

# Portfolio Optimization: New Capabilities and Future Methods

Ralph E. Steuer\* and Yue Qi  
Terry College of Business  
University of Georgia  
Athens, Georgia 30602-6253 USA

and

Markus Hirschberger  
Department of Mathematics  
University of Eichstätt-Ingolstadt  
Eichstätt, Germany

November 17, 2005

## Abstract

- In this paper we present what we feel is a superior capability for computing the “efficient frontier” (henceforth called the “nondominated frontier”) of many portfolio selection problems in finance. The computer capability is designated MPQ (for multi-parametric quadratic programming) and has been programmed in Java.
- MPQ possesses several advantages. One is that MPQ is useful on large-scale applications (up to at least 2,000 securities in mean-variance optimization). Another is that MPQ runs in reasonable time with dense covariance matrices thus obviating the need in many cases to diagonalize the covariance matrix structure for CPU-time purposes. A

---

\*Corresponding author [rsteuer@uga.edu](mailto:rsteuer@uga.edu)

third is that instead of solving an  $\varepsilon$ -constraint formulation repetitively to obtain a dotted or piecewise linear representation of a mean-variance nondominated frontier, MPQ can solve for the exact nondominated frontier in usually only a fraction of the time.

- Furthermore, by utilizing a multi-parametric part of MPQ, the computer capability also shows promise for computing the nondominated surfaces of portfolio problems with additional linear objectives such as dividends, liquidity, social responsibility, amount invested in R&D, and so forth.

**Keywords:** Portfolio selection, efficient frontiers, piecewise parabolic segments,  $\varepsilon$ -constraint methods, diagonalizing the covariance matrix structure, dense covariance matrices, quadratic multiple criteria optimization, paraboloidic platelets.

## 1 Introduction

Because of the degree to which the effectiveness of an economy rests upon its ability to most beneficially allocate its capital, no stone should be left unturned in trying to better understand the investment process. In this vein, our attention is drawn to the problem of portfolio selection in that much of what is “modern portfolio theory” (see Elton, Gruber, Brown and Goetzmann [8]) is closely linked to the efficient frontiers (but what we will call “nondominated frontiers”) of such problems.

With portfolio selection a problem at the intersection of finance and operations research (that is, quadratic programming and multiple criteria optimization), our purposes in this paper are to

- (a) survey the popular software package situation with regard to the computation of the nondominated frontiers of mean-variance portfolio selection problems
- (b) introduce a high-speed capability for computing mean-variance nondominated frontiers of large-scale problems
- (c) show how additional objectives such as dividends, liquidity, social responsibility, growth in sales, and so forth, can be incorporated into a theory of multiple objective portfolio selection

- (d) discuss the types of methods necessary for solving portfolio optimization problems that also possess additional objectives.

Our motivation with regard to item (a) stems from the current software situation as represented, for example, by Cplex [6], LINGO [35, 36], Matlab [29], Mathematica [41], Optimizer by Markowitz and Todd [27], and premium versions of Solver [40]. These packages presumably cover the tools used by most portfolio optimization users. However, only Optimizer is able to compute an exact nondominated frontier<sup>1</sup>, and only Optimizer is written for the public domain. Written in VBA (Visual Basic for Applications), this package's only drawback is that it is only able to run with a maximum of 248 securities as a consequence of the 256-column limitation of Excel.

The difficulties with the other packages, all of which are commercial, are several. Of course, each comes with a price tag. This is understood. But for your money, what is frustrating is that not one of the commercial packages enables the creation of an executable. This is inhibiting to research as nothing written on any of the packages can be written for the public domain. This means that a routine developed on one researcher's machine cannot in general be run on another researcher's machine without another purchased copy of the software installed on the other machine.

What is also a major disappointment, and this gets at item (b), is the computer performance of the packages on portfolio problems. Not being able to compute an exact nondominated frontier, the best they can do is to produce a dotted or piecewise linear representation of a nondominated frontier. Requiring repetitive optimizations, the CPU-times required to generate such representations quickly become onerous on problems with more than about 600-800 securities. About the only way to avoid severe CPU-times on large problems with the packages is to somehow simplify the covariance matrix structure, such as by diagonalizing it as in Markowitz and Perold [26]. Since such covariance matrix simplifications typically involve a loss of information, what then results is a dotted or piecewise linear representation of the nondominated frontier of an approximation of the original problem.

In contrast to the above packages, the computerized capability MPQ of Hirschberger, Qi and Steuer [13] offers the following advantages. MPQ (i) is written for the public domain for academic researchers, (ii) is written in Java

---

<sup>1</sup>Actually, LINDO, by means of its PARA command, is able to compute an exact nondominated frontier. However, LINDO is being replaced by LINGO, and the PARA command is not, as of this writing, included in LINGO.

for transportability, (iii) does not involve repetitive optimizations and (iv) can compute, without any necessity to simplify dense covariance matrices, the exact nondominated frontiers of problems with up to 2,000 securities. Moreover, the time taken by MPQ to compute an exact nondominated frontier is usually dramatically less than the time required by any of the commercial packages to compute a dotted or piecewise linear representation of a nondominated frontier.

Our motivation with regard to item (c) stems from the work of Aouni, Ben Abdelaziz and El-Fayedh [1], Arenas Parra, Bilbao Terol and Rodríguez Uría [2], Bana e Costa and Soares [3], Ehrgott, Klamroth and Schwehm [7], Guerard and Mark [11], Hallerbach and Spronk [12], Lo, Petrov and Wierzbicki [20], Ogryczak [32], Zopounidis and Doumpos [42], and others, involving multi-attribute portfolio selection. Since its inception in the 1950s by Markowitz [24, 25], portfolio selection has focused almost exclusively on the first two moments of the single random variable of portfolio return. However, in many of today’s more complex environments, portfolio managers are increasingly interested in monitoring other measures such as mentioned in (c) in the management of their portfolios. In this connection, we show how multiple objectives are able to enter the general theory of portfolio selection and how the multiple objectives should then be modeled for solution.

With regard to item (d) the goal is to be able to solve portfolio problems that possess multiple objectives. This involves understanding the nature of what is no longer a nondominated “frontier,” but a nondominated “surface.” Then the potential of MPQ to compute such nondominated surfaces is outlined along with strategies for searching a nondominated surface for a most preferred point on it. Preliminary computational experience is reported for certain problems with up to 400 securities.

The paper is organized as follows. In Section 2 we provide portfolio selection theory basics and in Section 3 we discuss the benefits of framing portfolio selection as a multiple criteria optimization problem. In Section 4 we review the properties of portfolio problems that can be solved for in closed form. In Section 5 we discuss the computation of nondominated frontiers of problems that require mathematical programming and report CPU-times. In Section 6 we show how the theory of mean-variance portfolio selection can be broadened to include additional objectives. Section 7 discusses future methods with regard to the solution of multiple objective portfolio selection problems, and Section 8 consists of concluding remarks.

## 2 Portfolio Theory Basics

The well-known problem of portfolio selection is as follows. Assume

- (a)  $n$  securities
- (b) an initial sum to be invested
- (c) a beginning of a holding period
- (d) an end of the holding period

Let  $\mathbf{x} = (x_1, \dots, x_n)$  be termed an *investment proportion vector*. We also call  $\mathbf{x}$  a *portfolio* as it specifies the proportions of the initial sum to be invested in the  $n$  securities at the beginning of the holding period that are to be held fixed until the end of the holding period. With expected value  $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)$  and  $n \times n$  covariance matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & & \\ \vdots & & & \vdots \\ \sigma_{n1} & & \cdots & \sigma_{nn} \end{bmatrix},$$

let  $\mathbf{r} = (r_1, \dots, r_n)$  specify the percent returns of the  $n$  securities to be realized over the holding period. Note that the  $r_i$  are random variables. Thus for a given  $\mathbf{x}$ , *portfolio return*

$$R(\mathbf{x}, \mathbf{r}) = \sum_{i=1}^n r_i x_i = \mathbf{r}^T \mathbf{x}$$

that is, the percent return to be earned on the portfolio over the course of the holding period, is a random variable. The reason to use  $R(\mathbf{x}, \mathbf{r})$  as notation for the portfolio return random variable is that both  $\mathbf{r}$  and  $\mathbf{x}$  are needed to define it. Note that the expected value and variance of  $R(\mathbf{x}, \mathbf{r})$  are given deterministically by

$$E[R(\mathbf{x}, \mathbf{r})] = \sum_{i=1}^n \mu_i x_i = \boldsymbol{\mu}^T \mathbf{x}$$

and

$$V[R(\mathbf{x}, \mathbf{r})] = \sum_{i=1}^n \sum_{j=1}^n x_i \sigma_{ij} x_j = \mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x}$$

Under the assumptions that security prices accurately reflect value and that the investor’s *overall focus*<sup>2</sup> in portfolio selection is to singularly pursue the goal of wealth maximization, the problem of portfolio selection is to maximize the portfolio return random variable as in

$$\begin{aligned} \max\{R(\mathbf{x}, \mathbf{r}) = \mathbf{r}^T \mathbf{x}\} \\ \text{s.t. } \mathbf{x} \in S \end{aligned} \tag{1}$$

where  $S$  is the set of all feasible investment proportion vectors. In general,

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{1}^T \mathbf{x} = 1, \alpha_i \leq x_i \leq \omega_i\} \tag{2}$$

where the  $\alpha_i$  and  $\omega_i$  are lower and upper bounds on the  $x_i$ . However, in many problems, it is not uncommon for  $\mathbf{A}$  to be vacuous. As its purpose is to reflect the investor’s overall focus, we will refer to (1) as the investor’s *stochastic reflection program*.

While (1) may look like a linear program, it is not a linear program. The difficulty with (1) is that the  $r_i$  are not known until the end of the holding period, but the  $x_i$  must be chosen at the beginning of the holding period. This is why the word “stochastic” has been included in the nomenclature for (1) – and how to solve a program with stochastic properties requires a decision. The decision involves how to replace the stochastic program, in this case the stochastic reflection program, with a deterministic one for solution. In the spirit of Caballero, Cerdá, Muñoz, Rey and Stancu-Minasian [4] and Steuer, Qi and Hirschberger [38], an ideal replacement problem is what we will call an *equivalent deterministic implementation problem* in the sense that all solutions that solve the equivalent deterministic implementation problem are potentially optimal in the stochastic program, and vice versa. We realize that the names given to the programs are a bit ponderous, but they should help with clarity as we proceed through the paper.

One of the contributions of Markowitz is his choice of

$$\begin{aligned} \min\{\mathbf{x}^T \Sigma \mathbf{x}\} \\ \text{s.t. } \boldsymbol{\mu}^T \mathbf{x} = \rho \quad \rho \in [a, b] \\ \mathbf{x} \in S \end{aligned} \tag{3}$$

---

<sup>2</sup>in the same sense that Saaty, for example in [34], uses the concept in his works

as the equivalent deterministic implementation problem<sup>3</sup> for (1) where  $[a, b]$  is the smallest interval by which (3) can generate the nondominated frontier. For a given  $\rho \in [a, b]$ ,  $\mathbf{x}^\rho$  solves (3) if random variable  $R(\mathbf{x}^\rho, \mathbf{r})$  has the smallest variance of all portfolio return random variables whose expected values equal  $\rho$ . In (variance, expected-return) space,  $((\mathbf{x}^\rho)^T \boldsymbol{\Sigma} \mathbf{x}^\rho, \boldsymbol{\mu}^T \mathbf{x}^\rho)$  is the nondominated point associated with  $\rho$ . Thus the task of (3) is to compute the set of all nondominated points

$$\{(\mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x}, \boldsymbol{\mu}^T \mathbf{x}) \in \mathbb{R}^2 \mid \mathbf{x} \text{ solves (3) for some } \rho \in [a, b]\} \quad (4)$$

Since the graph of (4) is an upward-sloping curve as shown in Figure 1, we see the penchant for use of the term “frontier.” In the graph,  $\mathbf{z}^{nmv}$  signifies the nondominated point of minimum variance and  $\mathbf{z}^{nmer}$  signifies the nondominated point of maximum expected return. Thus for the  $a$  and  $b$  of (3),

$$a = z_2^{nmv} \quad \text{and} \quad b = z_2^{nmer}$$

where  $z_2^{nmv}$  is the second component of  $\mathbf{z}^{nmv} \in \mathbb{R}^2$  and  $z_2^{nmer}$  is the second component of  $\mathbf{z}^{nmer} \in \mathbb{R}^2$ .

The durability of Markowitz’s designation of (3) as an equivalent deterministic implementation problem is derived from the fact that it has since been justified under two sets of assumptions. One is that (3) is valid provided the investor’s utility function is a function of only the single argument  $R(\mathbf{x}, \mathbf{r})$  and that the utility function is increasing and quadratic in that argument. Another is that (3) is valid as long as the vector of security returns  $\mathbf{r}$  follows the multinormal distribution. What is remarkable is that the theory of mean-variance portfolio selection that took hold in the 1950s is still the prevailing theory of portfolio selection today. Part of the reason for its longevity is that to supplant an entrenched theory, one must come up with a better theory, and this has not been done. However, the theory of multiple criteria portfolio selection outlined later in this paper shows that the theory of present-day mean-variance analysis can at least be expanded to a superset of itself under many circumstances.

---

<sup>3</sup>manytimes one sees this formulation with the  $\rho$ -constraint written in inequality form  $\boldsymbol{\mu}^T \mathbf{x} \geq \rho$

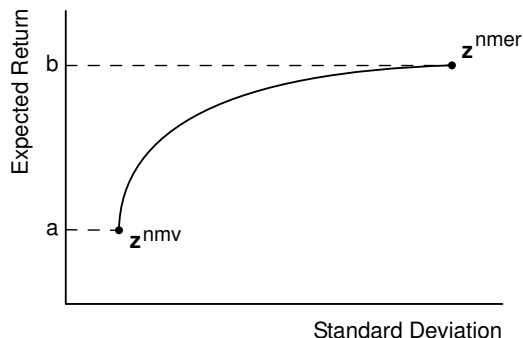


Figure 1: This is a typical nondominated frontier in (standard-deviation, expected-return) space. The last part of the portfolio process is to select a best point on the nondominated frontier and then identify an inverse image in  $S$  of the point as an optimal portfolio. Note that while theory and computation in portfolio selection are generally carried out in terms of variance, nondominated frontiers are most often shown to users with standard deviation on the horizontal axis.

### 3 Multiple Criteria Optimization

We now wish to convey the benefits of treating portfolio selection as a multiple criteria optimization problem (Ehrgott [9] is a recent reference). This is done because equivalent deterministic implementation problem (3) is recognized as a family of  $\varepsilon$ -constraint formulations whose purpose is to generate all nondominated points of the multiple criteria optimization problem

$$\begin{aligned}
 & \min\{ \mathbf{x}^T \Sigma \mathbf{x} \} \\
 & \max\{ \boldsymbol{\mu}^T \mathbf{x} \} \\
 & \text{s.t. } \mathbf{x} \in S
 \end{aligned} \tag{5}$$

Recall (see Steuer [37], Chap. 8) that in an  $\varepsilon$ -constraint formulation of a multiple criteria optimization problem, all but one of the objectives are converted to constraints with the  $\varepsilon_i$  right-hand sides of the constraints set to target values at the discretion of the user.

Formulation (5), in our opinion, is a more appropriate way to express the problem of mean-variance portfolio selection because it does not confound the problem with the method used to solve it – and the  $\varepsilon$ -constraint method is not the only way to solve for the set of all nondominated points. For



instance, the family of weighted-sums problems

$$\begin{aligned} \max \{ \lambda \boldsymbol{\mu}^T \mathbf{x} - \mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x} \} & \quad \lambda \in [0, \infty) \\ \text{s.t. } \mathbf{x} \in S & \end{aligned} \quad (6)$$

is also a valid way to solve for all mean-variance nondominated points.

As for terminology, consider the multiple criteria optimization problem

$$\begin{aligned} \max \text{ or } \min \{ f_1(\mathbf{x}) = z_1 \} & \quad (7) \\ \vdots & \\ \max \text{ or } \min \{ f_k(\mathbf{x}) = z_k \} & \\ \text{s.t. } \mathbf{x} \in S & \end{aligned}$$

in which  $k$  is the number of objectives and the  $z_i$  are *criterion values*. In single-criterion optimization there is the usual feasible region  $S$  in *decision space*  $\mathbb{R}^n$ . But when  $k > 1$ , it is useful to consider feasible region  $Z = \{ \mathbf{z} \in \mathbb{R}^k \mid \mathbf{z} = \mathbf{f}(\mathbf{x}), \mathbf{x} \in S \}$  in *criterion space*  $\mathbb{R}^k$ . In this way, each  $\mathbf{x} \in S$  in decision space has an *image*  $\mathbf{z} \in Z$  in criterion space, and each  $\mathbf{z} \in Z$  in criterion space has at least one *inverse image*  $\mathbf{x} \in S$  in decision space.

In a multiple criteria optimization problem, a *criterion vector*  $\mathbf{z} \in Z$  is either *nondominated* or *dominated*. Let  $J^+ = \{ i \mid f_i(\mathbf{x}) \text{ is to be maximized} \}$  and  $J^- = \{ j \mid f_j(\mathbf{x}) \text{ is to be minimized} \}$ . Then we have

**Definition 1** Let  $\bar{\mathbf{z}} \in Z$ . Then  $\bar{\mathbf{z}}$  is nondominated in (7) if and only if there does not exist another  $\mathbf{z} \in Z$  such that (i)  $z_i \geq \bar{z}_i$  for all  $i \in J^+$ , and  $z_j \leq \bar{z}_j$  for all  $j \in J^-$ , and (ii)  $z_i > \bar{z}_i$  or  $z_j < \bar{z}_j$  for at least one  $i \in J^+$  or  $j \in J^-$ . Otherwise,  $\bar{\mathbf{z}} \in Z$  is dominated.

The set of all nondominated criterion vectors is called the *nondominated set*. Whereas vectors  $\mathbf{z} \in Z$  in criterion space are either nondominated or dominated, points  $\mathbf{x} \in S$  in decision space are either *efficient* or *inefficient* as follows.

**Definition 2** Let  $\bar{\mathbf{x}} \in S$ . Then  $\bar{\mathbf{x}}$  is efficient in (7) if and only if its image criterion vector  $\bar{\mathbf{z}} = (f_1(\bar{\mathbf{x}}), \dots, f_k(\bar{\mathbf{x}}))$  is nondominated. Otherwise,  $\bar{\mathbf{x}}$  is inefficient.

The set of all efficient points is called the *efficient set*. Notice how Definitions 1 and 2 enable us to define nondominance in criterion space and efficiency in decision space regardless of the number objectives or their min-max status.

A side benefit of using formulation (5) along with Definitions 1 and 2 is that no longer should  $\varepsilon$ -constraint formulations such as in (3) induce imprecise definitions of the nondominated frontier as seen in Jones

“An efficient portfolio has the highest expected return for a given level of risk, or the lowest level of risk for a given level of expected return” [18] p. 526

Mayo [30] p. 163, and other writings. The imprecision of the quote is demonstrated in Figure 2 in which the nondominated frontier is the curve between  $\mathbf{z}^{nmv}$  and  $\mathbf{z}^{nmer}$ , inclusive. For each of the horizontal dashed-line levels of expected return, the portfolios of lowest risk are inefficient, and for each of the vertical dashed-line levels of risk, the portfolios of highest expected return are also inefficient, thus contradicting the quote.

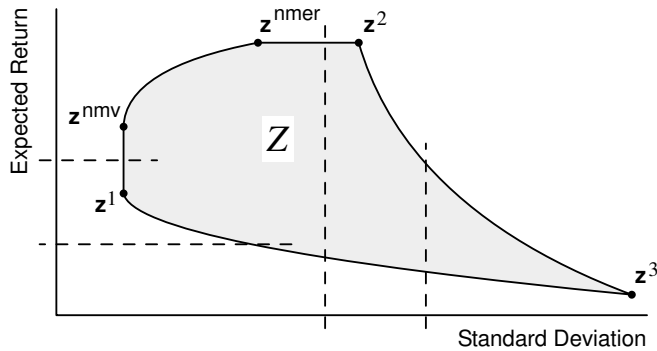


Figure 2: A troublesome portfolio-selection feasible region  $Z$  for imprecise definitions of the types of points that comprise the nondominated frontier.

To define optimality in multiple criteria optimization, let  $V: Z \rightarrow \mathbb{R}$  be the decision maker’s  $k$ -argument function (where the  $k$  arguments are the criterion values  $z_i$ ) to be maximized over  $Z$ . Then, any  $\mathbf{z}^o \in Z$  that maximizes  $V$  over  $Z$  is an *optimal criterion vector*, and any inverse image of  $\mathbf{z}^o$ , that is, any  $\mathbf{x}^o \in S$  such that  $(f_1(\mathbf{x}^o), \dots, f_k(\mathbf{x}^o)) = \mathbf{z}^o$ , is an *optimal solution*. We are interested in the efficient and nondominated sets because if  $V$  is such that *more-is-better-than-less* for each  $z_i, i \in J^+$ , and *less-is-better-than-more* for each  $z_j, j \in J^-$ , then any optimal criterion vector is a member

of the nondominated set, and any inverse image of an optimal criterion vector is a member of the efficient set. Since expected return is in the first category and variance is in the second, this means that to find an optimal portfolio one only needs to find a most preferred criterion vector in the nondominated set, and then take an inverse image. This is precisely Markowitz's protocol for identifying an optimal portfolio [24, 25, 27].

## 4 Computing Nondominated Frontiers I

We now focus on two categories of mean-variance portfolio selection models

$$\min \{ \mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x} = z_1 \} \quad (8.1)$$

$$\max \{ \boldsymbol{\mu}^T \mathbf{x} = z_2 \} \quad (8.2)$$

$$\text{s.t. } \mathbf{x} \in S \quad (8.3)$$

as distinctly different methods are pursued to obtain the nondominated frontier.

In the first category is the *invertible-covariance-matrix, unrestricted-variable* model. In this model,  $\boldsymbol{\Sigma}$  in (8.1) is invertible and  $S$  in (8.3) is given by

$$\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{1}^T \mathbf{x} = 1 \} \quad (9)$$

With no lower or upper bounds on any of the  $x_i$ , the model is susceptible to criticism about being unrealistic. Also, it is often hard for  $\boldsymbol{\Sigma}$  to be invertible as all covariance matrices derived from fewer than  $n$  observations are non-invertible. Nevertheless, the model has been of considerable academic interest in that its efficient and nondominated sets can be solved by formula as demonstrated in many places such as in Roll [33], Huang and Litzenberger [16], Ingersoll [17], Luenberger [22], and Campbell, Lo and Mackinlay [5].

Properties of invertible-covariance-matrix, unrestricted-variable models include the following.

- (a) Feasible region  $Z \subset \mathbb{R}^2$  in criterion space is unbounded to the right (in the direction of increasing variance or standard deviation).
- (b) The boundary of  $Z$  to the left, called the *minimum variance boundary*, is a parabola in (variance, expected-return) space – and a hyperbola in (standard-deviation, expected-return) space.

- (c) The nondominated frontier is the top half of the parabola, inclusive, in (variance, expected-return) space – and the top half of the hyperbola, inclusive, in (standard-deviation, expected-return) space
- (d) Let  $\mathbf{x}^1$  and  $\mathbf{x}^2$  be portfolios whose criterion vectors are on the minimum variance boundary. Then the criterion vectors of all linear combinations of  $\mathbf{x}^1$  and  $\mathbf{x}^2$  are also on the minimum variance boundary.
- (e) Because of the invertibility of  $\Sigma$ , each point on the nondominated frontier has one and only one inverse image in  $S$ .

To illustrate some of what can be computed by formula, form

$$\mathbf{T} = \begin{bmatrix} \boldsymbol{\mu}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu} & \boldsymbol{\mu}^T \boldsymbol{\Sigma}^{-1} \mathbf{1} \\ \mathbf{1}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu} & \mathbf{1}^T \boldsymbol{\Sigma}^{-1} \mathbf{1} \end{bmatrix} \equiv \begin{bmatrix} c & d \\ d & f \end{bmatrix}$$

in which  $\mathbf{1} \in \mathbb{R}^n$  is a vector of ones. This  $2 \times 2$  matrix is formed because  $\mathbf{T}$  itself, or elements thereof, are employed in many of the formulas. For instance,

- (a) the expected return of the minimum variance portfolio  $\mathbf{x}^{nmv}$  is given by

$$z_2^{nmv} = \frac{d}{f}$$

- (b) the variance component  $z_1^p$  of a point  $\mathbf{z}^p = (z_1^p, z_2^p)$  on the nondominated frontier whose expected return is fixed to be  $z_2^p$ , where  $z_2^p \geq z_2^{nmv}$ , is given by

$$z_1^p = [z_2^p \ 1] \mathbf{T}^{-1} \begin{bmatrix} z_2^p \\ 1 \end{bmatrix}$$

- (c) the (efficient) investment proportion vector  $\mathbf{x}^p$  whose image is the nondominated criterion vector  $\mathbf{z}^p$  is given by

$$\mathbf{x}^p = \boldsymbol{\Sigma}^{-1} [\boldsymbol{\mu} \ \mathbf{1}] \mathbf{T}^{-1} \begin{bmatrix} z_2^p \\ 1 \end{bmatrix}$$

Since all linear combinations of points in  $S$  whose criterion vectors are on the minimum variance boundary have criterion vectors on the minimum variance boundary, the set of all efficient points is given by

$$\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} = \mathbf{x}^{nmv} + \alpha(\mathbf{x}^p - \mathbf{x}^{nmv}), \alpha \geq 0\}$$

as long as criterion vector of  $\mathbf{x}^p \neq \mathbf{x}^{nmv}$  is some other point on the nondominated frontier (i.e.,  $z_2^p > z_2^{nmv}$ ). Then the nondominated frontier is given parametrically by

$$\{(\mathbf{x}^T \Sigma \mathbf{x}, \boldsymbol{\mu}^T \mathbf{x}) \in \mathbb{R}^2 \mid \mathbf{x} = \mathbf{x}^{nmv} + \alpha(\mathbf{x}^p - \mathbf{x}^{nmv}), \alpha \geq 0\}$$

## 5 Computing Nondominated Frontiers II

In the second category of mean-variance models portfolio selection models are those that require mathematical programming for the computation or characterization of their nondominated frontiers. A problem is typically in this category when its feasible region  $S$  is more complicated than in (9) such as when there are lower and upper bounds on the  $x_i$  as in

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{1}^T \mathbf{x} = 1, \alpha_i \leq x_i \leq \omega_i\} \quad (10)$$

or when its  $\Sigma$  is non-invertible<sup>4</sup>. Thus, this is the category of essentially all realistic, meaningful, practical, and sizable mean-variance problems.

Properties of portfolio selection models in this category include the following.

- (a) With  $S$  as given in (10), all problems have bounded feasible regions  $S$  and  $Z$ .
- (b) The minimum variance boundary of  $Z$  is piecewise parabolic in (variance, expected-return) space – and piecewise hyperbolic in (standard-deviation, expected-return) space.
- (c) Points on the nondominated portion of the minimum variance boundary separating one parabolic (or hyperbolic) segment from the next are called “turning points.”
- (d) When  $\Sigma$  is non-invertible, points on the nondominated frontier may have multiple inverse images.

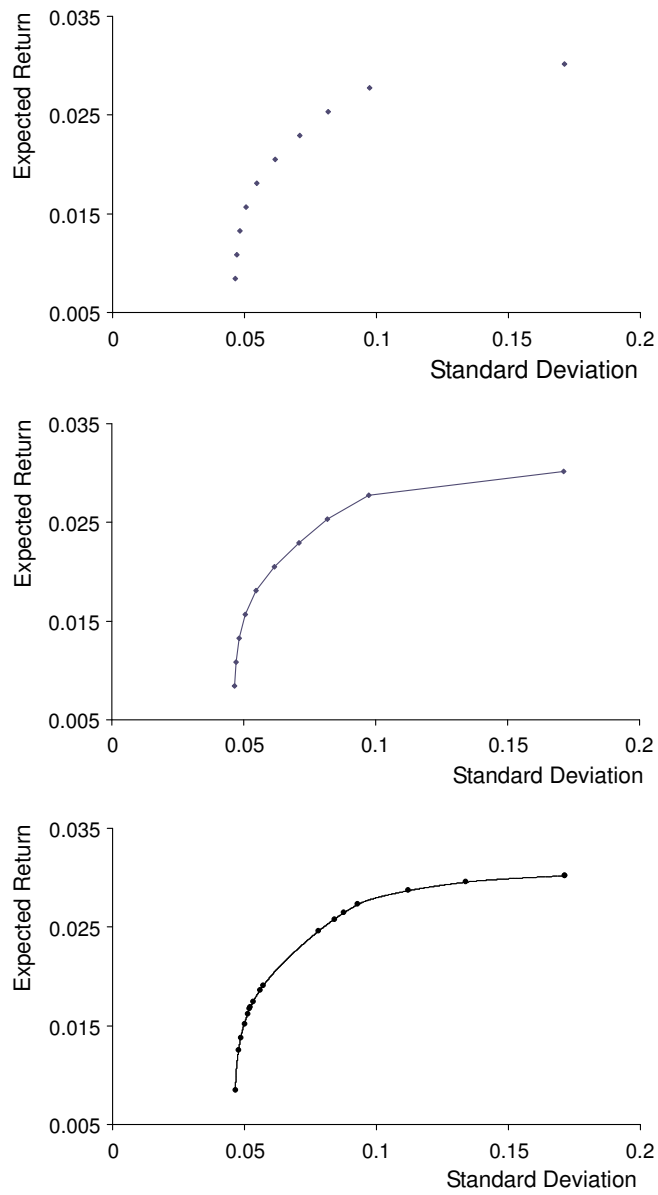


Figure 3: For an  $n = 25$  problem, above we have a 10-point dotted representation of the nondominated frontier, its piecewise linear representation of the nondominated frontier, and the problem's exact nondominated frontier (that consists of 16 hyperbolic segments, one of which is hardly discernable).

There are two approaches for computing the nondominated frontiers of problems that require mathematical programming. One is the  $\varepsilon$ -constraint approach which involves solving (3) repetitively for different values of  $\rho$  from  $[a, b]$  to produce a dotted or piecewise linear representation of the nondominated frontier as in the first two graphs of Figure 3. The other is to obtain the exact nondominated frontier as in the last graph of Figure 3 by computing all of its parabolic segments using some form of parametric quadratic programming.

Other than for the *invertible-covariance-matrix, unrestricted-variable* model, the  $\varepsilon$ -constraint approach is the *standard* approach. For problems requiring mathematical programming, it is the approach commonly described in textbooks and it is the approach commonly recommended when using any of the commercial packages such as Cplex, LINGO, Matlab, Mathematica, Solver and SAS [23]. This is because all of the mentioned packages do not possess any tools for conducting parametric quadratic programming.

The tasks to be carried out in the  $\varepsilon$ -constraint approach are as follows. First, it is necessary to obtain the upper and lower bounds of the interval  $[a, b]$ . To compute upper bound  $b$ , the expected return of the point of maximum expected return on the nondominated frontier, we solve the linear program

$$\begin{aligned} \max \{ & \boldsymbol{\mu}^T \mathbf{x} \} \\ \text{s.t. } & \mathbf{x} \in S \end{aligned} \tag{11}$$

With  $\mathbf{x}^{max} \in S$  the point returned by (11),  $b = \boldsymbol{\mu}^T \mathbf{x}^{max}$ .

Lower bound  $a$  cannot always be computed with certainty. To compute  $\hat{a}$ , the expected return of a point of minimum variance on the minimum variance boundary, we solve the quadratic programming problem

$$\begin{aligned} \min \{ & \mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x} \} \\ & \mathbf{x} \in S \end{aligned} \tag{12}$$

With  $\mathbf{x}^{nose} \in S$  the point returned by (12), we set  $\hat{a} = \boldsymbol{\mu}^T \mathbf{x}^{nose}$ . If the “nose” of  $Z$  is a single point, then  $a = \hat{a} = \boldsymbol{\mu}^T \mathbf{x}^{nose}$  and we are okay. Otherwise, if  $Z$  has a flat nose as in Figure 2 with vertical line segment  $\mathbf{z}^1$  to  $\mathbf{z}^{nmv}$ , then  $\boldsymbol{\mu}^T \mathbf{x}^{nose}$  might be less than  $a$ . This is because the image of  $\mathbf{x}^{nose}$  might be some point other than  $\mathbf{z}^{nmv}$  on the vertical segment, in which case  $\hat{a} < a$ .

---

<sup>4</sup> $\boldsymbol{\Sigma}$ , of course, must be positive semidefinite to be a valid covariance matrix.

What is frustrating with the packages is that they are not good at indicating whether or not alternative optima exist. Fortunately,  $Z$ 's with flat noses are rare in practice. It usually takes a contrived example to create a problem with a flat nose.

An  $\varepsilon$ -constraint routine that computes  $[\hat{a}, b] \supseteq [a, b]$  and then repetitively solves (3) for different values of  $\rho$  from  $[\hat{a}, b]$  is as in the following seven-step procedure.

**Step 1.** Input  $\Sigma$ ,  $\mu$ ,  $\alpha$ ,  $\omega$  and  $N$ .

Here,  $N$  is the number of points desired for the construction of a dotted or piecewise linear representation of the nondominated frontier.

**Step 2.** Solve linear program (11). Let  $\mathbf{x}^{max} \in S$  be the point returned by (11). Set  $b = \mu^T \mathbf{x}^{max}$ .

Note that if (11) has alternative optima such as illustrated by the horizontal line segment from  $\mathbf{z}^{nmer}$  to  $\mathbf{z}^2$  in Figure 2, there is no guarantee that the image of  $\mathbf{x}^{max}$  is  $\mathbf{z}^{nmer}$ . This is because its image could be some other point along the horizontal line segment. Thus, unless we have some definitive way of ruling out alternative optima, we cannot with confidence utilize the image of  $\mathbf{x}^{max}$  as the topmost point on the nondominated frontier.

**Step 3.** Solve quadratic program (12). Let  $\mathbf{x}^{nose} \in S$  be the point returned by (12). Set  $\hat{a} = \mu^T \mathbf{x}^{nose}$ ,  $\delta = (b - \hat{a})/N$  and let  $i = 0$ .

Note that if  $Z$  has a flat nose, the image of  $\mathbf{x}^{nose}$  may not be nondominated. Rather than wrestle with the possibility that  $Z$  has a flat nose, it is simplest to let the image of  $\mathbf{x}^{nose}$  be the bottommost point of the  $\varepsilon$ -constraint frontier.

**Step 4.** Let  $i = i + 1$ .

**Step 5.** Solve  $\varepsilon$ -constraint program (3) with  $\rho = b - \delta(i - 1)$ .

Note that on the first iteration, (3) solves for the topmost point on the nondominated frontier. Then, as  $\rho$  decreases with the incrementation of  $i$ , the routine produces points going down the minimum variance boundary, stopping just before the bottommost point computed in Step 3.

**Step 6.** If  $i < (N - 1)$ , go to Step 4.



Note that this routine involves  $N + 1$  optimizations because two are required to assure that a correct topmost point is obtained.

**Step 7.** With the  $N$  points generated, display the thus determined dotted or piecewise linear representation of the nondominated frontier. Stop.

The other approach for computing the nondominated frontiers of problems with feasible regions  $S$  as in (2) or (9) is to use parametric quadratic programming. There are three possibilities, and all produce exact nondominated frontiers. One is to use Markowitz's *critical line* algorithm. Introduced in [25], a recent re-description of the algorithm is available in [27]. However, it is not easy to master this algorithm, and this may explain why one sees so few implementations of it as compared to  $\varepsilon$ -constraint approaches.

A second possibility is to form the Kuhn-Tucker system of linear equations resulting from (3) and then perform parametric programming on the equations involving  $\rho$ . A third possibility is to form the Kuhn-Tucker system of linear equations resulting from (6) and then perform parametric programming on the equations involving  $\lambda$ .

After coding the seven-step  $\varepsilon$ -constraint routine in Matlab, experiments were conducted to compare CPU-times with MPQ. Results are in Table 1. The first column of the table specifies problem size in terms of the number of securities  $n$ . For each of the non-italicized entries in the table, the sample size was 10 problems. Feasible region  $S$  in all problems was as in (10) with all  $\alpha_i = 0$  and all  $\omega_i = 1$ . In all problems, the  $\Sigma$  covariance matrices were 100% dense and were randomly generated using the method described in Hirschberger, Qi and Steuer [15]. To correspond to real covariance matrices, the diagonal and off-diagonal elements had the same distributional characteristics as possessed by the real data alluded to in Table 3 of [15]. All experiments were conducted on a Dell 1.6GHz Centrino laptop with 1GB of RAM.

On the first line of the Table are the results for problems of size  $n = 200$ . Whereas it took Matlab on average 152 seconds to complete the 21 optimizations required for a 20-point  $\varepsilon$ -constraint representation of a nondominated frontier, it took MPQ on average less than 4 seconds to compute an entire exact nondominated frontier. On the second line, with problems of size  $n = 400$ , the numbers are 2,069 seconds and 50 seconds, respectively. What we are observing is that, on average, MPQ takes less time to compute an entire exact nondominated frontier than it takes Matlab to compute just a single point.

$n$	$\varepsilon$ -constraint average CPU-time	standard deviation	MPQ average CPU-time	standard deviation
200	152.0	6.5	3.7	0.1
400	2,069.3	17.6	50.2	9.6
600	9,027.2	230.4	237.5	53.1
800	<i>24,689.0</i>	-	685.5	131.8
1,000	<i>54,853.0</i>	-	1,108.2	265.1
1,200			2,585.5	886.1
1,400			3,223.5	516.3
1,600			5,478.7	1,046.9
1,800			8,351.8	3,460.8
2,000			17,706.6	5,838.6

Table 1: CPU-time results for the computation of 20-point  $\varepsilon$ -constraint representations of the nondominated frontier versus CPU-time results for the computation of the exact nondominated frontiers using MPQ. All times are in seconds.

The numbers on the fourth and fifth lines in the second column are in italics because they are estimates. They based on two optimizations and then multiplied by 10. Note that in the case of  $n = 800$ , 24,689 seconds is almost 7 hours, and in the case of  $n = 1000$ , 54,853 seconds is over 15 hours. These figures are to be compared against the roughly 11.5 and 18.5 minutes, respectively, reported by MPQ.

One might wonder whether the figures in Table 1 can be true. One might wonder whether MPQ cuts corners. One might wonder whether Matlab, despite its popularity, is a good choice for comparison. For accuracy, MPQ was tested against Optimizer on problems with up to 248 securities and results were identical up to about the seventh place to the right of the decimal point at all turning points. As for the choice of Matlab, we tested MPQ against Solver and found Solver slower. We also tested MPQ against LINGO. Although LINGO is somewhat faster than Matlab, for us LINGO encountered difficulties with 100% dense covariance matrix problems over about 600 securities. In fairness to Matlab and LINGO, it is our opinion that they were basically designed under the assumption that large portfolio problems would only be brought to computation after having had their covariance matrix

structures diagonalized. Unfortunately, diagonalizing a covariance matrix structure is typically accompanied by a loss of information. Actually, CPU-times on the order of those reported by MPQ might be possible using the “fast algorithm” outlined in Markowitz, Todd, Xu and Yamane [28] on certain problems. However, with that algorithm one must start with historical observations or scenarios, not a covariance matrix. MPQ has no such restrictions. Anyway, a coded version of that algorithm is not known to be anywhere available. Thus with MPQ, we appear to get the best of several worlds. Not only is it faster than anything available, it computes exact nondominated frontiers (as opposed to the approximations of the packages). Moreover, it can do so on dense covariance matrix problems that are larger than anything that can be handled, as demonstrated, by either Matlab or LINGO.

As seen in Table 1, for the most difficult of mean-variance problems (i.e., those with 100% dense covariance matrices), MPQ represents both a “better” and a “new” capability. For problems up to about 600-800 securities, it is a *better* capability as it can typically compute an exact nondominated frontier in much less time than it takes Matlab to produce an approximation. On problems over about 800 securities, MPQ represents a *new* capability for dense covariance matrix problems as both Matlab and LINGO each run into their own difficulties while MPQ is still able to operate well within reasonable time.

## 6 Multiple Criteria in Portfolio Selection

The mean-variance framework we have been discussing is summarized in Figure 4. With this figure as a reference point, the endeavor now is to expand upon the framework with multiple objectives to provide a theory of portfolio selection to meet the modeling needs of additional groups of investors for which the assumptions of conventional mean-variance analysis often fall short.

One such group of investors would consist of those who have the same overall focus as in the mean-variance theory we have been studying, except that they do not buy into the assumption that all elements in  $\boldsymbol{\mu}$  and  $\boldsymbol{\Sigma}$  can be known with certainty at the beginning of the holding period. Investors in this group might well wish to monitor their portfolios with other random variables such as dividends, growth in sales, and amount invested in R&D to

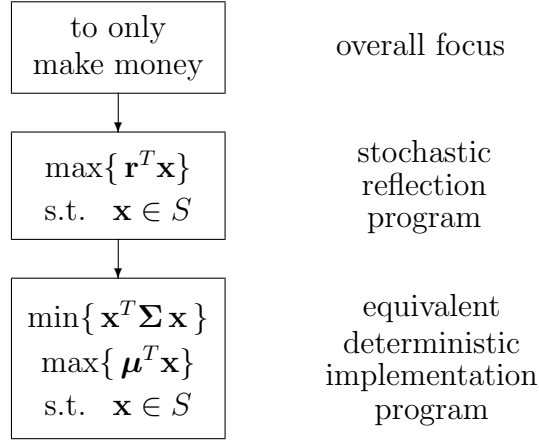


Figure 4: Hierarchical structure of the overall focus, stochastic reflection of the overall focus, and equivalent deterministic implementation of the stochastic reflection of conventional (mean-variance) portfolio selection.

hedge against errors that might be made when attempting to wrestle with information derived from inaccurate  $\boldsymbol{\mu}$ 's and  $\boldsymbol{\Sigma}$ 's alone. Another group of investors would consist of those who, in addition to end-of-holding-period portfolio return, distinctly have other stochastic criteria, perhaps ranging from liquidity to social responsibility, that they would like to have simultaneously maximized.

To articulate the concerns embedded in an investor's more complex overall focus, objectives such as from

$$\begin{aligned}
 \max\{\mathbf{d}^T \mathbf{x}\} & \quad \text{dividends} & (13) \\
 \max\{\mathbf{g}^T \mathbf{x}\} & \quad \text{growth in sales} \\
 \max\{\mathbf{a}^T \mathbf{x}\} & \quad \text{amount invested in R\&D} \\
 \max\{\mathbf{s}^T \mathbf{x}\} & \quad \text{social responsibility} \\
 \max\{\boldsymbol{\ell}^T \mathbf{x}\} & \quad \text{liquidity}
 \end{aligned}$$

where  $\mathbf{d}$ ,  $\mathbf{g}$ ,  $\mathbf{a}$ ,  $\mathbf{s}$ ,  $\boldsymbol{\ell} \in \mathbb{R}^n$  are random vectors, can be appended to portfolio return in the stochastic reflection program. Along with an equivalent deterministic implementation of the (multiobjective) stochastic reflection program, we have Figure 5.

At the overall focus level in Figure 5, the intent is to build a portfo-

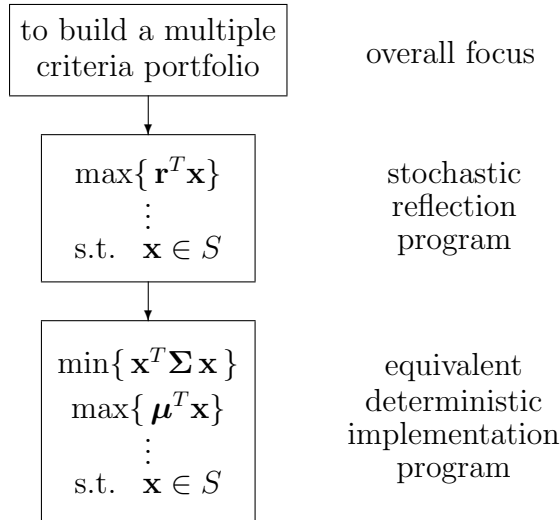


Figure 5: Hierarchical structure of the overall focus, stochastic reflection of the overall focus, and equivalent deterministic implementation of the stochastic reflection of multiple objective portfolio selection.

lio that achieves an investor’s optimal trade-off among various factors. The three vertical dots in the stochastic reflection program indicate the random variables in addition to portfolio return that are also to be optimized to the greatest extent possible. In the equivalent deterministic implementation program, the vertical dots indicate the deterministic objectives that are utilized to implement the objectives listed in the multiobjective stochastic reflection program. While one could see the creation of a pair of expected value and variance objectives for each stochastic objective, this does not always have to be the case. As discussed in Caballero et al. [4], other alternatives exist. One such alternative is to simply implement a stochastic objective in the form of just its expected value. This could be appropriate with random variables whose variance is much less or is not as important as others. One could perhaps argue that most of the stochastic objectives listed in (13) could be candidates for such treatment.

## 7 Future Methods

With the potential of having one or more quadratic and two or more linear objectives, equivalent deterministic implementation programs can become difficult. In fact, beyond the theoretical work of Guddat [10], there is little to nothing in the literature about how to compute the nondominated sets of such problems. However, in the case of one quadratic and two linear objectives, progress is being made in Hirschberger, Qi and Steuer [14] and on this we comment. With an understanding of nondominated sets in the one-quadratic-one-linear and one-quadratic-two-linear situations, one should be in a position to perceive the nature of nondominated sets in one-quadratic-three-or-more-linear situations. While the protocol of first computing the nondominated set and then selecting from it a most preferred solution remains the same, the mechanics of carrying out these tasks involve a significant step up in difficulty.

With three or more objectives, the nondominated set is no longer, in the parlance of mean-variance optimization, a *frontier*, but it is now a *surface*. Consider a one-quadratic-two-linear case. Instead in being piecewise parabolic in  $\mathbb{R}^2$ , the nondominated set is now platelet-wise *paraboloidic* in  $\mathbb{R}^3$  (like tiles on the front of a space shuttle) as in Figure 6. Fortunately, with some extra coding, the algorithm programmed into MPQ allows for a generalization to one quadratic and multiple linear objectives. We are not aware of any other research, either published or in progress, that can compute the exact nondominated sets of such problems. In computing an exact nondominated set, what MPQ is able to output includes: (1) the equation of the paraboloid that each nondominated platelet is a part of, (2) the corner points (as indicated by  $\mathbf{z}^1$  to  $\mathbf{z}^4$  in Figure 6) of each nondominated platelet in criterion space, and (3) the extreme points (not shown) of the *polyhedron subset* of efficient points in  $S$  that corresponds to each nondominated platelet. That is, the set of all images of points in an efficient polyhedron subset constitutes a nondominated platelet on the surface of  $Z$ , and the set of all inverse images of points on a nondominated platelet constitute an efficient polyhedron subset within or on the surface of  $S$ .

Preliminary computational experience with MPQ on computing the nondominated sets of portfolio problems with the same characteristics as in Table 1, but with an extra linear objective, are given in Table 2. For example, with a sample size of 10, problems of size  $n = 300$  had on average almost 2,132 platelets and took on average 23.5 seconds to compute. More extensive re-

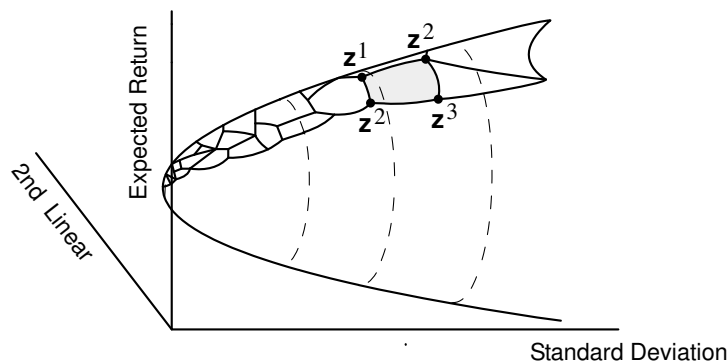


Figure 6: A portrayal of the platelet-wise nature of a nondominated surface of a one-quadratic-two-linear problem, with the corner points of the shaded platelet as indicated.

sults will be reported in the future.

$n$	number of platelets	standard deviation	MPQ average CPU-time	standard deviation
100	1,131.2	414.6	0.8	0.1
200	1,815.0	249.2	4.9	0.7
300	2,131.6	599.9	23.5	7.0
400	2,117.8	637.8	65.9	28.6

Table 2: Results for the computation of all platelets of the nondominated sets of one-quadratic-two-linear portfolio problems using MPQ. Times are in seconds.

With regard to the searching of a nondominated set of a problem such as in Table 2, one strategy is to discretize the nondominated set to some desired degree of resolution. This can be accomplished platelet by platelet as follows. For a given platelet, take convex combinations of the extreme points of its polyhedron subset in  $S$ . Because platelet size tends to increase the more distant a platelet is away from the point in  $Z$  that minimizes the quadratic objective, one would probably want to increase the number of convex combinations the further a platelet is away. Then with perhaps thousands, tens of thousands, or hundreds of thousands of points, the question is how to iden-

tify a most preferred one. Four strategies come to mind. One is to employ multiple probing as in the variant of the Tchebycheff Method described in Steuer, Silverman and Whisman [39]. Another is to pursue a projected line search strategy as advanced in Korhonen and Karaivanova [19]. A third is to utilize a criterion vector component classification scheme as, for instance, in Miettinen [31]. And a fourth might involve the utilization of some of the visualization techniques from Lotov, Bushenkov and Kamenev in [21].

## 8 Conclusions

The capabilities of MPQ provide not only a new capability for mean-variance portfolio optimization but a new world of possibilities with regard to the modeling and solution of multiple objective portfolio selection formulations. As discussed in Section 1, codes for solving for exact nondominated frontiers are generally not known to be available, either in public domain form or within the context of popular commercial packages, for problems with more than 248 securities. Furthermore, as shown in the leftmost two columns of Table 1, it is basically not possible using software such as Matlab to compute even approximations of the nondominated frontier in problems with dense covariance matrices beyond about 800 securities in reasonable time (say 4 hours). However, as shown in the rightmost two columns of Table 1, MPQ can compute exact nondominated frontiers of dense covariance matrix problems with up to near 2,000 securities in reasonable time.

In addition, as shown in Table 2, MPQ is able to compute exact nondominated frontiers of tri-criterion dense covariance matrix formulations with up to at least 400 securities in reasonable time. This opens up a whole new world of modeling possibilities, and with the importation of methods from multiple criteria optimization, new areas of portfolio optimization that weren't even worth contemplating because of the futility of the solution situation before can now be brought to the forefront.

## References

- [1] B. Aouni, F. Ben Abdelaziz, and R. El-Fayedh. *Chance constrained compromise programming for portfolio selection*. Laboratoire LARO-



- DEC, Institut Supérieur de Gestion, La Bardo 2000, Tunis, Tunisia, 2005.
- [2] M. Arenas Parra, A. Bilbao Terol, and M. V. Rodríguez Uría. A fuzzy goal programming approach to portfolio selection. *European Journal of Operational Research*, 133(2):287–297, 2001.
  - [3] C. A. Bana e Costa and J. O. Soares. A multicriteria model for portfolio management. *European Journal of Finance*, 10(3), 2004. 198-211.
  - [4] R. Caballero, E. Cerdá, M. M. Muñoz, L. Rey, and I. M. Stancu-Minasian. Efficient solution concepts and their relations in stochastic multiobjective programming. *Journal of Optimization Theory and Applications*, 110(1):53–74, 2001.
  - [5] J. Y. Campbell, A. W. Lo., and A. C. Mackinlay. *The Econometrics of Financial Markets*. Princeton University Press, Princeton, New Jersey, 1997.
  - [6] Cplex. “Cplex 9.1 User’s Manual,” ILOG, Inc., Mountain View, California, 2005.
  - [7] M. Ehrgott, K. Klamroth, and C. Schwehm. An MCDM approach to portfolio optimization. *European Journal of Operational Research*, 155(3):752–770, 2004.
  - [8] E. J. Elton, M. J. Gruber, S. J. Brown, and W. Goetzmann. *Modern Portfolio Theory and Investment Analysis*. John Wiley, New York, 6th edition, 2002.
  - [9] M. Ehrgott. *Multicriteria Optimization*. Springer, Berlin, 2nd edition, 2005.
  - [10] J. Guddat. Stability in convex quadratic programming. *Operationsforschung und Statistik*, 7:223–245, 1976.
  - [11] J. B. Guerard and A. Mark. The optimization of efficient portfolios: The case for an R&D quadratic term. *Research in Finance*, 20:217–247, 2003.

- [12] W. G. Hallerbach and J. Spronk. A multidimensional framework for financial-economic decisions. *Journal of Multi-Criteria Decision Analysis*, 11(3):111–124, 2002.
- [13] M. Hirschberger, Y. Qi, and R. E. Steuer. Quadratic parametric programming for portfolio selection with random problem generation and computational experience. In-preparation working paper, Department of Banking and Finance, University of Georgia, Athens, 2006.
- [14] M. Hirschberger, Y. Qi, and R. E. Steuer. Tri-criterion quadratic-linear programming. In-preparation working paper, Department of Banking and Finance, University of Georgia, Athens, 2006.
- [15] M. Hirschberger, Y. Qi, and R. E. Steuer. Randomly generating portfolio-selection covariance matrices with specified distributional characteristics. *European Journal of Operational Research*, 2006. forthcoming.
- [16] C. F. Huang and R. H. Litzenberger. *Foundations for Financial Economics*. Prentice-Hall, Englewood Cliffs, New Jersey, 1988.
- [17] J. E. Ingersoll. *Theory of Financial Decision Making*. Rowman & Littlefield, 1987.
- [18] C. P. Jones. *Investments: Analysis and Management*. John Wiley & Sons, New York, 7th edition, 2000.
- [19] P. Korhonen and J. Karaivanova. An algorithm for projecting a reference direction onto the nondominated set of given points. *IEEE Transactions on Systems, Man, and Cybernetics*, 29(5):429–435, 1999.
- [20] A. W. Lo, C. Petrov, and M. Wierzbicki. It’s 11pm – Do you know where your liquidity is? The mean-variance-liquidity frontier. *Journal of Investment Management*, 1(1):55–93, 2003.
- [21] A. B. Lotov, V. A. Bushenkov, and G. K. Kamenev. *Interactive Decision Maps: Approximation and Visualization of Pareto Frontier*. Kluwer Academic Publishers, Boston, 2004.
- [22] D. G. Luenberger. *Investment Science*. Oxford University Press, New York, 1997.

- [23] SAS Institute Inc. *Stock Market Analysis Using the SAS System: Portfolio Selection and Evaluation*, Cary, North Carolina, 1994.
- [24] H. M. Markowitz. Portfolio selection. *Journal of Finance*, 7(1):77–91, 1952.
- [25] H. M. Markowitz. The optimization of a quadratic function subject to linear constraints. *Naval Research Logistics Quarterly*, 3:111–133, 1956.
- [26] H. M. Markowitz and A. Perold. Portfolio analysis with factors and scenarios. *Journal of Finance*, 36(14):871–877, 1981.
- [27] H. M. Markowitz and G. P. Todd. *Mean-Variance Analysis in Portfolio Choice and Capital Markets*. Frank J. Fabozzi Associates, New Hope, Pennsylvania, 2000.
- [28] H. M. Markowitz, P. Todd, G. Xu, and Y. Yamane. Fast computation of mean-variance efficient sets using historical covariances. *Journal of Financial Engineering*, 1(2):117–132, 1992.
- [29] Matlab. *Optimization Toolbox for Use with Matlab*, Version 7.0.1 (R14), Mathworks, Inc., Natick, Massachusetts, 2004.
- [30] H. P. Mayo. *Investments: An Introduction*. Harcourt, Fort Worth, Texas, 6th edition, 2000.
- [31] K. M. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer, Boston, 1999.
- [32] W. Ogryczak. Multiple criteria linear programming model for portfolio selection. *Annals of Operations Research*, 97:143–162, 2000.
- [33] R. Roll. A critique of the asset pricing theory’s tests. *Journal of Financial Economics*, 4:129–176, 1977.
- [34] T. L. Saaty. *Decision Making for Leaders*. RWS Publications, Pittsburgh, 1999–2000 edition, 1999.
- [35] L. Schrage. *Optimization Modeling with LINGO*. Lindo Publishing, Chicago, 2003.
- [36] L. Schrage. *LINGO User’s Guide*. Lindo Publishing, Chicago, 2004.

- [37] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation and Application*. John Wiley, New York, 1986.
- [38] R. E. Steuer, Y. Qi, and M. Hirschberger. Multiple objectives in portfolio selection. *Journal of Financial Decision Making*, 1(1):11–26, 2005.
- [39] R. E. Steuer, J. Silverman, and A. W. Whisman. A combined Tchebycheff/aspiration criterion vector interactive multiobjective programming procedure. *Management Science*, 39(10):1255–1260, 1993.
- [40] Frontline Systems. “Premium Solver Platform 6.5,” Incline Village, Nevada, 2005.
- [41] S. Wolfram. *The Mathematica Book*. Wolfram Media, 5th edition, 2003.
- [42] C. Zopounidis and M. Doumpos. INVESTOR: A decision support system based on multiple criteria for portfolio selection and composition. In A. Colorni, M. Paruccini, and B. Roy, editors, *A-MCD-A Aide Multi Critere a la Decision - Multiple Criteria Decision Aiding*, pages 371–381. European Commission Joint Research Centre, Brussels, 2000.

## Summary

This paper addresses the problem of portfolio selection in finance. In many cases, currently available software to compute the efficient frontier runs into difficulty in problems with more than about 600 securities. To proceed beyond this size, it is often necessary to modify the problem in which case there is typically a loss of information. In this paper, we discuss a computer capability that can exactly compute mean-variance efficient frontiers of problems with up to 2,000 securities in very reasonable time (even if a problem's covariance matrix is 100% dense).

The paper also discusses an augmentation to the theory of portfolio selection that allows multiple objectives (such as dividends, liquidity, social responsibility, amount invested in R&D, and so forth) to be incorporated into the portfolio selection process. In such problems, the efficient set is no longer a "frontier," but is now best described as a "surface" with the interesting property that it is composed of platelets (like on the back of a turtle). Moreover, the computer capability that can extend exact computations of mean-variance efficient frontiers out to 2,000 securities also has, after additional coding, the ability to exactly compute all of the platelets of multiple objective efficient surfaces of problems with up to about 400 securities.