A SUCCESSIVE SURROGATE CONSTRAINT SEARCH METHOD FOR INDEX FUND PORTFOLIO OPTIMIZATION

著者	James Ross J.W, Ferd Glover, Nakagawa Yuji, Cesar Rego
page range	1-30
year	2010-10
URL	http://hdl.handle.net/10112/2581

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

A SUCCESSIVE SURROGATE CONSTRAINT SEARCH METHOD FOR INDEX FUND PORTFOLIO OPTIMIZATION

Yuji NAKAGAWA
Department of Informatics, Kansai University, Takatsuki-shi, 569–1095 Japan. nakagawa@res.kutc.kansai-u.ac.jp
Ross J.W. James
Department of Management, University of Canterbury, Private Bag 4800, Christchurch, 8140, New Zealand, ross.james@canterbury.ac.nz
César Rego
School of Business Administration, University of Mississippi, University, MS 38677, USA, crego@bus.olemiss.edu
Fred Glover
Leeds School of Business, University of Colorado, Boulder, CO 80309-0419, USA, fred.glover@colorado.edu

Large-scale separable nonlinear integer programming problems with multiple constraints can be solved optimally by using the improved surrogate constraints (ISC) method. We propose a successive optimization solver based on the ISC method to solve a series of non-separable and non-convex optimization problems. The successive optimization solver is applied to the index-tracking optimization problem for two variants of the index-plus-alpha-funds model. In this model the fund tracks a market benchmark and outperforms it by a small amount, alpha. The first index fund model is optimized by explicitly restricting the number of stocks while in the second model the number of stocks is implicitly determined by incorporating execution costs and trading units into the optimization process. The models are applied to the Tokyo Stock Market where high quality portfolios were found for the Japanese 225 NIKKEI stocks and 1440 TOPIX stocks.

Key words: Nonconvex programming, Multidimensional Nonlinear Knapsack Problem, Quadratic Discrete Programming, Index Fund Optimization, Interactive Optimization Method, Surrogate Constraint Method.

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

1. Introduction

In this paper we propose a successive discrete optimization method, which may be considered as an extension of successive linearization methods, for solving non-convex financial optimization problems with discrete variables. The main difference between the existing successive linearization method and the present method is what terms will be linearized. The existing successive linearization methods use linearization techniques for all nonlinear terms, but the present method only for nonseparable terms in all nonlinear terms. Therefore the present method doesn't easily receive the influence of the error caused by the linear approximation. Indeed we can use whole feasible region as the initial search region, but the other linearization methods need to use small regions (step bounds or trust region) for variables, The method presented is very powerful and can even solve non-convex portfolio optimization problems, which existing nonlinear solvers are unable to solve. The method applies to producing portfolios that not only track a market benchmark but also produce a return higher than the benchmark, i.e. a plus-alpha profit, under non-convex transaction cost constraint. A plus-alpha fund may enable a trust to lower or remove trust fee charged for managing the fund. Trust fees are relatively small amounts, but in total the cost can be high for a long term fund such as pension fund.

Successive Linear Programming (Griffith and Stewart. 1961, Palacios-Gomez et al. 1982, and Zhang et al. 1985) solves nonlinear optimization problems via a sequence of linear programs. Kanzow et al. (2005) present a successive linearization method with a trust region-type globalization for the solution of nonlinear semidefinite programs. The present interactive method solves nonconvex optimization problems via a sequence of the nonlinear knapsack problems and only uses linearization techniques for nonseparable terms.

This paper deals specifically with the index tracking optimization problem, which is a variant of the Markowitz (1952) mean-variance model, and although half a century has already passed, the model is still used in many industries and it remains a very important theoretical model. The index fund problem, which optimizes a portfolio in order to track a market index, is a descendant of the Markowitz model. The index fund is popular in the passive investments like the pension fund. The pension fund is a set of common assets (often stocks) pooled to generate stable growth over the long term with the aim of providing pensions for employees when they retire. Some employees receive their pensions after contributing for 40 years, but the money paid 40 years ago and invested to the stock market has reduced by up to 40% due to trust fees. The trust fees in Japan are 0.5-1.2% per year for index tracking funds. The trust fees of enhanced index tracking funds are much higher. Plus-alpha funds may enable the trust to return invested money without incurring trust fees or even providing an additional return on the money. The risk of the proposed plus-alpha fund appears to be almost the same as an index- tracking fund with a cardinality constraint.

In real life passive investment management applications, the following requirements need to be taken into consideration (Perold 1984, Edirisinghe, Naik, and Uppal. 1993):

1) Restrictions on the trading unit for stocks: Often there is a minimum quantity (lot) for trading on the market. This means the portfolio weights can be modeled with discrete values.

2) Transaction Costs: The transaction cost is the difference between the execution price and the true value of the stock. It includes explicit costs (e.g. commissions and taxes) and the market impact cost which is dependent on market liquidity and how the stock price changes with the transaction. These costs are generally separable nonlinear non-convex functions of the portfolio weights.

Although these requirements are crucial for the effective optimization of index funds, their consideration requires the solution of non-separable non-convex discrete optimization problems that are very difficult to solve. To overcome these difficulties, optimization models proposed to date have focused on different types of linearization techniques to reduce the problem complexity.

Index fund problems are generally modeled as a mixed integer program with a nonlinear objective function. The standard deviation is usually taken into the objective function as the measure of risk, following the pioneering work of Markowitz (1952). These problems are usually very difficult to solve exactly, even for a relatively small number of stocks (Gaivoronski et al. 2004). In order to reduce the complexity of the problem, Konno and Yamazaki (1991) use an absolute deviation (a linear function) instead of the standard deviation (a quadratic function) for the calculation of the tracking error of the portfolio. Due to the difficulty of the problem, approximation algorithms, e.g., reoptimization heuristic , genetic algorithms, simulated annealing, and tabu search algorithms Chang et a. 2000, Jobst et al. 2001), are usually required to find high quality solutions, however these are not necessarily optimal. Tabata and Takeda (1995) formulate this problem as a multi-objective 0-1 quadratic programming problem that simultaneously minimizes the number of stocks and the expected value of the squared difference between the return on the tracking portfolio and the return on a benchmark index. More advanced heuristic methods rely on a number of metaheuristic strategies (. For a discussion and a computational study of these approaches we refer the reader to Chang et al. (2000). More recently, Gaivoronski et al. (2004) implemented the following two stage procedure based on ideas originally proposed in Jobst et al. (2001):

1) Compute weights of current stocks by solving the portfolio selection problem without any restriction on the number of different stocks. Rank the stocks in descending order of their weight.

2) Select a subset of the highest ranked stocks in the current portfolio to create a reduced portfolio selection problem. Repeat steps 1 and 2 until the number of required stocks is reached.

Jobst et al. (2001) called this approach the “reoptimization heuristic”. Their computational results showed that the reoptimization heuristic is preferable to applying meta-heuristic methods to this problem. Reoptimization heuristics can produce high quality solutions for index tracking optimization models with cardinality constraints, but the effectiveness of the heuristic depends on the software used to solve the portfolio selection instances without any limitation on the number of different stocks.

Konno and Hatagi (2005) use branch and bound to construct an index-plus-alpha portfolio with concave transaction costs based on historical data of the stocks in the market. Like Konno and Yamazaki (1991), they use absolute deviation to assess the tracking error of the portfolio which they minimize by using a branch and bound algorithm.

The surrogate constraints approach was first proposed by Glover (1965) to solve 0-1 Integer Programs. The approach breaks the original multi-constraint problem into a series of surrogate problems with a single constraint. The problem with surrogate constraint methods, however, is that they may fail to produce an exact optimal solution of the original problem due to the existence of a surrogate duality gap (Nakagawa and Miyazaki 1981). To overcome this, Nakagawa (1998, 2003) proposed the improved surrogate constraints (ISC) method for solving multidimensional nonlinear knapsack problems (MNK). In order to close the surrogate duality gap, the ISC method enumerates all solutions within a target region which is known to include the exact optimal solution. The ISC method has proven to be very effective at solving large instances from the standard MNK benchmarks as well as instances arising in classical reliability problems (Ohnishi et al. 2007). However, the preceding ISC method does not handle nonseparable objective functions. In our earlier work dealing with separable reliability optimization problems (Hikita et al. 1992), we converted non-separable functions into the separable functions using a first order approximation. Such nonseparability arises, for instance, in models for financial optimization problems, for example the index tracking problem under cardinality constraints and/or transaction cost. The index tracking problem is important as an investment product for pension plans, since pension funds are more commonly passively managed to generate stable growth over the long term (Beasley, Meade, Chang 2003 and Gaivoronski, Krylov, van der Wijst, 2005).

In order to solve non-separable non-convex discrete optimization problems, this paper proposes a successive discrete optimization technique, called the "CubeWalk". It is similar to successive linear programming in that an area around a given starting point is evaluated with an approximation and a new point is selected from this area and this becomes the new starting point for the next iteration. However the method of approximation and optimization technique used in CubeWalk is very different to those proposed in the successive linear programming literature. CubeWalk is applied to non-convex quadratic discrete optimization models, for example index tracking optimization models with a cardinality constraint, i.e. a limit on the number of stocks allowed into the portfolio, while also considering trading units and transaction costs. For index tracking optimization models that consider trading units and transaction cost, the trading unit is the minimum quantity for trading on the markets, for example one lot is 1000 shares. For transaction costs, we use a commission table from a Japanese security company and the execution costs are randomly generated.

This paper extends the work of Konno and Hatagi (2005) by developing index-plus-alpha portfolios based on historical stock data but, unlike Konno and Hatagi (2005), the historical data is used to create a pseudo trajectory to choose stocks that are improving over time, rather than the stocks that have had little growth. Also unlike Konno and Hatagi (2005) we also take into account trading unit size. The technique we propose can converge to the exact optimal of the index-tracking problem with a nonconvex transaction cost constraint and minimum trade size restrictions. The models were tested using data from the Tokyo Stock Market and using the 225 NIKKEI and 1440 TOPIX stocks over the period April 1994 - March 1999.

The paper is organized as follows. In Section 2, we discuss non-convex discrete optimization problems and present a new method for solving these problems which we call CubeWalk. In Section 3, we discuss the index fund problem with a cardinality constraint and two types of rates-of-return. Additionally we consider creating Index-Plus-Alpha portfolios, although we do not explicitly consider Index-tracking portfolios, which are a special case of the Index-Plus-Alpha portfolio. In Section 4, we describe the Index-Plus-Alpha Fund considering the trading units and transaction cost. Some example problems are solved. Finally, in Section 5, we conclude and summarize the material presented

2. Index Fund Problems

2.1 Index Fund Portfolios with a constraint limiting the number of stocks

Let R_0 be a random variable of the rate of return of the target index (e.g., NIKKEI 225) to be tracked, R_i be the return of a stock $i \in \{1,2,\ldots,n\}$ and R be the random variable of the return of the portfolio obtained, i.e. $R = \sum_{i=1}^{n} R_i \xi_i$ where ξ_i is the weight of stock *i*. The problem is formulated as:

$$\mathrm{P^A}: \ \text{Minimize}\, S(\boldsymbol{\xi}) = E[(R_0 - R)^2]$$

$$= {\mu_0}^2 + \sigma_{00} + \sum_{i=1}^{n}\sum_{s=1}^{n}(\mu_i\mu_s + \sigma_{is})\xi_i\xi_s$$

$$-2\sum_{i=1}^{n}(\mu_0\mu_i + \sigma_{0i})\xi_i$$

$$\text{subject to} \quad \sum_{i=1}^{n}\xi_i = 1,$$

$$\sum_{i=1}^{n} z_i(\xi_i) = q,$$

$$\xi_i^L \le \xi_i \le \xi_i^U \quad (i \in N),$$

where $E[\bullet]$ represents the expected value of the random variable $\bullet$, μ_i is the mean rate of return for stock i, i.e. $\mu_i = E[R_i]$, σ_{is} is covariance between stocks i and s, i.e. $\sigma_{is}=E[(R_i - \mu_i)(R_s - \mu_s)]$. $\mu_0=E[R_0]$, and $\sigma_{0s}=E[(R_0 - \mu_0)(R_s - \mu_s)]$ $(s = 0,1,\ldots,n)$, q is the number of stocks selected and

$$z_i(\xi_i) = \begin{cases} 0 & (\xi_i = 0) \\ 1 & (\xi_i > 0) \end{cases}.$$

The objective function, i.e. the active risk, is also defined as:

$$S(\boldsymbol{\xi}) = E[(R_0 - R)^2] = V[R_0 - R] + (E[R_0] - E[R])^2,$$

where $V[\bullet]$ means the variance of $\bullet$. This active risk includes not only the variance error but also the mean error of returns.

In order to satisfy the equality constraints in the problem, the following inequality constraints are used instead of the equality constraints:

$$\left|\sum_{i=1}^{n}\xi_i - 1\right| \le \varepsilon$$

$$\sum_{i=1}^{n} z_i(\xi_i) - q \le \delta, \quad \sum_{i=1}^{n} z_i(\xi_i) \ge q$$

Note that any slack remaining in the ε can always be eliminated by normalizing the weights of the stocks, forcing $\sum_{i=1}^{n} \xi_i$ to be equal to 1. Also any slack remaining in δ can be resolved by removing the stocks with the smallest weight and then normalizing the weights. Also note that in practice δ is limited to between 0% - 10% of the value of q.

We define the rate of return R_i for each stock $i \in \{0,1,2,\ldots,n\}$. Note that stock i = 0 means the market index that the portfolio should track. Let t_C be the time of investment, i.e. the time that the stock was purchased. The stock prices of the interval $t \in \{0,1,\ldots,t_C\}$ can be used for optimizing the portfolio. The rate of return that we consider is based on a periodical return rate (%) for each stock $i \in \{0,1,2,\ldots,n\}$. This is used by Tabata and Takeda (1995) and is normally used by Securities companies and most of the literature, and is defined as:

$$R_{it} = 100 \times \left(\frac{v_{it} - v_{i,t-1}}{v_{i,t-1}} \right) \ (i \in \{0,1,2\ldots,n\},\ t \in \{2,\ldots,t_C,t_C+1,\ldots\}\ ),$$

where v_{it} is the price of stock i at time t.

3. 2. Index-Plus-Alpha Portfolios with Cardinality Constraint

In this section we formulate an Index-Plus-Alpha portfolio, which is a generalization of the Index-tracking portfolio. The pseudo trajectory can be created in a number of different ways, depending on the type of portfolio that is wanted. The simplest form is a level pseudo trajectory R^{Pse} whereby

$$R^{Pse} = R_0 + \alpha \ ,$$

where R_0 is the periodical return rate of the market index. This provides our basic plus alpha portfolio. However a potentially more useful pseudo trajectory is to consider those shares that have been increasing in value over the time period we have historical data and converges to the index. We call this an up-trend pseudo trajectory with a β % difference, whereby the β is the amount the portfolio converges to the index per month. The pseudo trajectory is therefore defined as:

$$v_t^{Pse} = v_{0t}(1-\beta)^{(t_c - t)} \ \left(t \in \{1, \ldots, t_C\}\right).$$

A random variables R^{Pse} of return of the pseudo index   is defined such that.

$$R_t^{Pse} = 100 \times \left(\frac{v_{0t}^{Pse} - v_{0,t_c}^{Pse}}{v_{0,t_c}^{Pse}} \right) \ (t \in \{1, \ldots, t_C, t_C + 1, \ldots\} \ ).$$

The trajectory R^{Pse} is the past returns of stock based on the specific time t_C that the plus-alpha portfolio is purchased. which we are wanting to track and includes an extra amount of return, or alpha level, based on the stock price. R^F is the return on the stocks based on the stock price, that is:

$$R_{it}^F = 100 \times \left(\frac{v_{it} - v_{i,t_c}}{v_{i,t_c}} \right) \ (i \in \{1, 2 \ldots, n\}, \ t \in \{1, \ldots, t_C, t_C + 1, \ldots\} \ )$$

Essentially the formulation is the same as the Index tracking portfolio except that the objective is re-defined as:

$\mathrm{P^B}$:   Minimize $S(\xi) = E[(R^{Psd} - R^F)^2]$

where $R^F = \sum_{i=1}^{n} R_i^F \xi_i$.

Using the stock prices for the returns in the optimization provides good tracking in the price trajectory, as we see in our computational results.

Clearly down-trend pseudo trajectories are not desirable to model in this way as in these cases deviating from the index in order to maintain stock value is desirable.

When we evaluate the index-plus-alpha portfolio, we calculate the variance based on the periodical return rate, that is:

$$e = V[R_0 - R]$$

3.3 Index-Plus-Alpha Portfolio with Trading Units and Transaction Cost

The present index fund problem is formulated as a nonconvex discrete programming problem with a non-smooth constraint function. Considering the trading unit ω_i for each stock $i \in \{1,2,\ldots,n\}$, the problem is formulated as:

$$\text{P}^{\text{B}}: \quad \text{Minimize} \quad S(\xi) = E[(R^{Pse} - R^F)^2]$$

$$\text{subject to} \quad \left| \sum_{i=1}^{n} \xi_i - 1 \right| \leq \varepsilon$$

$$\sum_{i=1}^{n} h_i(\xi_i) \leq \theta C,$$

$$\xi_i \in \{0, \omega_i, 2\omega_i, 3\omega_i, \ldots, a_i\omega_i\} \quad (i = 1,2,\ldots,n)$$

Where ε is the maximum error allowed in the total weight of the portfolio (buying the exact portfolio can be impossible due to having to buy in trading unit lots), C is the total investment, θ is the maximum transaction cost allowed as a proportion of the total investment C, and $h_i(k_i\omega_i)$ is the execution cost associated with the investment $k_i\omega_i$ $(k_i = 0,1,\ldots,a_i,\ i = 1,2,\ldots,n)$.

3. CubeWalk Method for Non-convex Discrete Optimization Problem

Consider the following non-separable nonconvex optimization problem:

$$\text{P}: \quad \text{maximize} \quad f(\xi) = r(\xi) + \sum_{i=1}^{n} f_i(\xi_i)$$

$$\text{subject to} \quad g_j(\xi) = \sum_{i=1}^{n} g_{ji}(\xi_i) \leq b_j \quad (j \in M)$$

$$\xi_i^L \leq \xi_i \leq \xi_i^U \quad (i \in N)$$

where the variables $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$, the number of constraints $M = \{1,2,\ldots,m\}$, the number of variables $N = \{1,2,\ldots,n\}$, the function $r(\xi)$ may be a differentiable non-separable function, and the functions $f_i(\xi_i)$, $g_{ji}(\xi_i)$ $(i \in N,\ j \in M)$ are separable and are not assumed to be differentiable. It should be noted that we do not assume convexity for this problem. In order to apply the ISC method to this non-separable problem, we consider a separable problem with a hyper-cube region in the neighborhood of each current solution (or pivot). Let $\boldsymbol{\xi}^{(\ell)}$ $(\ell = 0,1,2,\ldots)$ denote the pivots and ρ ($\rho \le \max_{i\in N}\left|\xi_i^U - \xi_i^L\right|$) be the size of the sides of the cube. We introduce the discrete variable $x_i \in \{1,2,\ldots,d+1\}$ to divide the neighborhood $\left|\xi_i - \xi_i^{(\ell)}\right| \le 0.5\rho$ $(i \in N)$ of the pivot $\boldsymbol{\xi}^{(\ell)}$ into d segments. $\xi_i = \xi_i^{(\ell)} + \Delta\xi(x_i, \rho, d)$,

$$\Delta\xi(x_i, \rho, d) = \rho(x_i - 1)/d - 0.5\rho.$$

The cube optimization problem in the neighborhood of the pivot $\boldsymbol{\xi}^{(\ell)}$ is as follows:

$$\mathrm{P}(\boldsymbol{\xi}^{(\ell)}, \rho, d) : \text{maximize} \quad f^{(\ell)}(\mathbf{x}) = r(\boldsymbol{\xi}^{(\ell)}) + \sum_{i=1}^{n} f_i^{(\ell)}(x_i, \rho, d)$$

$$\text{subject to} \quad g_j^{(\ell)}(\mathbf{x}) = \sum_{i=1}^{n} g_{ji}^{(\ell)}(x_i, \rho, d) \le b_j \quad (j \in M)$$

$$\xi_i^L \le \xi_i^{(\ell)} + \Delta\xi(x_i, \rho, d) \le \xi_i^U \quad (i \in N),$$

$$x_i \in \{1,2,\ldots,d+1\} \quad (i \in N),$$

where $f_i^{(\ell)}(x_i, \rho, d) = \partial r(\boldsymbol{\xi}^{(\ell)})/\partial\xi_i \cdot \Delta\xi(x_i, \rho, d) + f_i(\xi_i^{(\ell)} + \Delta\xi(x_i, \rho, d))$, $g_{ji}^{(\ell)}(x_i, \rho, d) = g_{ji}(\xi_i^{(\ell)} + \Delta\xi(x_i, \rho, d))$, and $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. The problem $\mathrm{P}(\boldsymbol{\xi}^{(\ell)}, \rho, d)$ is a separable discrete optimization problem that can be formulated as a nonlinear knapsack problem that can be solved exactly and efficiently (Nakagawa, 2003). The accuracy of this approach will depend on the size of $\Delta\xi(x_i, \rho, d)$. As $\Delta\xi(x_i, \rho, d) \to 0$, $\mathrm{P}(\boldsymbol{\xi}^{(\ell)}, \rho, d) \to$ the local optimum. The interactive optimization procedure for solving the original problem P is outlined in Figure 1. We assume that an experienced operator is interacting with this algorithm.

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

Figure 1: Interactive CubeWalk Procedure

Input: the original problem P , initial solution $\boldsymbol{\xi}^{(0)}$, allowable error ε ;
1. Set $\ell \leftarrow 0$;
2. **Repeat**
3. Set $\ell \leftarrow \ell + 1$;
4. Determine the cube size ρ and the number of divisions d **by** using the empirical knowledge of the operater;
5. Consider a problem $\mathrm{P}(\boldsymbol{\xi}^{(\ell-1)}, \rho, d)$ corresponding to $\boldsymbol{\xi}^{(\ell-1)}$ and yield an optimal solution $\mathbf{x}^{(\ell)}$ by using the and appropriate solver, in this case the ISC Solver ;
6. Set $\xi_i^{(\ell)} \leftarrow \xi_i^{(\ell-1)} + \Delta\xi(x_i^{(\ell)}, \rho, d)$ $(i = 1, 2, \ldots, n)$;
7. **Until**(ρ and $f(\boldsymbol{\xi}^{(\ell)})$ are small enough)
8. Output the current solution $\xi^{(\ell)}$ as the optimal solution of P;

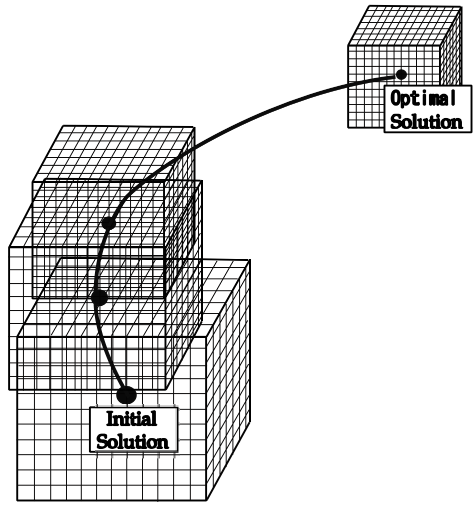
The size of ρ and the number of divisions *d* on line 4 are determined by the operator through their experience and the history of previous solutions. They also determine whether the procedure is to terminate on line 7.

The empirical knowledge used in the Cube Walk is as follows:

1) When the current solution $\boldsymbol{\xi}^{(\ell-1)}$ does not improve the objective function value $f(\boldsymbol{\xi}^{(\ell-1)})$, the cube size ρ is made smaller.
2) When the improvement $\left| f(\boldsymbol{\xi}^{(\ell-1)}) - f(\boldsymbol{\xi}^{(\ell-2)}) \right|$ becomes very small continuously several times, the cube size ρ is enlarged a little.
3) In order to speed up the algorithm, the cube size ρ is enlarged a little after around 10 times with the same size.

Figure 2: CubeWalk

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010



2.2 Interaction Guidelines

The Algorithm in Figure 1 requires the operator to choose the size of the cube, ρ, and the number of divisions, *d*. These two parameters will determine the diversification (how widely the search looks) and intensification (how carefully the search looks around the current solution) of the search. The choice of *d* will also dictate the computational time that is required at each iteration, the larger the number of divisions, the more CPU intensive the search will be.

In our experiments we have found that a good approach is to start with a small ρ, in order to intensify the search around the initial solution, then after 10 iterations or if the improvements in the solution quality are small, increase ρ in order to diversify the search and enable it to get out of local minima for a small number of iterations, then once again make ρ small and repeat the process. In or experiments we kept *d* as a constant. Its value would be dictated to by the efficiency of the solver being used. In the case of the ISC solver, discussed below, we set *d* to 100.

The performance of the solver used directly influences the performance of the Cube Walk and dictates the number of divisions that can be practically solved at each iteration.

The improved surrogate constraints method was proposed by Nakagawa (1998, 2003) and can be used for solving separable nonlinear integer programming problems with multiple constraints. This approach, like the standard surrogate constraints methods (Glover 1968), solves a succession of surrogate constraints problems that have a single constraint rather than the original multiple constraint problem. The constraints are combined using a surrogate multiplier vector which, if optimal, will solve the original problem exactly if there is no duality gap, however duality gaps are common and therefore the optial solution is not found. . The improved surrogate constraint (ISC) method closes the surrogate duality gap by enumerating solutions that are at least as good as a particular target objective value while still meeting all of the original constraints. The target is obtained through the use of a heuristic. As the value of the target is improved, the duality gap is reduced and the optimal solution can eventually be found.

In the literature the ISC method has been used to solve both linear and nonlinear problems. Nakagawa, James, Rego, (2007) report computational results on 0-1 Knapsack problems found that even though the performance of the ISC is not as competitive as CPLEX for 10 constraint instances, The ISC is very efficient for solving problems with 5 or less constraints. Nakagawa et al. (2007) compared global optimization solvers such as Bonmin (Bonami and Lee 2006, Bonami et al., 2005), Baron (Sahinidis and Tawarmalani 2005, Tawarmalani and Sahinidis, 2004), Interval Global solver in Frontline Premium Solver Platform (2005), and for convex quadratic knapsack instances, the CPLEX quadratic solver and LP/Quadratic solver in Frontline Premium Solver Platform were compared. In all cases the ISC performed very well and solved many problems that were unable to be solved using the other solvers. The ISC has also outperformed many heuristic techniques when solving Non-Convex Separable Integer Programming problems (Ohnishi et al. 2007). The ISC has also been used successfully as a heuristic to solve 0-1 knapsack instances (Nakagawa et al. 2005b) outperforming many of the metaheuristic approaches proposed in the literature, for example Vasques and Hao (2001), Chu and Beasley (1998) and Vesquez and Vimont (2005).

The termination criteria of the CubeWalk procedure in Figure 1 requires both the size of the cube, ρ, and the amount of improvement in the objective function, $f(\xi^{(\ell)})$, to be "small". This again is determined by the operator. We require the cube size to be small because if the cube size is large, then we are in a very diverse phase of the search and therefore the objective may not improve. If however we have a small cube size then we in an intensification phase of the search and we should be able to find improved solutions, if they are present. Also we need to consider the amount of the improvement, if significant gains in the objective function are still being made, then we would not want to terminate the search however, if little or no gains are getting made then the search should terminate. What is considered to be "large" and "small" is problem dependent and needs to be determined by the operator of the procedure.

4. Computational Experiments

The performance of the Cube Walk will now be evaluated on various index fund problems. We use the expected value of the squared difference between the return on the tracking portfolio and the return on a benchmark index, see Tabata and Takeda (1995), as the means of evaluating how well the the portfolio matches the historic index data .

4.1 Index-plus alpha fund problems with cardinality constraints

In this section, we present computational results for solving P^A using the Reoptimization Heuristic with the Generalized Reduced Gradient solver (RH(GRG)) (Jobst et al., 2001) and also a hybrid of the GRG and CubeWalk (GRG&CubeWalk) solver which was developed to improve the overall speed of the CubeWalk procedure. The hybrid GRG&CubeWalk solver uses GRG to solve the problem to a local minima and then uses CubeWalk to diversify the search away from that local minima. GRG is then used to find another local minima. The process is repeated until CubeWalk cannot find a better solution than the local minima obtained by GRG. The Frontline Systems Premium Solver Version 6.5, GRG solver is used. To test the solvers, data from the Nikkei 225 (Japan), and was used in Beasley, Meade, and Chang (2003) (available from OR-Library http://people.brunel.ac.uk/~mastjjb/jeb/info.html). The data set includes all weekly prices from March 1992 to September 1997 and includes 225 stocks . The index and index-plus-alpha problems were solved with 41 weeks of tracking data and the portfolios were restricted to 50 stocks. Four different problems were created by using different intervals within the available data set, that is 40^{th} (0-40), 80^{th} (40-80), 120^{th} (80-120), 160^{th} (120-160) weeks. Table 1 shows the computational results for the index fund problem, where the objective function uses fixed returns as explained previously. The RH(GRG) solver uses 7 iterations (reducing stocks from 225, 170, 130, 100, 80, 60, and 50 stocks) for each instance. The RH(GRG)&CubeWalk produces solutions that are on average 96.9% better than RH(GRG). The time required to run each trial of the CubeWalk procedure takes several minutes, depending on the problem. .

Table 1. Reoptimization Heuristic using GRG and The RH(GRG) and CubeWalk Hybrid**.**

Method	**Period (weeks)**	**40th week 0-40**	**80th week 40-80**	**120th week 80-120**	**160th week 120-160**
RH(GRG)	Objective function value	1.42E-4	1.37E-4	4.66E-5	7.61E-4
	No. of Stocks	50	50	49	50
RH(GRG)&CubeWalk	Objective function value	1.39E-9	1.52E-9	1.03E-9	3.47E-5
	No. CubeWalk trials	4	2	4	4
	No. of Stocks	50	50	50	50

Table 2 shows the computational result for the index tracking problems with 0.2% plus-alpha profit every week and without restricting the number of stocks. This plus alpha value provides a very difficult problem for GRG to solve. The RH(GRG) fails to produce an improved solution for of the 120th and 160th week problems. The GRG with a starting solution from CubeWalk works very well for plus-alpha index tracking problems.

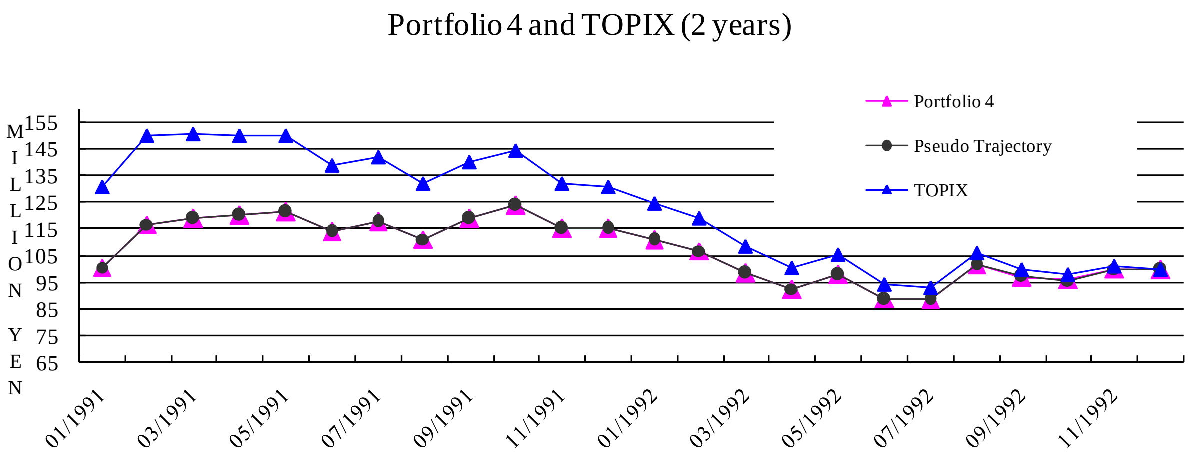
Table 2: Computational results for the RH(GRG) solving index tracking problem with 0.2% plus-alpha weekly profit.

Initial sol.		40th week 0 - 40	80th week 40 - 80	120th week 80 - 120	160th week 120 – 160
All 0.0001	Objective function value	0.0863	0.0898	14.9634	4.1244
	No. of stocks[1]	27, 56, 98	16, 84, 181	-	-
All 1/225	Objective function value	0.0250	0.0751	14.9685	4.1181
	No. of stocks	32, 52, 85	16, 83, 182	-	-
Solution by CubeWalk	Objective function value	0.0184	0.0250	0.0140	0.0002
	No. of stocks	29, 60, 110	33, 57, 108	37, 54, 106	39, 70, 100

[1]The numbers are the number of stocks whose weights are larger than 0.01, 0.005 and 0.0001 respectively

To evaluate the CPU time for problem sizes, the CubeWalk is applied to a problem with a 1.0 % up-trend trajectory of the TOPIX 1440 stocks for 24 months. The 50 stock portfolio obtained (Portfolio 4) is illustrated in Figure 6. The tracking error of the portfolio is 0.012% and sum of weights is just 1.0. A personal computer (Pentium IV with a 3.2GHz processor and 1GB of memory) was used for these experiments. Using this computer the problem took several days to solve. Each cube problem can be solved in a few minutes but the convergence of the CubeWalk becomes very slow once the active risk reduces to approximately 1%. A solution with an active risk of 1% is obtainable within 1 hour.

Figure 6: An index-plus-alpha portfolio for TOPIX



4.2 Index-Plus-Alpha Fund Considering Trading Units and Transaction Cost

The CubeWalk is applied to monthly data (stock price on the last day of the month) from the Tokyo stock market (NIKKEI 225) from April 1994 to March 1996. The trading unit size is assumed to be 1000 stocks.

The first experiment treats the case that only the commission is considered as a transaction cost. The commission used is outlined in Table 3, and was supplied by a securities company in Japan.

Table 3: Commission.

Amount of Transaction a (million yen)	Commission (million yen)		
~ 0.5	0.0140a		
0.5 ~ 0.7	0.0110a	+	0.0015
0.7 ~ 1	0.0090a	+	0.0029
1 ~ 3	0.0085a	+	0.0034
3 ~ 5	0.0080a	+	0.0049
5 ~ 10	0.0068a	+	0.0109
10 ~ 30	0.0055a	+	0.0239
30 ~ 50	0.0025a	+	0.1139
50 ~	0.0010a	+	0.1889

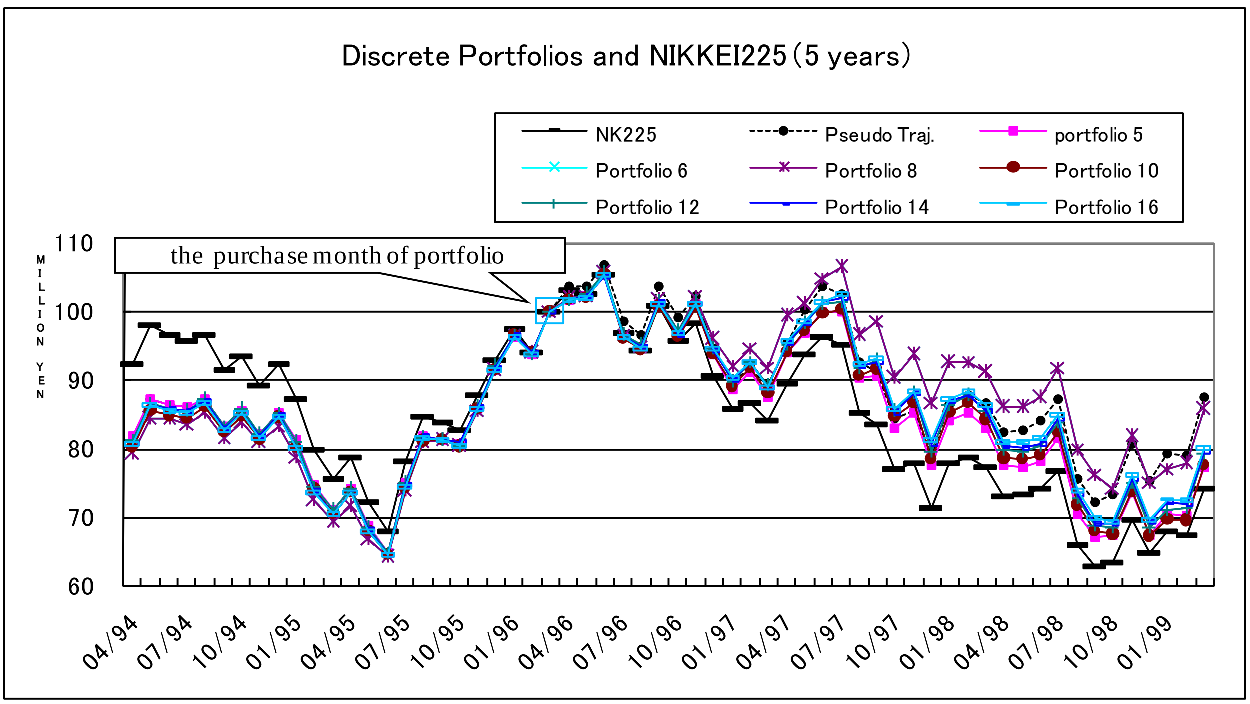
Computational results are summarized in Table 4 and Figure 7. Portfolio 5 provides a pseudo index without transaction cost or trading unit limitations. It is used as initial pivot solution for the cube walk and is also used as the first order approximation of $f_i^{(\ell)}(x_i, \rho, d)$. This is then used to generate the discrete portfolios 7-16 with trading unit limitations and different transaction cost rates. Portfolio 5 takes tens of minutes of CPU time to compute. Portfolios 7-16 are obtained in about ten seconds for generating each porfolio The portfolios are exact optimal solutions except for the precision lost through the size of the grid used. The total investment is 100 Million Yen. The maximum transaction cost rate ranges from θ = 0.95, 0.96, …, 1.05, as shown in Table 4. The larger the maximum rate θ we use, the more stocks that are included in the portfolio. From the viewpoint of having a more diversified investment, the portfolio obtained improves with more stocks. The active risk (the objective function) of the discrete portfolio changes from 0.0712% to 2.7597%, but the monthly tracking errors before the purchase are more stable, for example from 0.6031% to 0.8864%. The out of sample tracking errors are less than two times the error before the purchase. The maximum and average of ex-post returns vary from 8.94 to15.93% and from 3.97 to 9.17%, respectively. Figure 8 illustrates Portfolio 8 as an example. The actual discrete weights of portfolios are shown in Table 5.

Table 4: Discrete portfolios with commission.

Portfolio no.	Max. rate θ (%)	Trans. cost (%)	Tracking error (Objective value %)	Sum of weights	Variance (%) Before	Variance (%) Ex-post	No. of stocks	Relative difference[1] (%) Maximum	Relative difference[1] (%) Mean
5		1.08445	0.0009	1.00000	0.0069	0.8954	85	7.44	3.56
6	0.95	0.9494	2.7597	1.00010	0.3698	2.6452	34	15.93	9.17
7	0.96	0.9600	2.2636	1.00006	0.1836	1.8933	37	11.60	6.59
8	0.97	0.9698	0.8483	1.00012	0.1782	2.0025	39	13.00	7.28
9	0.98	0.9793	0.7673	1.00001	0.0939	1.3567	41	9.17	4.23
10	0.99	0.9899	0.6897	1.00006	0.0899	1.3149	43	8.94	3.97
11	1.00	0.9999	0.1975	1.00007	0.1251	1.5486	46	10.80	5.07
12	1.01	1.0100	0.0712	1.00005	0.0627	1.3898	49	10.18	4.98
13	1.02	1.0198	0.1118	1.00001	0.0708	1.1189	52	9.09	4.58
14	1.03	1.0300	0.2742	1.00010	0.0910	1.1800	55	10.21	5.17
15	1.04	1.0397	0.4658	1.00100	0.0959	1.4626	58	11.93	6.38
16	1.05	1.0484	0.3823	1.00000	0.0659	1.2720	62	10.42	5.53

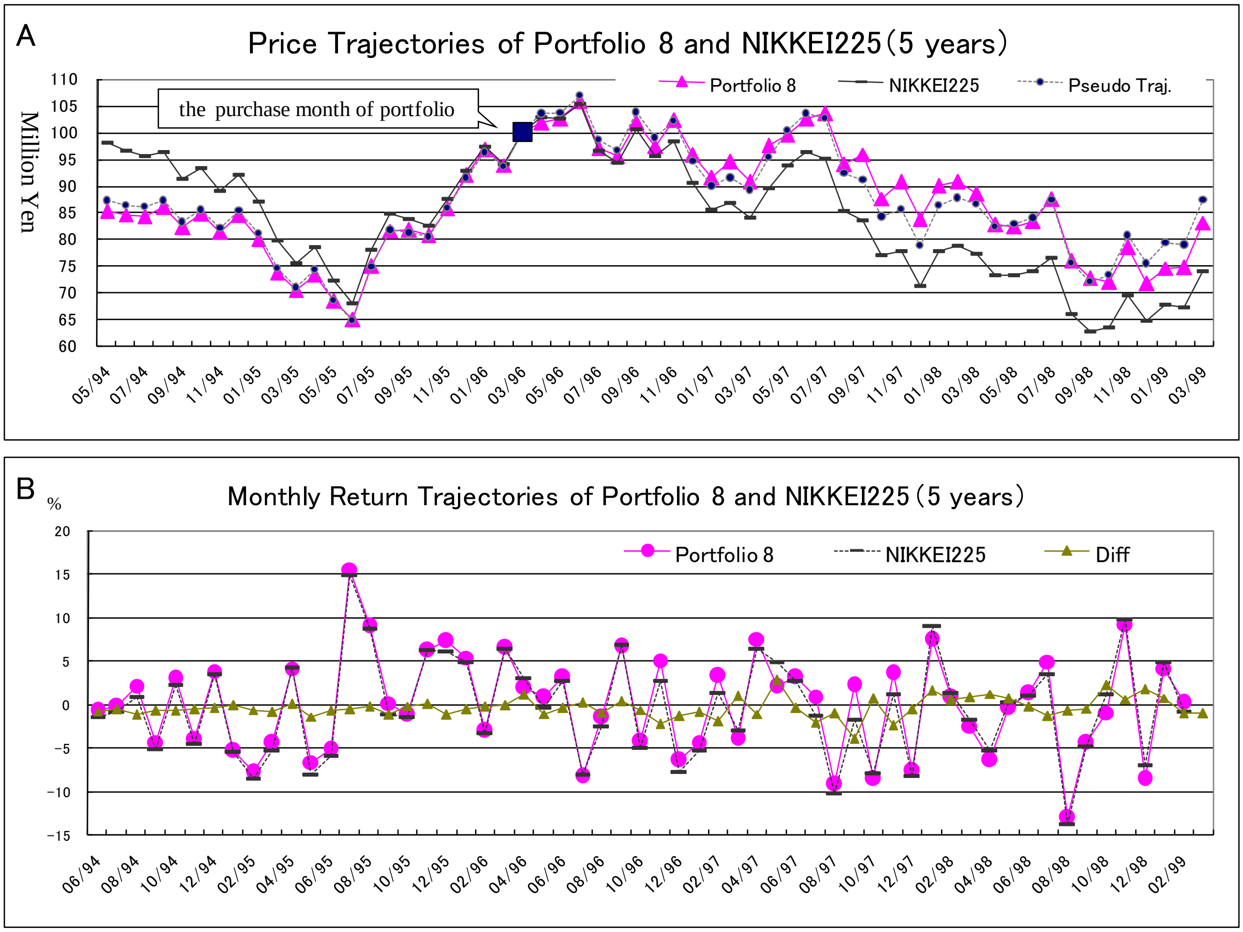
[1]Relative difference between out-of-sample return and the index as a percentage of the initial investment

Figure 7: Discrete portfolios and NIKKEI225 (5 years).



Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

Figure 8: Index-Plus-Alpha Tracking Portfolio 8 (24 month data used).



Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

Table 5: Portfolio 8

(stock code, weight, stock price at 03/96, number of lots (one lot is 1000 shares)).

Stock No.	1	2	3	4	5	6	7	8	9	10
Stock code	1301	1802	1812	4041	4042	4064	4092	4151	4501	4502
Weight	0.01618	0.03672	0.023	0.03004	0.01916	0.0258	0.0214	0.0204	0.049	0.0167
Stock price	809	918	1150	751	479	645	1070	1020	2450	1670
No. of stocks	2	4	2	4	4	4	2	2	2	1
Stock No.	11	12	13	14	15	16	17	18	19	20
Stock code	4503	4901	5331	5479	5707	5901	6103	6310	6461	6473
Weight	0.0238	0.0612	0.0105	0.0298	0.02139	0.0754	0.0121	0.01431	0.0232	0.0204
Stock price	2380	3060	1050	596	713	3770	1210	477	580	1020
No. of stocks	1	2	1	5	3	2	1	3	4	2
Stock No.	21	22	23	24	25	26	27	28	29	30
Stock code	6703	6758	6902	7011	7012	7202	7203	7231	7267	7951
Weight	0.01644	0.0639	0.0216	0.01848	0.01638	0.01238	0.0236	0.02044	0.0233	0.0195
Stock price	822	6390	2160	924	546	619	2360	511	2330	1950
No. of stocks	2	1	1	2	3	2	1	4	1	1
Stock No.	31	32	33	34	35	36	37	38	39	
Stock code	8252	8311	8315	8317	8402	8604	8802	9008	9202	
Weight	0.0233	0.0209	0.0226	0.0233	0.017	0.0235	0.0294	0.0314	0.0222	
Stock price	2330	2090	2260	2330	1700	2350	1470	628	1110	
No. of stocks	1	1	1	1	1	1	2	5	2	

Table 6: Execution cost (%).

amount of transaction (million)	1	2	…	210	211	212	…	219	…	225
	1301[1]	1331		9101	9104	9105		9501		9681
$10 \le c < 30$	0.607	0.000	…	1.680	1.840	0.221	…	1.080	…	0.000
$30 \le c < 50$	1.817	0.000	…	2.014	1.840	0.221	…	2.120	…	0.000
$50 \le c < 100$	1.817	0.000	…	2.014	2.386	0.221	…	4.090	…	0.000
$100 \le c < 200$	1.817	0.000	…	2.014	2.598	0.221	…	4.090	…	0.000
$200 \le c < 300$	1.817	0.000	…	2.014	2.598	1.601	…	5.100	…	0.000
$300 \le c < 400$	1.817	0.000	…	3.954	3.104	1.601	…	6.870	…	0.000
$400 \le c < 500$	2.937	0.000	…	3.954	3.104	3.081	…	8.800	…	0.000
$500 \le c < 600$	4.377	0.000	…	3.954	3.104	4.321	…	10.020	…	0.000
$600 \le c < 700$	4.377	0.000	…	3.954	3.461	4.321	…	11.490	…	0.000
$700 \le c < 800$	6.337	0.000	…	3.954	3.461	4.321	…	12.920	…	0.000
$800 \le c < 900$	8.227	0.000	…	5.824	5.081	4.321	…	12.920	…	0.000
$900 \le c < 1000$	8.227	0.000	…	7.174	5.081	4.857	…	12.920	…	0.000
$1000 \le c$	8.227	0.000	…	7.174	5.081	5.620	…	12.920	…	0.000

[1]means company code

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

Table 7: Discrete portfolios with execution costs and minimum trading units.

Portfolio no.	Max. rate θ (%)	Trans. cost (%)	Fixed time error (Objective value %)	Sum of weights	Periodic var. error (%) Before	Periodic var. error (%) Ex-post	No. of stocks	Relative difference[1] (%) Maximum	Relative difference[1] (%) mean
17	0.60	0.6000	0.4992	1.00000	0.3033	1.1474	62	17.74	10.74
18	0.64	0.6400	0.3254	1.00001	0.2820	1.1293	63	17.10	10.31
19	0.68	0.6800	0.2426	1.00000	0.2244	1.1027	67	16.33	9.65
20	0.72	0.7200	0.1686	1.00000	0.1654	1.0796	71	15.55	8.97
21	0.76	0.7600	0.1573	1.00000	0.1240	1.0470	75	14.66	8.31
22	0.80	0.8000	0.1097	1.00000	0.1897	1.0921	69	15.78	9.18
23	0.84	0.8400	0.0715	1.00000	0.1559	1.0965	73	15.13	8.67
24	0.88	0.8800	0.0535	1.00000	0.1314	1.0759	75	14.28	8.15
25	0.92	0.9200	0.0481	1.00000	0.1196	1.0539	81	13.50	7.57
26	0.96	0.9600	0.0460	1.00001	0.1020	1.0249	92	12.10	6.59
27	1.00	1.0000	0.0371	1.00000	0.1016	1.0095	98	11.62	6.22

[1]Relative difference between out-of-sample return and the index as a percentage of the initial investment

The second experiment assumes that the total investment is 5000 Million Yen. The commission and execution cost are considered as part of the transaction cost. The commission in Table 3 is used. The execution costs are generated randomly and are detailed in Table 6. One third of stocks (73 companies) are assumed to have execution costs such as market impact costs and market timing costs. Execution costs are increased by random numbers of interval [0.000, 2.000] and with 50% probability. Note that although these execution costs are discontinuous nonconvex, the CubeWalk can solve problems with the execution costs as the core solution engine of CubeWalk is the ISC that can effectively solve non-monotone non-convex discrete optimization problems (see Nakagawa, James, Rego 2007). Table 7 shows the performance of Portfolios 17-27. Naturally the Fixed Time errors (optimal objective function values) of Portfolios 17-27 are better than those of Portfolios 6-16 due to the larger portfolios a larger investment (as shown in Table 8) can provide. The in-sample performances of Portfolios 17-27 are also more stable than those of Portfolios 6-16.

Author: *Y. Nakagawa, R. J.W. James, C. Rego, F. Glover*
Kansai University Repository, Oct. 2010

Figure 9: Portfolio 18.

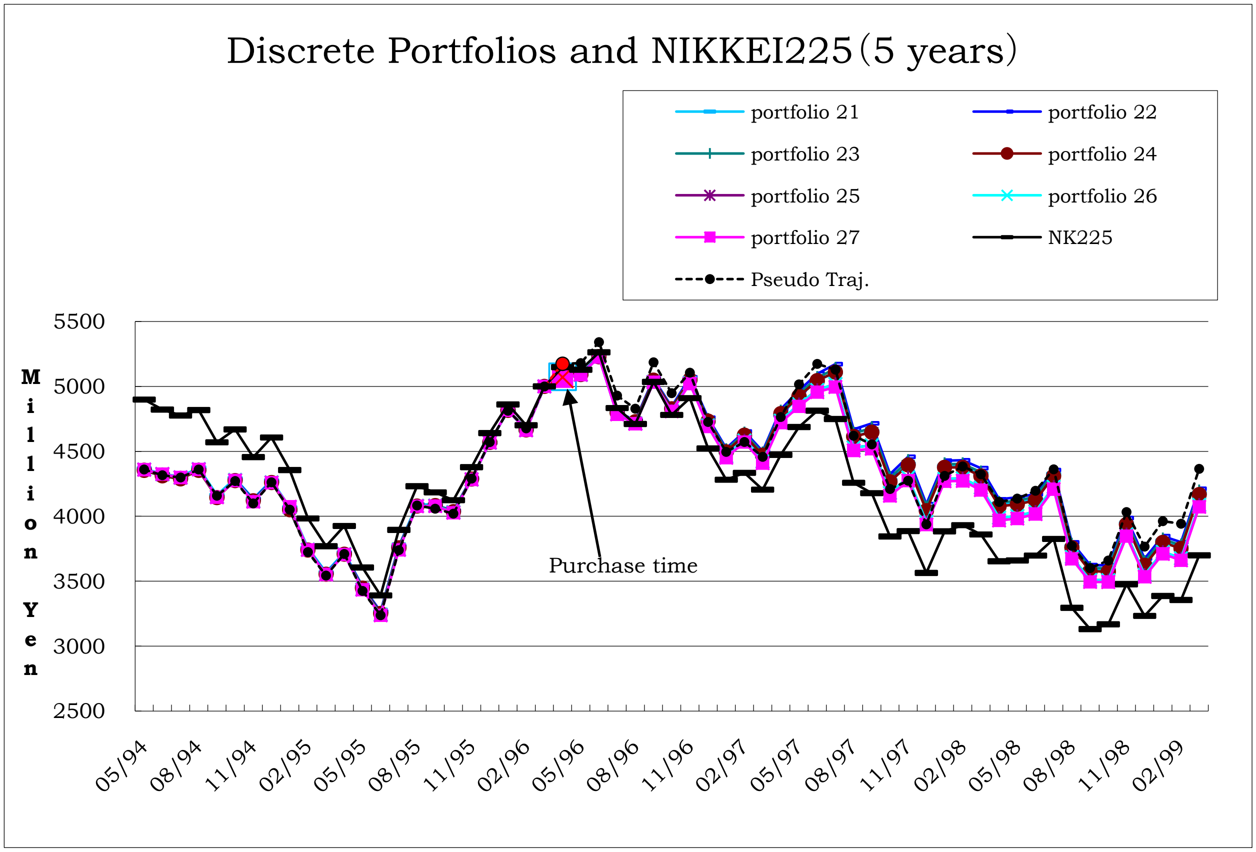
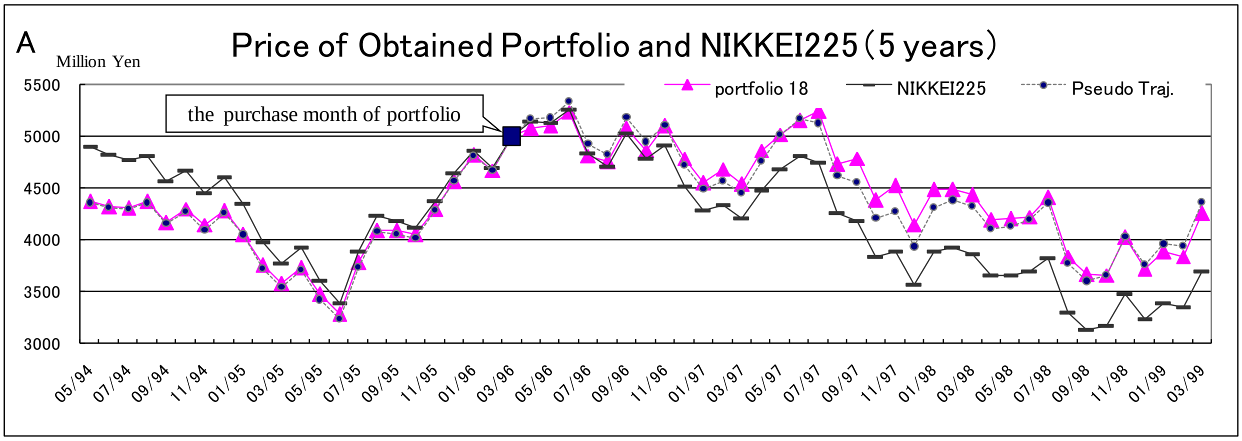


Figure 10: Portfolio 18.


A
Price of Obtained Portfolio and NIKKEI225 (5 years)
Million Yen
portfolio 18
NIKKEI225
Pseudo Traj.
the purchase month of portfolio
5500
5000
4500
4000
3500
3000
05/94
07/94
09/94
11/94
01/95
03/95
05/95
07/95
09/95
11/95
01/96
03/96
05/96
07/96
09/96
11/96
01/97
03/97
05/97
07/97
09/97
11/97
01/98
03/98
05/98
07/98
09/98
11/98
01/99
03/99

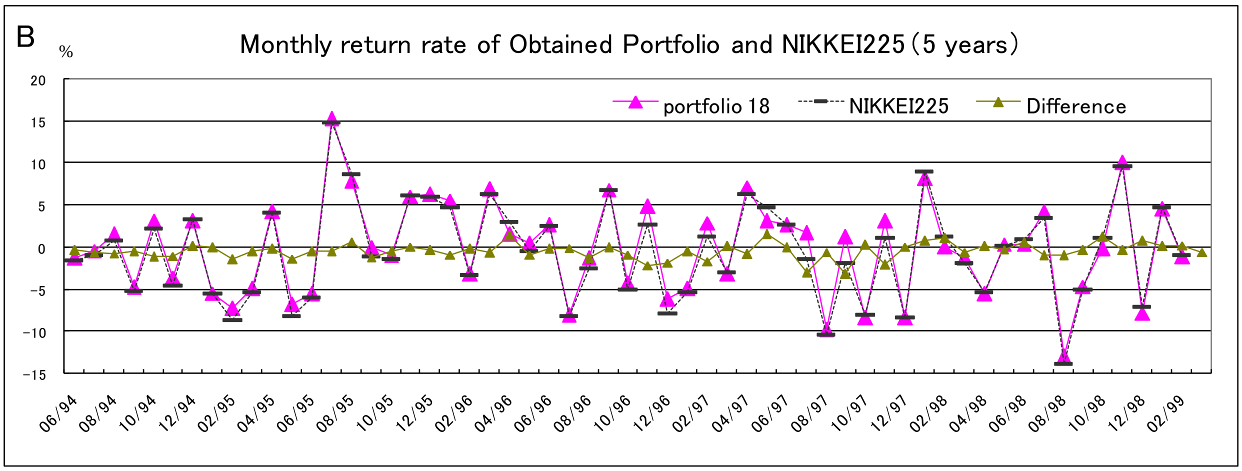

B
Monthly return rate of Obtained Portfolio and NIKKEI225 (5 years)
%
portfolio 18
NIKKEI225
Difference
20
15
10
5
0
-5
-10
-15
06/94
08/94
10/94
12/94
02/95
04/95
06/95
08/95
10/95
12/95
02/96
04/96
06/96
08/96
10/96
12/96
02/97
04/97
06/97
08/97
10/97
12/97
02/98
04/98
06/98
08/98
10/98
12/98
02/99

Table 8: Portfolio 18 (stock price at 03/96).

Stock No.	1	2	3	4	5	6	7	8	9	10
Stock code	1301	1802	1803	2001	2108	2501	2502	2531	3110	3403
Weight	0.005987	0.029743	0.029718	0.007258	0.008832	0.02495	0.024108	0.008658	0.008614	0.011214
Stock price	809	918	1170	648	640	998	1230	1170	365	630
No. of stocks	37	162	127	56	69	125	98	37	118	89
Stock No.	11	12	13	14	15	16	17	18	19	20
Stock code	3404	3863	3865	4041	4042	4063	4064	4092	4208	4401
Weight	0.007396	0.014955	0.00416	0.015471	0.029985	0.017304	0.010578	0.00963	0.006293	0.021956
Stock price	451	763	1040	751	479	2060	645	1070	414	963
No. of stocks	82	98	20	103	313	42	82	45	76	114
Stock No.	21	22	23	24	25	26	27	28	29	30
Stock code	4501	4502	4503	4901	5301	5331	5351	5405	5479	5707
Weight	0.02548	0.009018	0.039508	0.040392	0.00799	0.01113	0.010384	0.008911	0.005483	0.005989
Stock price	2450	1670	2380	3060	579	1050	1180	335	596	713
No. of stocks	52	27	83	66	69	53	44	133	46	42
Stock No.	31	32	33	34	35	36	37	38	39	40
Stock code	5721	5901	6103	6461	6473	6474	6479	6503	6758	6773
Weight	0.009108	0.046748	0.012342	0.014384	0.020604	0.008772	0.008235	0.019582	0.03834	0.01452
Stock price	660	3770	1210	580	1020	510	915	796	6390	2200
No. of stocks	69	62	51	124	101	86	45	123	30	33
Stock No.	41	42	43	44	45	46	47	48	49	50
Stock code	6841	6902	7011	7012	7202	7203	7231	7267	7751	7912
Weight	0.012544	0.024192	0.001848	0.011575	0.011885	0.02124	0.020031	0.027494	0.02652	0.01287
Stock price	1120	2160	924	546	619	2360	511	2330	2040	1950
No. of stocks	56	56	10	106	96	45	196	59	65	33
Stock No.	51	52	53	54	55	56	57	58	59	60
Stock code	7951	8252	8311	8315	8317	8318	8320	8322	8604	8802
Weight	0.01989	0.013048	0.015466	0.010848	0.011184	0.009936	0.01849	0.004096	0.0141	0.007938
Stock price	1950	2330	2090	2260	2330	2160	2150	1280	2350	1470
No. of stocks	51	28	37	24	24	23	43	16	30	27
Stock No.	61	62	63							
Stock code	9008	9009	9681							
Weight	0.024115	0.008568	0.0184							
Stock price	628	1190	2000							
No. of stocks	192	36	46							

5. Conclusions

In this paper non-convex quadratic programming problems have been used to solve two different types of portfolio optimization problems. Previously index tracking optimization problems were considered to be quite difficult to solve when there are non-convex transaction costs and lot size constraints for variables, since the problem is a discrete non-convex quadratic optimization problem. However using the techniques proposed in this paper we have shown that it is possible to solve practical problems of this type efficiently. How effective portfolios generated by this method are in terms of their out-of-sample returns is a topic for future research Other kinds of non-convex programming problems will be able to be solved in the future using techniques similar to this. This paper determined the cube size empirically, hence a technique for determining the size systematically is a topic for future research.

Acknowledgments

The authors would like to thank Yoshio TABATA, Osaka University, and Yoshitaka KAI , Kwanseiga-kuin University, and , N.C.P. Edirisinghe, University of Tennessee for their contributions to this paper.

References

Beasley, J.E., N. Meade, T.-J. Chang. 2003. An evolutionary heuristic for the index tracking problem. European Journal of Operational Research. 148 621–643.

Bonami, P., L.T. Biegler, A.R. Conn, G. Cornuejols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N.Sawaya, A. Waechter. 2005. An Algorithmic Framework for Convex Mixed Integer Non-linear Programs. IBM Research Report RC23771.

Bonami, P., J. Lee. 2006. BONMIN users' manual.

Chang T.-J., N. Meade, J.E. Beasley, Y.M. Sharaiha. 2000. Heuristics for cardinality constrained portfolio optimization, Computers & Operations Research 27 1271-1302.

Chu, P. C. , J. E.. Beasley. 1998. A Genetic Algorithm for the Multidimensional Knapsack Problem. Journal of Heuristics 4 63–86.

Coit,D. W., A. E.Smith. 1996a.. Reliability optimization of series-parallel systems using a genetic algorithm. IEEE Transactions on Reliability. 45(2) .254-260.

Coit, D.W., A. E. Smith. 1996b. Adaptive penalty methods for genetic optimization of constrainted Combinatorial problems. INFORMS J. on Computing 8 173-182.

Edirisinghe, V, C.. Naik, and R. Uppal. 1993. Optimal replication of options with transactions costs and trading restrictions, Journal of Financial and Quantitative Analysis, 28, 117–138.

Fyffe, D.E. Hines,W.W. and Lee,N.K. 1968. System reliability allocation and a computation algorithm. IEEE Transactions on Reliability. 17 64-69.

Gaivoronski A.A., S. Krylov, N. van der Wijst. 2005. Optimal portfolio selection and dynamic bench-mark tracking. European Journal of Operational Research 163 115-131.

Glover F. 1965. A Multiphase-Dual algorithm for the Zero-One Integer Programming Problem, Oper. Res. 13 879-919.

Glover F. 1968. Surrogate constraints. Oper. Res. 16 741-749.

Griffith, R.E., R.A. Stewart. 1961. A nonlinear programming technique for the optimization of continuous processing systems. Management Science 7 379-392.

Hikita M., Y. Nakagawa, K. Nakashima, H. Narihisa. 1992. Reliability optimization of systems by a surrogate-constraints algorithm. IEEE Transactions on Reliability. R-41 473-480.

Jobst N.J., M.H. Horniman, C. Lucas, G. Mitra. 2001. Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints, Quantitative Finance 1 489-501.

Kai, Y., Y. Nakagawa, Y. Tabata. 2005. Using an improved surrogate constraints method to non-separable non-convex programming problems. IEICE Transactions on Fundamentals, J88-A, 422-424 (in Japanese).

Kanzow, C., C. Nagel, H. Kato, M. Fukushima. 2005. Successive linearization methods for nonlinear semidefinite programs. Computational Optimization and Applications 31 251-273.

Kim, H., Bae, C. and Park, S. 2004. Simulated annealing algorithm for redundancy optimization with multiple component choices. Advanced Reliability Modeling, Proceedings of the 2004 Asian International Workshop, World Scientific, 237-244

Konno, H., H.Yamazaki. 1991. Mean-absolute deviation portfolio optimization model and its application to Tokyo stock market. Management Sci. 37 519–531.

Konno, H., A.Wijayanayake. 2001. Portfolio optimization problems with concave transaction costs and minimal transaction unit constraints. Math. Program. 89 233–250

Konno H., T.Hatagi, 2003. Index-Plus-Alpha tracking under concave transaction cost. Working Paper [2003-07]. Department of Industrial and Systems Engineering, Chuo University.

Kulturel-Konak, S., A.Smith, D. Coit, 2003. Efficiently solving the redundancy allocation problem using tabu search. IIE Transactions, 35(6) 515-526.

Liang, Y-C., Smith, A. 2004. An ant colony optimization algorithm for the redundancy allocation problem (RAP). IEEE Transactions on Reliability, 53, 417-423.

Markowitz, H., 1952 Portfolio Selection, Journal of Finance, 7, 77-91.

Marsten R. E., T. L. Morin. 1977. A hybrid approach to discrete mathematical programming. Math. Program. 14 21-40.

Mine H. (1959) Reliability of physical system, Transactions International. Symposium Circuit and Information Theory.

Moskowitz F., McLean, J.W. (1956) Some reliability aspects of system design, IRE Trans. Qual. Contr. RQC-8, 7-35.

Nakagawa, Y. , S. Miyazaki.1981. Surrogate constraints algorithm for reliability optimization problems with two constraints. IEEE Transactions on Reliability. 30 175-180.

Nakagawa, Y., M. Hikita, H. Kamada. 1984. Surrogate constraints algorithm for reliability optimization problems with multiple constraints. IEEE Transactions on Reliability. R-33 301-305.

Nakagawa, Y. 1998. A reinforced surrogate constraints method for separable nonlinear integer programming. RIMS 1068, Kyoto University. 194-202.

Nakagawa, Y. 2003. An improved surrogate constraints method for separable nonlinear integer programming. Journal of Operations Research Society of Japan. 46 145-163.

Nakagawa, Y. 2004. A heuristic method for multi-constraint separable discrete optimization. IEICE Trans. Fundamentals J87-A 690-693. (in Japanese).

Nakagawa, Y., Y. Kai, Y. Tabata, R.J.W. James. 2004. Nonlinear discrete programming for solving the combinatorial and continuous index plus alpha fund optimization. ICCOPT-I (WA 4).

Nakagawa, Y., Y. Isada, R.J.W. James. 2005a. Interactive improved surrogate constraints method for nonconvex portfolio optimization problems. Working Paper [2005-01]. Faculty of Informatics, Kansai University.

Nakagawa, Y., Y. Isada, 2005. A branch-and-difficulty for solving multidimensional separable nonlinear knapsack problem. IEICE Trans. Fundamentals, J88-A 1002-1005 (in Japanese).

Nakagawa Y., Y. Isada, R.J.W. James. 2005b. A near-optimal method for solving multidimensional nonlinear knapsack problems. Working Paper [2005-03] Faculty of Informatics, Kansai University.

Nakagawa Y., R.J.W. James, C. Rego. 2007. Entropy of problem difficulty and improved surrogate constraint method for multidimensional nonlinear knapsack problems. submitted to Management Science.

Nomura Institute of Capital Markets Research, 2002. The stocks operation and investment strategy, Kinzai Institute for Financial Affairs, Inc.(in Japanese)

Ohnishi J., S. Kimura, R.J.W. James, Y. Nakagawa .2007. Solving the redundancy allocation problem with a mix of components using the improved surrogate constraint method, IEEE Trans. Rel.,56 94-101.

Palacios-Gomez, F. , L. Lasdon, M. Engquist. 1982. Nonlinear optimization by successive linear programming. Management Science 28 1106-1120.

Perold A. F. 1984. Large-scale portfolio optimization. Management Sci. 30 1143–60

Premium Solver Platform User Guide. 2005. Frontline Systems, Inc.

Sahinidis, N. V., M. Tawarmalani [2005] BARON 7.2.5: Global Optimization of Mixed-Integer Nonlinear Programs, User's manual.

Tawarmalani, M., N. V. Sahinidis [2004] Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. Mathematical Programming, 99 563-591.

Tabata, Y., E. Takeda. 1995. Bicriteria optimization problem of designing an index fund. Journal of the Operations Research Society. 46 1023-1032.

Vasquez, Michel. and Hao, Jin-Kao 2001. A Hybrid Approach for the 0-1 Multidimensional Knapsack Problem. Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence. IJ-CAI 2001, Seattle, Washington, USA, 328– 333.

Vasquez, Michel. and Vimont, Yannick. 2005. Improved results on the 0-1 Multidimensional Knapsack Problem.. European Journal of Operational Research. 165 70-81.

You P.S. , T.C.Chen. 2005. An efficient heuristic for series-parallel redundant reliability problems. Computer and Operations Research, 32, 2117-2127.

Zhang, J. N.-H. Kim, L. Lasdon. 1985. An improved successive linear programming algorithm. Management Science 31 1312-1331.