Journal of Global Optimization **22:** 137–154, 2002.

Portfolio optimization under D.C. transaction costs and minimal transaction unit constraints

HIROSHI KONNO[1] and ANNISTA WIJAYANAYAKE[2]
[1]*Department of Industrial Engineering and Management and Center for Research in Advanced Financial Technologies, Tokyo Institute of Technology, Tokyo, Japan (E-mail: konno@indsys.chuo-u.ac.jp);* [2]*Department of Industrial Engineering and Management, Tokyo Institute of Technology, Tokyo, Japan*

Abstract. This paper addresses itself to a portfolio optimization problem under nonconvex transaction costs and minimal transaction unit constraints. Associated with portfolio construction is a fee for purchasing assets. Unit transaction fee is larger when the amount of transaction is smaller. Hence the transaction cost is usually a concave function up to certain point. When the amount of transaction increases, the unit price of assets increases due to illiquidity/market impact effects. Hence the transaction cost becomes convex beyond certain bound. Therefore, the net expected return becomes a general d.c. function (difference of two convex functions). We will propose a branch-and-bound algorithm for the resulting d.c. maximization problem subject to a constraint on the level of risk measured in terms of the absolute deviation of the rate of return of a portfolio. Also, we will show that the minimal transaction unit constraints can be incorporated without excessively increasing the amount of computation.

Key words: Portfolio optimization, D.c. programming, Nonconvex transaction cost, Minimal transaction unit constraint, Mean-absolute deviation model

1. Introduction

In a recent series of papers [8, 9], authors proposed practical algorithms for solving portfolio optimization problems under concave transaction costs and minimal transaction unit constraints.

Associated with purchasing an asset is a transaction fee. The unit fee is larger when the amount of transaction is smaller and it decreases as the amount increases. Therefore, the transaction fee can be represented as a piecewise linear increasing concave function, so that maximization of the net return becomes a convex maximization problem.

A traditional and well known approach to these problems is the use of integer programming methods by introducing a number of 0–1 variables to approximate the convex cost function. However, computation time often explodes as the number of assets increases [5]. In fact, it is very difficult to solve this problem if the number of assets is more than, say 50. As a result, exact treatment of transaction costs has long been set aside and standard mean variance model or its variations [2, 11] were used to construct a portfolio. Among few exceptions are the works by Perold [14] and Mulvey [12], in which a convex approximation of nonconvex cost function is employed.

Fortunately, however a remarkable progress in global (nonconvex) optimization in the last decade enabled one to solve a class of nonconvex minimization problems in an efficient manner (see [6, 16] for recent progress in this area). In particular, branch-and-bound methods based on the idea of Falk and Soland [3] turned out to be an efficient method for minimizing a concave objective function under linear constraints [4, 5, 15]. Encouraged by these new developments, the authors proposed a branch-and-bound algorithm and succeeded in solving a practical portfolio optimization problem with concave transaction costs [8]. The keys to the success of this approach are the use of:

(i) a mean-absolute deviation model for formulating the portfolio optimization model,
(ii) an elaborate problem reduction technique,
(iii) an ω-subdivision strategy for partitioning the feasible region.

Later in [9], we extended this approach to a more difficult problem with minimal transaction unit constraints on each asset, usually 1000 stocks in the Tokyo Stock Exchange. The problem can again be formulated as a linear programming problem with integer variables which require a huge amount of computation time. However, the problem reduction worked extremely well for this problem. In fact, computational results reported in [9] show that we can obtain a close to optimal portfolio where the assets with fractional units are very few.

In this paper, we will extend these results to a more difficult problem with market illiquidity effects. If there is not enough supply to meet the demand, the unit price will increase. Also, if an investor purchases a significant amount of assets, the unit price will increase. This is called ‘illiquidity’ or ‘market impact’ effect. Therefore the transaction cost becomes convex when the amount exceeds some point. As a result, the total transaction cost is concave (due to transaction fee) when the amount is smaller but it becomes convex (due to illiquidity/market impact) beyond certain point. The problem thus becomes a d.c. optimization problem.

In the next section, we will briefly discuss the mean-absolute deviation model and the structure of cost function. Section 3 will be devoted to the description of branch-and-bound algorithm. In Section 4, we will present the results of numerical simulation using market data. Finally, in Section 5, we will state the future direction of research.

2. Mean-absolute deviation model under transaction costs

In 1991, one of the authors proposed a mean-absolute deviation (MAD) portfolio model [7] to formulate a very large scale portfolio optimization problem. This serves as an alternative to the standard mean-variance (MV) model, where variance of the rate of return of a portfolio is adopted as the measure of risk.

As demonstrated by Ogryczak and Ruszczynski [13], the absolute deviation is an authentic measure of risk in view of its compatibility with von Neumann's principle of 'expected utility maximization'. Also, since the model can be casted into a linear programming problem, it can be solved much faster than mean-variance models. Also, linear programming formulation has computational advantages over quadratic programming formulation when we treat integer constraints and nonconvex cost function to be discussed below.

Let R_j be the rate of return of jth asset $(j = 1, \cdots, n)$ and let $\boldsymbol{x}=(x_1, \cdots, x_n)$ be a portfolio, a vector of proportion of investments into each asset. Let X be an investable set, i.e, a set of feasible portfolios. We will assume for simplicity that X is a set defined below:

$$X = \{\boldsymbol{x} = (x_1, \cdots, x_n) \mid \sum_{j=1}^{n} x_j = 1, \ \ 0 \leqslant x_j \leqslant \alpha_j, \ \ j = 1, \cdots, n\}. \tag{1}$$

The rate of return $R(\boldsymbol{x})$ of the portfolio $\boldsymbol{x}$ is given by

$$R(\boldsymbol{x}) = \sum_{j=1}^{n} R_j x_j \,. \tag{2}$$

Let r_j be the expected value of R_j. The absolute deviation $W(\boldsymbol{x})$ of the rate of return $R(\boldsymbol{x})$ of the portfolio $\boldsymbol{x}$ is given by

$$W(\boldsymbol{x}) = E[\,|\,R(\boldsymbol{x}) - E[R(\boldsymbol{x})]\,|\,]\,. \tag{3}$$

Let us assume that $\mathbf{R} = (R_1, \cdots, R_n)$ is distributed over a finite set of points $\{(r_{1t}, \cdots, r_{nt}), \ t = 1, \cdots, T\}$ and that the probability of occurrence of $(r_{1t}, \cdots, x_{nt})$ is given by $p_t, \ t = 1, \cdots, T$. Then

$$r_j = \sum_{t=1}^{T} p_t r_{jt}$$

and

$$W(\boldsymbol{x}) = \sum_{t=1}^{T} p_t \,|\, \sum_{j=1}^{n} (r_{jt} - r_j) x_j \,|\,. \tag{4}$$

The mean-absolute deviation (MAD) portfolio optimization model is defined as follows :

$$\left|\begin{array}{ll} \text{minimize} & W(\boldsymbol{x}) \equiv \displaystyle\sum_{t=1}^{T} p_t \,|\, \sum_{j=1}^{n} (r_{jt} - r_j) x_j \,| \\ \text{subject to} & \displaystyle\sum_{j=1}^{n} r_j x_j = \rho, \\ & \boldsymbol{x} \in X, \end{array}\right. \tag{5}$$

where ρ is a given constant representing the expected rate of return of the portfolio. The MAD model can be formulated in an alternative way :

$$\left| \begin{array}{ll} \text{maximize} & \sum_{j=1}^{n} r_j x_j \\ \text{subject to} & \sum_{t=1}^{T} p_t \left| \sum_{j=1}^{n} (r_{jt} - r_j) x_j \right| \leqslant w, \\ & \boldsymbol{x} \in X, \end{array} \right. \tag{6}$$

where w is a given constant representing the tolerable level of risk. Both (5) and (6) can be used interchangeably to generate an efficient frontier.

Let us now consider transaction cost $c(\boldsymbol{x})$ associated with purchasing a portfolio $\boldsymbol{x}$, which consists of two components :

(i) transaction fee,

(ii) illiquidity/market impact cost.

Transaction fee is usually determined by transaction fee table provided by each agent. When the amount of purchase is small, the unit transaction cost is larger and it decreases as the amount increases. Therefore, the transaction fee $c_j(x_j)$ associated with purchasing x_j units of jth asset is an increasing piecewise linear concave function.

When x_j reaches some bound, 'illiquidity' cost may be incurred. Since the purchase is associated with a sale of some other investors, the unit purchasing price will increase if there is not sufficient sale. Therefore, the expected rate of return r_j of the jth asset becomes a function of x_j. Therefore, the total rate of return $r(\boldsymbol{x})$ of the portfolio $\boldsymbol{x}$ can be represented as follows :

$$r(\boldsymbol{x}) = \sum_{j=1}^{n} \left\{ r_j x_j - c_j(x_j) \right\}, \tag{7}$$

where $c_j(x_j)$ is a d.c. function described in Figure 1.

Another difficulty associated with the practical portfolio optimization is the existence of minimal transaction unit constraints, usually 1000 stocks in the Tokyo Stock Exchange. Therefore x_j has to satisfy a constraint $x_j \in U_j \equiv \{0, x_{j1}, \cdots, x_{jn}\}$ where x_{jn} is the largest investable unit below the given upper bound α_j. This constraint is negligible if the total amount of investment is large, since rounding to the nearest feasible point will have only a small effect on the overall structure of the portfolio. However, a rounding procedure can have a significant effect if the amount of fund is relatively small.

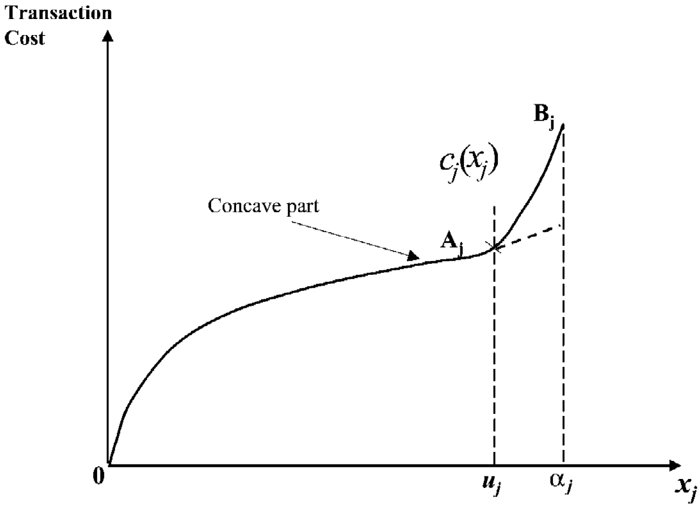

Figure 1. D.C. transaction cost function

3. Branch-and-bound algorithm

3.1. PROBLEM REFORMULATION

The discussion of the previous section leads to the following d.c. optimization problem :

$$\left|\begin{array}{lll} \text{maximize} & f(\boldsymbol{x}) \equiv \sum_{j=1}^{n} \left\{ r_j x_j - c_j(x_j) \right\} & \\ \text{subject to} & \sum_{t=1}^{T} p_t \left| \sum_{j=1}^{n} (r_{jt} - r_j) x_j \right| & \leqslant w, \\ & \sum_{j=1}^{n} x_j & = 1, \\ & x_j \in U_j \quad j = 1, \cdots, n. & \end{array}\right. \tag{8}$$

By standard results in linear programming [1, 8] the problem (8) can be converted to a linear system of inequalities as follows :

$$\left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) \equiv \sum_{j=1}^{n} \left\{ r_j x_j - c_j(x_j) \right\} \\ \text{subject to} & \sum_{t=1}^{T} y_t \leqslant w/2, \\ & y_t \geqslant p_t \sum_{j=1}^{n} (r_{jt} - r_j) x_j, \quad y_t \geqslant 0, \quad t = 1, \cdots, T, \\ & \sum_{j=1}^{n} x_j = 1, \quad x_j \in U_j, \qquad j = 1, \cdots, n. \end{array}\right. \tag{9}$$

THEOREM 1. *Problem (8) is equivalent to (9).*

Proof : The proof can be found in [8]. However, we will present it for completeness. Let us introduce a pair of non-negative variables $y_t, z_t,\ t = 1, \cdots, T$ and represent

$$\begin{aligned} &y_t - z_t = p_t \sum_{j=1}^{n} (r_{jt} - r_j) x_j, \quad t = 1, \cdots, T, \\ &y_t z_t = 0,\ \ y_t \geqslant 0,\ \ z_t \geqslant 0, \qquad t = 1, \cdots, T. \end{aligned}$$

Then the problem (8) is equivalent to

$$\left| \begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) \equiv \sum_{j=1}^{n} \left\{ r_j x_j - c_j(x_j) \right\} \\ \text{subject to} & \sum_{t=1}^{T} (y_t + z_t) \leqslant w, \\ & y_t - z_t = p_t \sum_{j=1}^{n} (r_{jt} - r_j) x_j,\ \ t = 1, \cdots, T, \\ & y_t z_t = 0,\ \ y_t \geqslant 0,\ \ z_t \geqslant 0, \quad t = 1, \cdots, T, \\ & \sum_{j=1}^{n} x_j = 1,\ \ x_j \in U_j, \qquad j = 1, \cdots, n. \end{array} \right. \tag{10}$$

Let $(x_1^*, \cdots, x_n^*,\ y_1^*, \cdots, y_T^*,\ z_1^*, \cdots, z_T^*)$ be an optimal solution of the problem (10) without complementarity constraints. Let us define

$$\hat{y}_t = \max(y_t^* - z_t^*, 0), \qquad \hat{z}_t = -\min(0, y_t^* - z_t^*), \quad t = 1, \cdots, T.$$

Then $(x_1^*, \cdots, x_n^*,\ \hat{y}_1, \cdots, \hat{y}_T\ \hat{z}_1, \cdots, \hat{z}_T)$ is also an optimal solution of (10) satisfying the condition $\hat{y}_t \hat{z}_t = 0,\ t = 1, \cdots, T$. This means that the complementarity conditions $y_t z_t = 0, t = 1, \cdots, T$ can be removed from (10). By noting the relation

$$\sum_{t=1}^{T} (y_t - z_t) = \sum_{t=1}^{T} p_t \sum_{j=1}^{n} (r_{jt} - r_j) x_j = \sum_{t=1}^{T} \sum_{j=1}^{n} p_t (r_{jt} - r_j) x_j = 0,$$

the relation

$$\sum_{t=1}^{T} (y_t + z_t) \leqslant w$$

is equivalent to

$$\sum_{t=1}^{T} y_t \leqslant w/2.$$

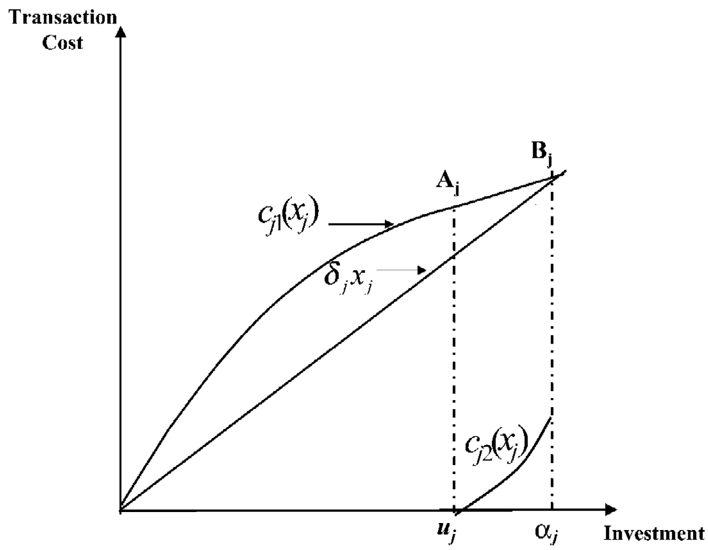

Figure 2. Underestimation of the concave cost function.

Also, the second constraint of (10) can be represented as

$$y_t \geqslant \sum_{j=1}^{n} p_t (r_{jt} - r_j) x_j, \quad t = 1, \cdots, T,$$

by noting the condition $z_t \geqslant 0$. This completes the proof of equivalence. □

Let

$$F = \{(\boldsymbol{x}, \boldsymbol{y}) | \sum_{t=1}^{T} y_t \leqslant w/2, \ \sum_{j=1}^{n} x_j = 1, \ y_t - p_t \sum_{j=1}^{n} (r_{jt} - r_j) x_j \geqslant 0, \ y_t \geqslant 0, \quad t = 1, \cdots, T\}. \tag{11}$$

Then the problem (10) can be denoted as follows :

$$(P_0) \left| \begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) = \sum_{j=1}^{n} \left\{ r_j x_j - c_j(x_j) \right\} \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & x_j \in U_j, \quad j = 1, \cdots, n, \end{array} \right. \tag{12}$$

where the cost function $c_j(x_j)$ is a d.c. function depicted in Figure 1.

Let us decompose $c_j(x_j)$ into the sum of two components $c_{j1}(x_j), c_{j2}(x_j)$, where $c_{j1}(\cdot)$ is a concave function and $c_{j2}(\cdot)$ is a convex function (see Figure 2).

The function $c_{j1}(x_j)$ is the maximal concave extension of the concave part, which means that $c_{j1}(x_j)$ is linear between u_j and α_j with its slope at point A_j.

The problem (12) is thus represented as follows.

$$(P_0)\quad \left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) = \sum_{j=1}^{n} \left\{ r_j x_j - c_{j1}(x_j) - c_{j2}(x_j) \right\} \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & x_j \in U_j\,, \quad j = 1, \cdots, n. \end{array}\right. \tag{13}$$

3.2. RELAXATIONS OF THE PROBLEM

First let us relax the conditions $x_j \in U_j$, $j = 1, \cdots, n$ and consider the following linearly constrained problem :

$$(Q_0)\quad \left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) = \sum_{j=1}^{n} \left\{ r_j x_j - c_{j1}(x_j) - c_{j2}(x_j) \right\} \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & 0 \leqslant x_j \leqslant \alpha_j, \quad j = 1, \cdots, n. \end{array}\right. \tag{14}$$

Second, let us replace the concave function $c_{j1}(x_j)$ by its convex envelope, i.e., a linear underestimating function $\delta_j x_j$ connecting 0 and B_j (See Figure 2). Let

$$g_j(x_j) = r_j x_j - \delta_j x_j - c_{j2}(x_j), \quad j = 1, \cdots, n, \tag{15}$$

and consider the following relaxation of Q_0 :

$$(\bar{Q}_0)\quad \left|\begin{array}{ll} \text{maximize} & g(\boldsymbol{x}) \equiv \sum_{j=1}^{n} g_j(x_j) \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & 0 \leqslant x_j \leqslant \alpha_j, \quad j = 1, \cdots, n. \end{array}\right. \tag{16}$$

This is a linearly constrained concave maximization problem which can be solved by standard methods.

Let $(\boldsymbol{x}^0, \boldsymbol{y}^0)$ be an optimal solution of $\bar{Q}_0$. Also, let $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ be an optimal solution of Q_0.

THEOREM 2. *The following relation holds :*

$$g(\boldsymbol{x}^0) \geqslant f(\boldsymbol{x}^*) \geqslant f(\boldsymbol{x}^0). \tag{17}$$

Proof. By definition, $g(\boldsymbol{x}) \geqslant f(\boldsymbol{x})$ for all feasible $(\boldsymbol{x}, \boldsymbol{y})$. Therefore

$$\begin{aligned} &\max\{g(\boldsymbol{x}) \mid (\boldsymbol{x}, \boldsymbol{y}) \in F, \ \ \boldsymbol{0} \leqslant \boldsymbol{x} \leqslant \boldsymbol{\alpha}\} \\ &\geqslant \max\{f(\boldsymbol{x}) \mid (\boldsymbol{x}, \boldsymbol{y}) \in F, \ \ \boldsymbol{0} \leqslant \boldsymbol{x} \leqslant \boldsymbol{\alpha}\} \ \geqslant \ f(\boldsymbol{x}^0). \end{aligned}$$

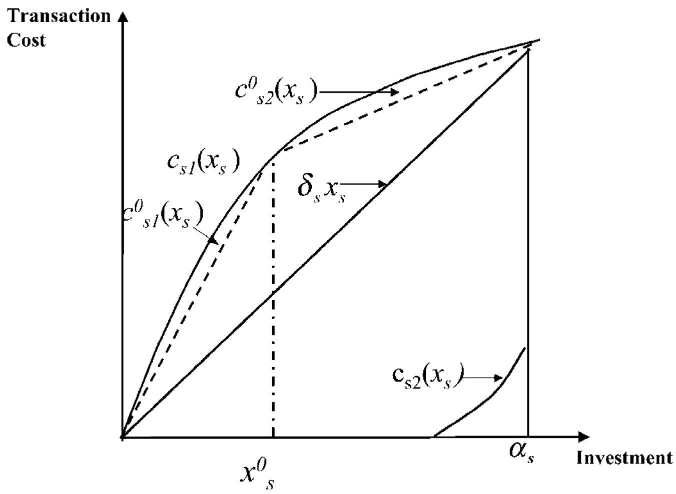

Figure 3. Subdividing the cost function

since $(\boldsymbol{x}^0, \boldsymbol{y}^0)$ is feasible for Q_0. □

Case 1: $g(\boldsymbol{x}^0) - f(\boldsymbol{x}^0) \leqslant \varepsilon$ for some small enough $\varepsilon > 0$. In this case, $\boldsymbol{x}^0$ is an approximate optimal solution of Q_0.
Case 2: $g(\boldsymbol{x}^0) - f(\boldsymbol{x}^0) > \varepsilon$

Let

$$x_s^0 = \operatorname{argmax}\{\, c_{j1}(x_j^0) - \delta_j x_j^0 \mid j = 1, \cdots, n \,\}$$

Here we apply the ω-subdivision scheme [6], in which the region $[0, \alpha_s]$ is divided into two subintervals $[0, x_s^0]$ and $[x_s^0, \alpha_s]$ where x_s^0 is the component of the optimal solution of $\boldsymbol{x^0}$ of $(\bar{Q}_0)$ which has the largest difference between the nonlinear transaction costs and underestimating linear transaction costs as shown in Figure 3.

Let us define two subproblems :

$$(Q_1)\left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & 0 \leqslant x_j \leqslant \alpha_j, \quad j \neq s \\ & 0 \leqslant x_s \leqslant x_s^0. \end{array}\right.$$

$$(Q_2)\left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & 0 \leqslant x_j \leqslant \alpha_j, \quad j \neq s \\ & x_s^0 \leqslant x_s \leqslant \alpha_s. \end{array}\right.$$

We are now ready to propose a branch-and-bound algorithm for solving Q_0. (See [6, 16] for more detailed explanation of the basic concepts of branch-and-bound algorithm.)

Algorithm 1 (Branch-and-Bound Algorithm)

(1) $\boldsymbol{Q} = \{Q_0\},\ \hat{f} = -\infty,\ k = 0.$

$\boldsymbol{\beta^0 = 0};\ \ \boldsymbol{\alpha^0 = \alpha};\ X^0 = \{\boldsymbol{x} \mid \boldsymbol{\beta^0} \leqslant \boldsymbol{x} \leqslant \boldsymbol{\alpha^0}\}$
Solve $\bar{Q}_0$ and let $(\boldsymbol{x^0}, \boldsymbol{y^0})$ be its optimal solution.

(2) If $\boldsymbol{Q} = \{\phi\}$, then go to (9); Otherwise go to (3).

(3) Choose a problem $Q_k \in \boldsymbol{Q}$: where $g(\boldsymbol{x^k}, \boldsymbol{y^k}) = \max\{g(\boldsymbol{x^l}, \boldsymbol{y^l}) \mid Q_l \in \boldsymbol{Q}\}$ where $(\boldsymbol{x^l}, \boldsymbol{y^l})$ is an optimal solution of $\bar{Q}_l$. $\boldsymbol{Q} = \boldsymbol{Q} \backslash \{Q_k\}$.

$$(Q_k)\quad \left|\begin{array}{ll} \text{maximize} & f(\boldsymbol{x}) = \displaystyle\sum_{j=1}^{n} \left\{r_j x_j - c_{j1}(x_j) - c_{j2}(x_j)\right\} \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & \boldsymbol{\beta}^k \leqslant \boldsymbol{x} \leqslant \boldsymbol{\alpha}^k. \end{array}\right.$$

(4) Let $c_{j1}^k(x_j)$ be a linear underestimating function of $c_{j1}(x_j)$ over the interval $\beta_j^k \leqslant x_j \leqslant \alpha_j^k,\ (j = 1, \cdots, n)$. Define the following nonlinear problem.

$$(\bar{Q}_k)\quad \left|\begin{array}{ll} \text{maximize} & g_k(\boldsymbol{x}) = \displaystyle\sum_{j=1}^{n} \left\{r_j x_j - c_{j1}^k(x_j) - c_{j2}(x_j)\right\} \\ \text{subject to} & (\boldsymbol{x}, \boldsymbol{y}) \in F, \\ & \boldsymbol{\beta}^k \leqslant \boldsymbol{x} \leqslant \boldsymbol{\alpha}^k. \end{array}\right.$$

If $\bar{Q}_k$ is infeasible then go to (2). Otherwise let $\boldsymbol{x^k}$ be an optimal solution of $\bar{Q}_k$.
If $|g_k(\boldsymbol{x^k}) - f(\boldsymbol{x^k})| > \varepsilon$ then go to (8). Otherwise let $f_k = f(\boldsymbol{x^k})$.

(5) If $f_k < \hat{f}$ then go to (7); Otherwise go to (6).

(6) Let $\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}}^k$ and eliminate all the subproblems Q_t for which $g_t(\boldsymbol{x^t}) \leqslant \hat{f}$.

(7) If $g_k(\boldsymbol{x^k}) \leqslant \hat{f}$ then go to (2). Otherwise go to (8).

(8) Let $c_{s1}(x_s^k) - c_{s1}^k(x_s^k) = \max\ \{\ c_{j1}(x_j^k) - c_{j1}^k(x_j^k) \mid j = 1, \cdots, n\ \}$,

$$\begin{aligned} S_{l+1} &= S_k \cap \{\boldsymbol{x} \mid \beta_s^k \leqslant x_s \leqslant x_s^k\}, \\ S_{l+2} &= S_k \cap \{\boldsymbol{x} \mid x_s^k \leqslant x_s \leqslant \alpha_s^k\}, \end{aligned}$$

and define two subproblems :

$$(Q_{l+1})\ \ \text{maximize}\ \{f(\boldsymbol{x}) \mid (\boldsymbol{x}, \boldsymbol{y}) \in F,\ \boldsymbol{x} \in S_{l+1}\},$$

$$(Q_{l+2})\ \ \text{maximize}\ \{f(\boldsymbol{x}) \mid (\boldsymbol{x}, \boldsymbol{y}) \in F,\ \boldsymbol{x} \in S_{l+2}\}.$$

$\boldsymbol{Q} = \boldsymbol{Q} \cup \{Q_{l+1}, Q_{l+2}\}, k = k + 1$ and go to 3°.

(9) Stop : $\hat{\boldsymbol{x}}$ is an ε optimal solution of Q_0.

THEOREM 3. *The algorithm provides an ε-optimal solution of Q_0 in finitely many steps.*

Proof. See Theorem 5.5 of Tuy [16]. □

In case α_j^l's are located below inflection point, then $\bar{Q}_l$ is a linear programming problem. Since the set F consists of $T+2$ linear inequality constraints, those components x_j such that $\beta_j^l < x_j < \alpha_j^l$ at an optimal solution is at most $T+2$, usually less than $T/2$ [8, 9]. Also, those variables at lower or upper bound are not subject to approximation error (note that the linear underestimating functions pass these points). This means that the upper bound generated by this procedure is a reasonably good approximation of the optimal solution.

3.3. PROBLEM REDUCTION AND MINIMAL TRANSACTION UNIT CONSTRAINTS

The problem Q_0 can, in principle, be solved by Algorithm 1. However, computation time is expected to increase rapidly as the number of assets increases. Further our eventual goal is to solve an even more difficult problem P_0 with minimal transaction unit (MTU) constraints.

To cope with these difficulties, we will use the following heuristic procedures which worked remarkably well when the cost function is concave [8, 9].

3.3.1. *(a) Elimination of Variables*

The first such procedure is problem reduction by eliminating all variables x_j such that $x_j^0 = 0$. Let us assume without loss of generality that the first $J (\leqslant T+2)$ components of $\boldsymbol{x^0}$ are positive and define the reduced problem as follows:

$$(Q_0') \quad \left| \begin{array}{ll} \text{maximize} & \tilde{f}(\boldsymbol{x}) = \sum_{j=1}^{J} \{r_j x_j - c_{j1}(x_j) - c_{j2}(x_j)\} \\ \text{subject to} & (x_1, \cdots, x_J, 0, \cdots, 0, \ \boldsymbol{y}) \in F, \\ & 0 \leqslant x_j \leqslant \alpha_j, \quad j = 1, \cdots, J. \end{array} \right. \tag{18}$$

Since the unit transaction cost is larger when x_j is smaller, those variables with zero investment in the optimal solution of $\bar{Q}_0$ is expected to remain zero throughout the branch-and-bound procedure. Numerical experiments reported in [9] show that this observation is usually true. In fact, only a few variables x_j such that $x_j^0 = 0$ takes positive value in the true optimal solution.

To solve the problem (P_0) with minimal transaction unit constraints, we may even remove those variables x_j such that x_j^0 is less than minimal transaction unit x_{j1}. This will further reduce the number of variables in the subsequent branch and bound procedure.

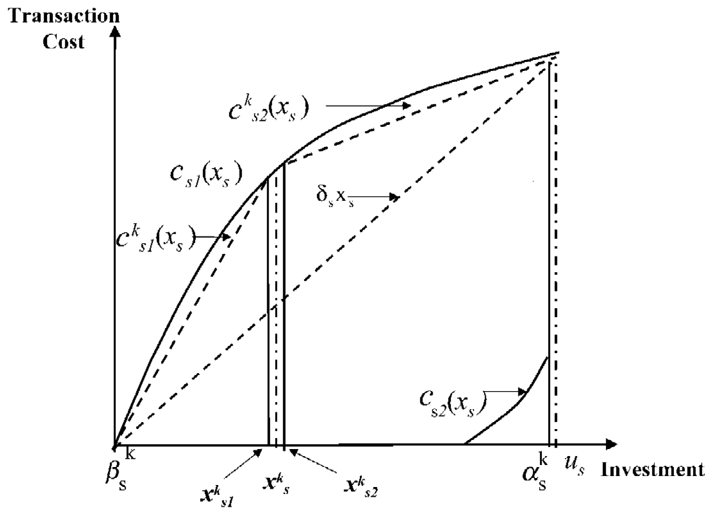

Figure 4. Subdivision strategy.

3.3.2. *(b) Alternative subdivision strategy*

The second such procedure is to replace the subdivision strategy as follows (See Figure 4). Let x_s be a subdivision variable in P_k. Then we will subdivide the interval $[\beta_s^k, \alpha_s^k]$ into subintervals $[\beta_s^k, x_{sl}^k]$ and $[x_{sl+1}^k, \alpha_s^k]$, where x_{sl}^k is the largest point of U_s to the left of $\hat{x}_s^k$ and x_{sl+1}^k is the smallest point of U_s to the right of $\hat{x}_s^k$. This will push many variables to take integral value in the optimal solution of the resulting subproblems.

4. Computational Experiments

We conducted numerical tests of Algorithm 1 using monthly data of 200 stocks chosen from Nikkei 225 Index. The program was coded by C^{++} and was tested on Pentium Pro 500 MHz with 256 MB memory. We choose $\varepsilon = 10^{-3}$ and 10^{-5} in our computation. We tested the algorithm for three different levels of investment, namely 15×10^9 yen, 25×10^9 yen, and 3×10^{10} yen using the transaction cost table of a leading security company of Japan. According to this table, the transaction cost is specified up to 1 billion yen, where the transaction cost function is a well-defined concave function. We assumed that the cost function is convex beyond this point. Also, we assumed that this part of the cost function is quadratic (see Figure 5). We may use an alternative functional form such as a piecewise linear convex function.

4.1. COMPUTATIONAL RESULTS OF PORTFOLIO CONSTRUCTION PROBLEM

Figure 6 shows the computation time for solving the problem Q_0 by Algorithm 1 for different number of assets without using problem reduction strategy. We solved

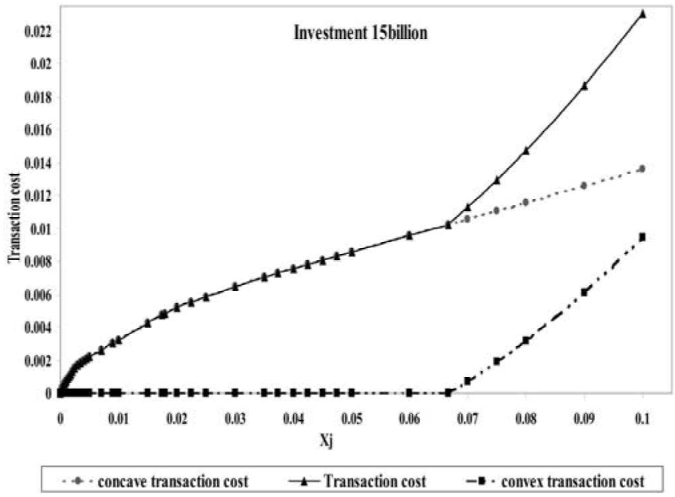

Figure 5. Transaction cost function.

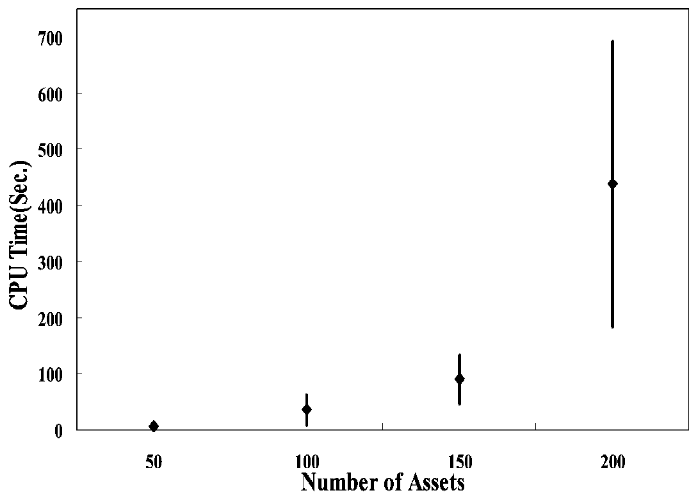

Figure 6. CPU time for solving Q_0 for different n.

ten test problems corresponding to different sets of data and plotted the average computation time and its standard deviation.

In general, the computation time increases exponentially as n increase. But we see from this figure that the average and the variance of the computation time increases less rapidly. This is primarily due to the fact that the starting solution generated by $\bar{Q}_0$ is a good feasible solution of Q_0, so that many subproblems are fathomed by bounds.

Figure 7 shows the computation time when the problem reduction strategy is employed. We see that the average computation time is less than 10% of Algorithm 1 without problem reduction. Also, the relative difference of the objective function

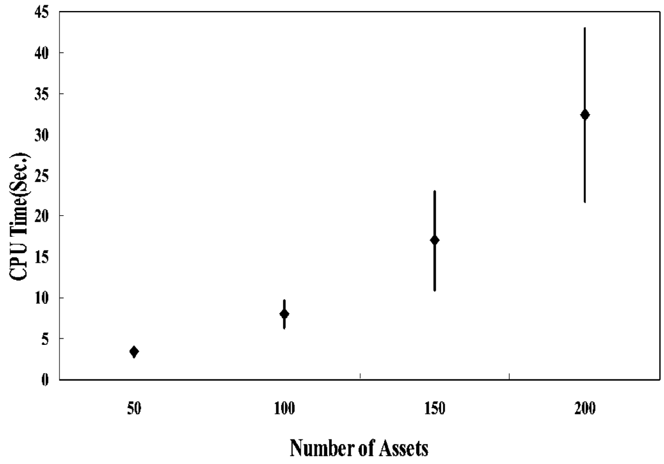

Figure 7. CPU Time with Problem Reduction

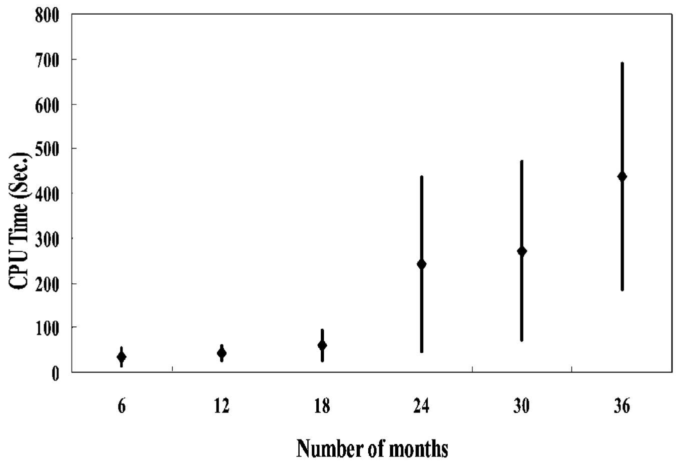

Figure 8. CPU time for solving Q_0 for different T

value of the algorithm with and without problem reduction is at most 1%, usually much less (less than 0.3%). We conclude from this that problem reduction is a very effective strategy for solving problems with large n. The average computation time increases less mildly as n increases (see Figure 6).

Figures 8 and 9 show the computation time as a function of T, when the number of data is $n = 200$. We see that the increase of the average computation time is very mild. We can safely conclude that the problem up to $T = 60$ can be solved in less than a few minutes. Let us note that the maximal size of T is usually less than 60 in practical applications.

Figures 10 and 11 show the efficient frontiers for different amount of investments and different level of upper bounds. It is clear that when investment increases and/or the upper bounds increases, the maximal net expected return tends to de-

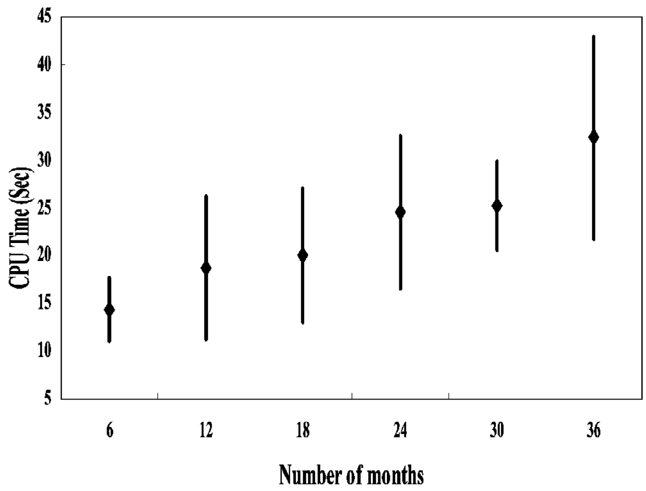

Figure 9. CPU time for different T with problem reduction.

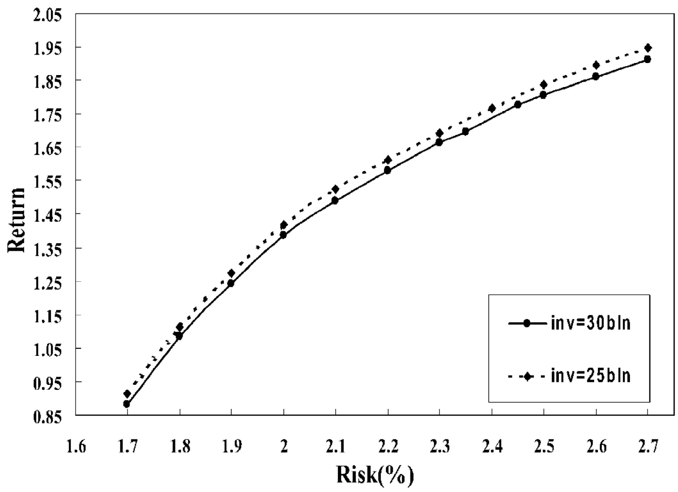

Figure 10. Efficient frontiers for different investments.

crease because the unit price of the assets increases when the amount of transaction increases, due to illiquidity/market impact effects.

Also we tested our algorithm for different levels of ε when $n = 200$, and $T = 36$. When ε is 10^{-3} and 10^{-5}, the average computation time is 32.3 and 91.3 s, respectively. The objective values are also more or less the same (see Figure 12). Therefore, the quality of the solution is not very sensitive to the level of ε.

4.2. PORTFOLIO CONSTRUCTION PROBLEM WITH MTU CONSTRAINTS

We conducted similar experiments for the problem with minimal transaction unit (MTU) constraints. There is no guarantee that a solution exists where all

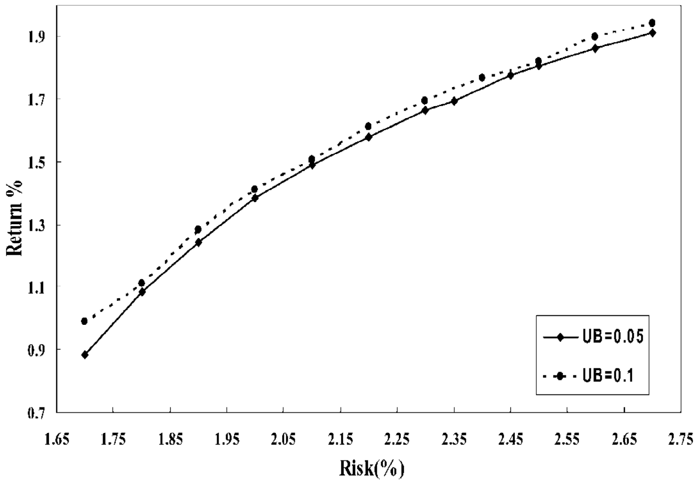

Figure 11. Efficient frontiers for different level of upper bounds.

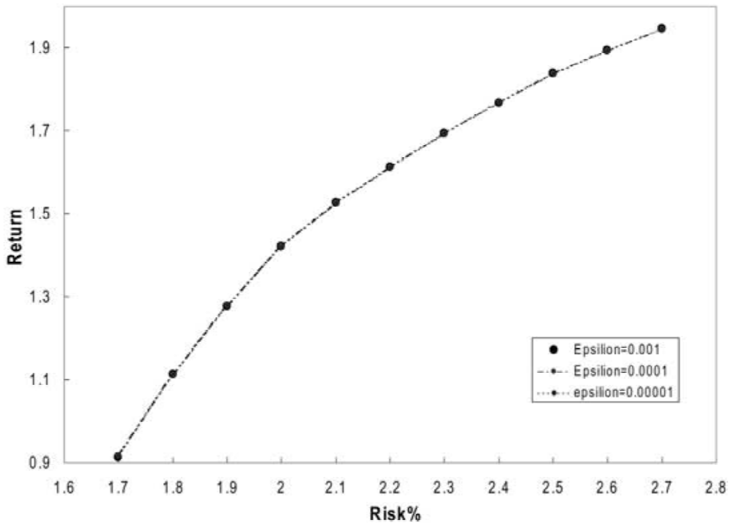

Figure 12. Efficient frontiers for different levels of ε.

the assets in the optimal solution satisfy MTU constraints since the amount of total investment is fixed. Therefore, we terminated computation as soon as the error becomes less than ε and rounded the solution to the nearest solution satisfying MTU constraints. Table 1 shows the statistics of the computation time for $n = 200, T = 36, \alpha_j = 0.05$ and 0.1 and for different levels of investments, where the problem reduction strategy was employed. We see that the number of assets which do not satisfy MTU constraints in the solution is at most 8. The necessary amount of fund adjustment needed to round all the assets to the nearest solution satisfying MTU constraints is less than 0.01% of the specified amount of

Table 1. Statistics for Different level of α and M

α_j's values	w	M billion (yen)	ρ	# of branc-hing	CPU time (s)	No. of assets violating MTU const-raints	No. of assets in the portfolio	Fund adjustment
0.05	1.8	30	1.08347	10	30	6	23	0.00214
	2.3	30	1.66373	10	29	5	22	0.002806
	1.8	25	1.11403	12	36	5	23	0.0062
	2	25	1.42034	14	41	7	22	0.000424
	2.6	25	1.89402	6	18	4	22	-4.72E-03
0.1	1.8	30	1.11027	18	50	6	13	-0.00215
	2.3	30	1.69506	6	16	6	12	-0.000806
	1.8	15	1.32266	22	65	6	15	0.0097066
	2.1	15	1.70371	14	41	6	12	-0.006793
	2.6	15	2.1002	6	16	3	12	0.0037

investment $M = 15 \times 10^9$, and it decreases to 0.003% when $M = 3 \times 10^{10}$, which is almost negligible from a practical point of view.

5. Conclusions and future research

We showed in this paper that the portfolio construction problem under d.c. transaction costs can be solved in a practical amount of time. The success depends upon the use of mean absolute deviation model, elaboration of the classical branch-and-bound method using ω-subdivision strategy and the problem reduction strategy using the special structure of the problem.

Let us emphasize that there are still a number of difficult (nonconvex) minimization problems and d.c. maximization problems in the field of financial optimization, some of which may be solved successfully by applying algorithms developed in the various fields of mathematical programming. We are now extending the branch-and-bound method proposed in this paper to index tracking problems with concave transaction costs and minimal transaction unit constraints, the results of which will be reported subsequently.

6. Acknowledgements

This research was supported by Grant-in-Aid for Scientific Research B(2)(10450041) and B(2)(11558046) of the Ministry of Education, Science, and Culture. Also, the first author acknowledges the generous support of IBJ-DL Financial Technologies, Co., and The Tokyo Trust and Banking, Co., Ltd.

References

1. Chvátal, V., *Linear Programming,* Freeman and Co., 1983.
2. Elton, J.E. and Gruber, M.J., *Modern Portfolio Theory and Investment Analysis* (Fourth edition), John Wiley & Sons, New York, 1991.
3. Falk, J.E. and Soland, R.M., An Algorithm for Separable Nonconvex Programming Problems, *Management Science,* 15 (1969) 550–569.
4. Konno, H., Suzuki,T. and Kobayashi, D., A Branch and Bound Algorithm for Solving Mean-Risk-Skewness Model, *Optimization Methods and Software*, 10 (1998) 297–317.
5. Konno, H. and Suzuki, K., A Mean Variance Skewness Portfolio Optimaization Model, *J. of the Operation Research Society of Japan* 38 (1995) 173–187.
6. Konno, H., Thach, P.T and Tuy, H., *Optimization on Low Rank Nonconvex Structures,* Kluwer Academic Publishers, 1997.
7. Konno, H. and Yamazaki, H., Mean Absolute Deviation Portfolio Optimization Model and Its Application to Tokyo Stock Market, *Management Science* 37 (1991) 519–531.
8. Konno, H. and Wijayanayake, A., Mean-Absolute Deviation Portfolio Optimization Model under Transaction Costs, *J. of the Operation Research Society of Japan* 42 (1999) 422–435.
9. Konno, H. and Wijayanayake, A., Portfolio Optimization Problems under Concave Transaction Costs and Minimal Transaction Unit Constraints, *Mathematical Programming* 89 (2001) 233–250.
10. Luenberger D.G., *Linear and Nonlinear Programming* (Second edition), Addison-Wesley Publishers, Reading, MA, 1984.
11. Markowitz, H.M., *Portfolio Selection : Efficient Diversification of Investments (Second edition),* John Wiley & Sons, New York, 1991.
12. Mulvey, J.M., Incorporating Transaction Costs in Models for Asset Allocation, in *Financial Optimization* (W. Ziemba et al., eds.), Cambridge University Press, New York, (1993) pp. 243–259.
13. Ogryczak, W. and Ruszczynski, A., From Stochastic Dominance to Mean-Risk Model, *European J. of Operational Research* 116 (1999) 33–50.
14. Perold, A.F., Large Scale Portfolio Optimization, *Management Science* 30 (1984) 1143–1160.
15. Phong, T.Q., An, L.T.H. and Tao, P.D., Decomposition Branch and Bound Method for Globally Solving Linearly Constrained Indefinite Quadratic Minimization Problems, *Operations Research Letters* 17 (1995) 215–220.
16. Tuy, H., *Convex Analysis and Global Optimization,* Kluwer Academic Publishers, Dordrecht, 1998.