$ and $y_{m_2}^{k^*} \geq 0$ or if no element of x^j coincides with a breakpoint, then from Lemma 2.1, X^j is an optimal solution for (2).

If $y_{m_1}^{k^*} \leq y_{m_2}^{k^*}$ and $y_{m_1}^{k^*} < 0$, then from Step 2 of the algorithm the constraints for the jth subproblem are obtained from those of subproblem $j-1$ by replacing the constraint $x_{m_1} = \xi_{k^* m_1}$ with $\xi_{k^* m_1} \leq x_{m_1} \leq \xi_{k^*+1, m_1}$ as well as imposing $x_i = \xi_{ki}$ for all $i \in I^k(x^j) - \{m_1\}$, $k = 1, \dots, 2K+1$. From Lemma 6.1 in the appendix of [5], it follows that

$$G(X^{j+1}) < G(X^j). \tag{24}$$

The case for $y_{m_2}^{k^*} \leq y_{m_1}^{k^*}$ and $y_{m_2}^{k^*} < 0$ is similar and also leads to (24). Therefore, (24) is satisfied for all iterations.

In each subproblem, each variable x_i, $i = 1, \dots, n$, is restricted according to $\xi_{ki} \leq x_i \leq \xi_{k+1,i}$ or $x_i = \xi_{ki}$, $k = 1, \dots, 2K+1$. Thus, there are only finitely many subproblems and, from (24), none can be repeated. Thus, the algorithm terminates in a finite number of steps and, by Theorem 2.1, it will terminate with an optimal solution for (2). This concludes the proof of the theorem. □

4 Conclusions

We have addressed the problem of maximizing an expected utility function of n assets, such as the mean-variance or power utility function. Associated with a change in an asset's holdings from its current or target value is a transaction cost. That must be accounted for in practical problems. To more accurately formulate the transaction costs, we used a model with K purchase and K sales transaction costs functions which depend on the size of the transaction. A straightforward way of accounting for these costs results in a $(2K+1)n$-dimensional optimization problem with $(4K+1)n$ additional constraints. This higher dimensional problem is computationally expensive to solve. We formulated and proved a theorem which gave sufficient conditions for optimality for the $(2K+1)n$-dimensional problem in terms of n-dimensional quantities. Based on this theorem, we presented a method for solving the $(2K+1)n$-dimensional problem by solving a sequence of n-dimensional optimization problems, which account for the transaction costs implicitly rather than explicitly. We proved finite termination of our method with a nondegeneracy assumption.

References

1. Markowitz, H.M.: Portfolio Selection: Efficient Diversification of Investment. Wiley/Yale University Press, New Haven (1959)
2. Grauer, R.R., Hakansson, N.H.: On the use of mean-variance and quadratic approximations in implementing dynamic investment strategies: A comparison of returns and investment policies. Manag. Sci. **38**, 856–870 (1992)
3. Best, M.J., Hlouskova, J.: Portfolio selection and transaction costs. Comput. Optim. Appl. **24**, 95–116 (2003)
4. Best, M.J., Hlouskova, J.: Quadratic programming with transaction costs. Comput. Oper. Res. **35**, 18–33 (2008)
5. Best, M.J., Hlouskova, J.: An algorithm for portfolio optimization with transaction costs. Manag. Sci. **51**, 1676–1688 (2005)
6. Schattman, J.B.: Portfolio selection under nonconvex transaction costs and capital gains taxes. Ph.D. thesis, Rutgers Center for Operations Research, Rutgers University (2000)
7. Best, M.J., Potaptchik, M.: Portfolio selection using nonsmooth convex transaction costs. Combinatorics and Optimization Research Report 2004-30, Faculty of Mathematics, University of Waterloo (2004)
8. Hiriart-Urruty, J.-B., Lemarechal, C.: Convex Analysis and Minimization Algorithms, Part 1: Fundamentals. Springer, New York (1996)
9. Best, M.J., Kale, J.K.: Quadratic programming for large–scale portfolio optimization. In: Keyes, J. (ed.) Financial Services Information Systems, pp. 513–529. CRC Press, Boca Raton (2000)
10. Grauer, R.R., Hakansson, N.H.: A half-century of returns on levered and unlevered portfolios of stocks, bonds and bills, with and without small stocks. J. Bus. **59**, 287–318 (1986)
11. Jonsson, M., Keppo, J., Meng, X.: Option pricing with an exponential effect function. Working paper available at the Social Science Research Network (2004)
12. Kalin, D., Zagst, R.: Portfolio optimization under liquidity costs. Working Paper 04-01, Risklab Germany GmbH, Germany (2004)
13. Best, M.J., Ritter, K.: A class of accelerated conjugate direction methods for linearly constrained minimization problems. Math. Comput. **30**, 478–504 (1976)
14. Mangasarian, O.L.: Nonlinear Programming. McGraw-Hill, New York (1969)