$\mathcal{I}^2\mathcal{SDS}$
The Institute for Integrating Statistics in Decision Sciences

Technical Report TR-2008-16
September 2, 2008

An Exact Solution Approach for Portfolio Optimization Problems Under Stochastic and Integer Constraints

Pierre Bonami
Universite de la Mediterranee, France

Miguel A. Lejeune
Department of Decision Sciences
The George Washington University, USA

An Exact Solution Approach for Portfolio Optimization Problems Under Stochastic and Integer Constraints

Pierre Bonami[1], Miguel A. Lejeune[2]

[1] Universite de la Mediterranee, Marseille, France. pierre.bonami@lif.univ-mrs.fr
[2] George Washington University, Washington, DC, mlejeune@gwu.edu

In this paper, we study extensions of the classical Markowitz mean-variance portfolio optimization model. First, we consider that the expected asset returns are stochastic by introducing a probabilistic constraint which imposes that the expected return of the constructed portfolio must exceed a prescribed return threshold with a high confidence level. We study the deterministic equivalents of these models. In particular, we define under which types of probability distributions the deterministic equivalents are second-order cone programs, and give closed-form formulations. Second, we account for real-world trading constraints (such as the need to diversify the investments in a number of industrial sectors, the non-profitability of holding small positions and the constraint of buying stocks by lots) modeled with integer variables. To solve the resulting problems, we propose an *exact* solution approach in which the uncertainty in the estimate of the expected returns and the integer trading restrictions are *simultaneously* considered. The proposed algorithmic approach rests on a non-linear branch-and-bound algorithm which features two new branching rules. The first one is a static rule, called *idiosyncratic risk* branching, while the second one is dynamic and is called *portfolio risk* branching. The two branching rules are implemented and tested using the open-source Bonmin framework. The comparison of the computational results obtained with state-of-the-art MINLP solvers (MINLP_BB and CPLEX) and with our approach shows the effectiveness of this latter which permits to solve to optimality problems with up to 200 assets in a reasonable amount of time. The practicality of the approach is illustrated through its use for the construction of four fund-of-funds now available on the major trading markets.

Keywords: Programming: stochastic, integer: nonlinear, branch-and-bound,
Finance: portfolio, trading constraints, market risk, downside risk measure

1 Introduction

Since Markowitz groundbreaking work in portfolio selection [40], portfolio optimization has been receiving sustained attention from both asset liability professionals and academics. The mean-variance approach studies how investors can construct optimal portfolios taking into consideration the trade-off between market volatility and expected returns. Out of a universe of r risky assets and one non-risky asset characterized by a known return μ_0 that usually reflects the interest rate on the money market, an efficient frontier of optimal portfolios can be constructed. Portfolios on the efficient frontier offer the maximum possible expected return for a given level of risk. The original Markowitz model assumes that the expected returns $\bar{\mu} \in \mathcal{R}^r$ of the risky assets and the variance-covariance matrix $\Sigma \in \mathcal{R}^{r \times r}$ of the returns are known. One of the several formulations of the mean-variance portfolio selection problems involves the construction of a portfolio with minimal risk provided that a prescribed return level R is attained. This model is formulated by the following mathematical program:

$$\begin{aligned} & \min w^T \Sigma w \\ \text{subject to } & \mu_0 w_0 + \bar{\mu}^T w \geq R \\ & \sum_{j=0}^{r} w_j = 1 \\ & w \in \mathcal{R}^{r+1} \end{aligned} \quad . \tag{1}$$

In the problem above, the decision variables w_j, $j = 1, \ldots, r$ represent the proportion of capital invested in the risky asset j, while w_0 is the fraction of capital invested in the money market. The objective function aims at minimizing the variance of the portfolio $w^T \Sigma w$, and the constraint

$$w_0 + \sum_{j=1}^{r} w_j = 1 \tag{2}$$

enforces that the sum of the investments is equal to 1. Clearly, the investor can allocate part of the available capital K to the money market w_0.

In the last decade, much effort has been devoted to extending Markowitz work and making the modern portfolio theory more practical. In this study, we propose models that account for two limitations associated with the mean-variance approach, namely *(i)* the randomness in the parameters describing the model and *(ii)* some of the trading restrictions of stock markets.

The classical mean-variance framework relies on the perfect knowledge of the expected returns of the assets and the variance-covariance matrix. However, these returns are unobservable and unknown. Even obtaining accurate estimates of them is very complicated. Indeed, many possible sources of errors (e.g., impossibility to obtain a sufficient number of data samples, instability of data, differing personal views of decision makers on the future returns [45], etc.) affect their estimation leading to the so-called *estimation risk* [4] in portfolio selection. The estimation risk has been shown to be the source of very erroneous decisions, for, as pointed in [10, 16], the composition of the optimal portfolio is very sensitive to the mean and the covariance matrix of the asset returns, and minor perturbations in the moments of the random returns can result in the construction of very different portfolios. Decision-makers would often rather trade-off some return for a more secure portfolio that performs well under a wide set of realizations of the random variables. The need for constructing portfolios that are much less impacted by inaccuracies in the estimation of the mean and the variance of the return is therefore clear.

The focus here is on the uncertainty associated with the estimation of the expected returns. It is indeed a widespread belief among portfolio managers, and it was shown in [8, 14], that the portfolio estimation risk is mainly due to errors in the estimation of the expected return and not so much to errors in the estimation of the variance-covariance matrix [10]. In this paper, we assume that the expected return is stochastic and characterized by a probability distribution, and we require that the expected return of the portfolio is larger than a given target with a high confidence level. We show that the associated problem takes the form of a *probabilistically constrained problem with random technology matrix* [30, 47] that can be reformulated as a non-linear optimization problem (not necessarily convex). We define under which conditions and for which classes of probability distributions the deterministic equivalent problem is convex and takes the form of a second-order cone problem. If a closed-form formulation of the deterministic equivalent cannot be obtained, we provide convex approximations that are obtained by using variants of the Chebychev's inequality [41] and whose tightness depends on the properties of the probability distribution. This convexity analysis of the model gives insights about its applicability and its computational tractability. In related studies, Costa and Paiva [17], Tütüncü and Koenig [58] and Goldfarb and Iyengar [25] have also studied the mean-variance framework in a *robust* context, assuming that the expected return is stochastic. They characterize the parameters involved in the mean and the variance-covariance matrix with specific types (polytopic, box, ellipsoidal) of uncertainty, and build semi-definite or second-order cone programs. In [20], a risk-averse approach is used for the value-at-risk formulation of the optimization problem, in which only partial information about the probability distribution is known.

The performance measure used in this paper falls within the Markowitz framework where the trade-off between expected return and variance is analyzed. More precisely, this measure belongs to the family of downside risk measures which focus on avoiding the return to fall below a specified target. Our risk measure is related to Roy's safety-first risk criterion [51] that identifies as optimal the portfolio for which the probability of its return falling below a prescribed threshold is minimized. Roy's risk measure, which was later extended in [35] for multi-period portfolio selection problems, is close to the Sharpe Ratio [53] which maximizes the ratio of excess return to risk. Many other risk measures exist, such as value-at-risk [43], conditional value-at-risk [50], stochastic dominance [18, 19, 26], semi-deviation [46], excess probabilities [56], mean-absolute deviation [32], semi-absolute deviation [21], measures of outperformance with respect to a benchmark, etc. Criteria for the selection of risk measures is a topic widely discussed in the literature and it is analyzed under different angles, including the coherence of risk measures [1], portfolio holdings [9], convexity properties [23], and computational tractability [38]. Needless to say, provided the variety of criteria and objectives, that there is no universally recommendable risk measure.

The need to account for stock market specifics exacerbates the complexity of the portfolio selection problem. Real-life trading restrictions, such as minimum amount to invest in an asset, requirements to buy assets in large lots, or purchase of assets in a minimal number of industrial sectors, are not considered in the classical mean-variance models. In the present study, we consider these requirements that are respectively called *buy-in threshold*, *round lot*, and *diversification* trading constraints. The modeling of such constraints involves the introduction of integer variables and further challenges the computational tractability of the associated problems [16, 54]. In the next paragraph, we proceed to a review of the literature in which the construction of optimal portfolios satisfying such integer constraints is addressed.

Bienstock [6] looks at variants of the Markowitz model which feature a cardinality constraint and buy-in threshold constraints. He shows that the problem is NP-complete when a cardinality constraint on the number of assets in the portfolio is present. A branch-and-cut solution framework is developed and computational results are presented. In [29], an exact branch-and-bound solution approach is proposed for problems subject to buy-in threshold, cardinality and round lot constraints. Heuristic approaches have been proposed for models enforcing the mean-variance [15], mean-absolute deviation [39], and the mean-semi-absolute deviation [31] risk criteria, and including round lot constraints. Frangioni and Gentile [24] also consider buy-in threshold constraints, and develop a new family of cutting planes to handle them. Computational results for problems with up to 300 assets are reported. Using mean absolute deviation as optimization criterion, Konno and Yamamoto [33] take into account cardinality and fixed transaction cost constraints and solve problems in which up to 54 assets can be included in the portfolio. An exact solution approach is proposed in [36] for the mean-variance model containing a cardinality and a concave transaction cost constraints. Computational results were reported for portfolios containing up to 30 securities. In [42], a branch-fix-and-relax algorithm is proposed to solve a multi-factor model in which the expected return of the portfolio is maximized subject to the satisfaction of cardinality or buy-in threshold constraints. It is important to remark that all the studies above do not account for uncertainty in the problem parameters.

To the best of our knowledge, this study is the first one to propose an *exact* solution approach for portfolio optimization problems in which uncertainty in the estimate of the expected return and real-life market restrictions modeled with integer constraints are *simultaneously* considered. The combination of integer and probabilistic constraints makes such problems very difficult to solve. These problems belong to the family of Mixed Integer Non-Linear Programs (MINLP) for which only very few solvers are available. In this paper, we use the computational framework offered by the open-source mixed-integer non-linear solver `Bonmin` [7]. We propose a non-linear branch-and-bound algorithmic approach, and we develop two new branching rules, called *idiosyncratic risk* and *portfolio risk* branching rules. Extended computational experiments on problems containing up to 200 assets clearly show the effectiveness and utility of the two new branching rules. The reader will note that, although the results reported in the paper are obtained for one of the variants of the probabilistic Markowitz model (i.e., risk minimization subject to the attainment of a predefined return level), the proposed solution approach can be easily extended to the other variants. The relevance of the performance measure and the optimization model, as well as the applicability of the solution method are confirmed, by their use by the Private Banking Group of a major financial institution to construct four long-only absolute return Fund-of-Funds (FoF).

The paper is organized as follows. In the first part of Section 2, we describe the characteristics of the constraint enforcing that the portfolio return exceeds with a probability p a given prescribed return level. We present the problem formulation and its deterministic equivalent, we study under which condition it is convex, and we propose exact or approximate closed-form formulations of the deterministic equivalent problem. The second part of Section 2 is devoted to the formulation of the integer constraints and models associated with three types of trading restrictions. Section 3 describes the proposed solution approach. Section 4 reports and comments on the computational results. Section 5 illustrates the applicability and relevance of the optimization model and solution method. Section 6 provides concluding remarks and suggests extensions to the proposed study.

2 Problem formulation and properties

2.1 Probabilistic portfolio optimization model

The proposed portfolio optimization model takes the form of a probabilistically constrained optimization [12] model with random technology matrix. We refer the reader to [30, 47] for a first study of probabilistic constraints with random technology matrix in applications pertaining to the transportation and diet problems, and to [13, 27, 49] for more recent studies.

We denote by ξ the random vector of expected returns of the r risky assets; ξ has an r-variate distribution with the following mean vector

$$\mu = (\mu_1, \mu_2, \ldots, \mu_r)^T\,, \quad \mu_j = E(\xi_j), j = 1, \ldots, r,$$

and variance-covariance matrix

$$\Sigma = E[(\xi - \mu)(\xi - \mu)^T]\,.$$

The probabilistic constraint

$$\mathcal{P}\Big(\mu_0 w_0 + \sum_{j=1}^{r} \xi_j w_j \geq R\Big) \geq p\,, \tag{3}$$

in which the coefficients ξ multiplying the decision variables w are stochastic and not (necessarily) independent, guarantees that the expected return of the portfolio $\mu_0 w_0 + \sum_{j=1}^{r} \xi_j w_j$ is above the prescribed minimal level of return R with a high probability p, typically defined on $[0.7, 1)$.

The stochastic version of Markowitz mean-variance portfolio optimization problem [40] reads:

$$\begin{aligned} & \min\ w^T \Sigma w \\ \text{subject to }\ & \mathcal{P}\Big(\mu_0 w_0 + \sum_{j=1}^{r} \xi_j w_j \geq R\Big) \geq p \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & w \in \mathcal{R}_+^{r+1} \end{aligned} \quad . \tag{4}$$

The decision variables are given by the $[r+1]$-dimensional vector w of portfolio positions. We recall that w_0 is the proportion of the available capital K invested in the money market with fixed return μ_0, w_j, $j = 1, \ldots, r$ is the proportion of the capital K invested in the risky asset j, and the objective function $w^T \Sigma w$ represents the variance of the portfolio. In our model, we assume that the variables are positive, not allowing short-selling positions. This constraint can be removed without affecting the nature of the problem.

2.2 Deterministic equivalent

We shall first show that the deterministic equivalent of the probabilistic portfolio optimization model is a non-linear programming optimization problem. Defining by $\psi = \frac{\xi^T w - \mu^T w}{\sqrt{w^T \Sigma w}}$ the random variable with mean 0 and variance 1, thereafter referred to as the normalized portfolio return, it follows that

$$\mathcal{P}(\xi^T w \geq R) = \mathcal{P}\left(\psi \geq \frac{R - \mu^T w}{\sqrt{w^T \Sigma w}}\right) = 1 - F_{(w)}\left(\frac{R - \mu^T w}{\sqrt{w^T \Sigma w}}\right) \quad , \tag{5}$$

where $F_{(w)}$ is the cumulative probability distribution of the (normalized) portfolio return and $F_{(w)}^{-1}$ is its inverse. We note that the exact form of the probability distribution F depends on the holdings w of the portfolio and has always mean 0 and standard deviation 1. Therefore, the probabilistic constraint (3) becomes

$$\begin{aligned} & 1 - F_{(w)}\left(\frac{R - \mu^T w}{\sqrt{w^T \Sigma w}}\right) \geq p \\ \Leftrightarrow\ & F_{(w)}\left(\frac{R - \mu^T w}{\sqrt{w^T \Sigma w}}\right) \leq 1 - p \\ \Leftrightarrow\ & \mu^T w + F_{(w)}^{-1}(1-p)\sqrt{w^T \Sigma w} \geq R \end{aligned} \quad , \tag{6}$$

where $F_{(w)}^{-1}(1-p)$ is the $(1-p)$-quantile of $F_{(w)}$.
The deterministic equivalent of (4) is the following non-linear optimization problem:

$$\begin{aligned} & \min\ w^T \Sigma w \\ \text{subject to}\ & \mu^T w + F_{(w)}^{-1}(1-p)\sqrt{w^T \Sigma w} \geq R \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & w \in \mathcal{R}_+^{r+1} \end{aligned} \quad . \tag{7}$$

In the next-subsections, we shall study under which conditions, i.e. for which classes of probability distributions the above problem is a second-order cone optimization problem (i.e., thus convex, and solvable in polynomial time). We shall see that it is not always possible to derive an exact closed-form formulation of the second-order cone problem for each probability distribution. We shall, therefore, using variants of Chebychev's inequality, derive closed-form approximations of the second-order cone problem that are valid for some families of probability distributions.

2.2.1 Convexity results

a) Symmetric probability distributions

The probability distribution F of a random variable ξ is symmetric around its mean μ if $P(\xi \geq \mu + b) = P(\xi \leq \mu - b)$, $b \in \mathcal{R}$, and is centrally symmetric if $P(\xi \geq b) = P(\xi \leq -b)$. We provide a more formal definition below.

Definition 2.1 *A probability distribution of an r-variate random vector is centrally symmetric if its density function f is such that $f(A) = f(-A)$ for all Borel sets $A \subseteq \mathcal{R}^r$.*

Theorem 2.2 *If $p \in [0.5, 1)$ and if the probability distribution of $\xi^T w$ is symmetric, the deterministic equivalent $\mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R$ of the probabilistic constraint $\mathcal{P}(\xi^T w \geq R) \geq p$ is a second-order cone constraint.*

Proof: The matrix of variance-covariance Σ is positive semidefinite, and thus the function $\sqrt{w^T \Sigma w}$ is convex. To show that $\mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R$ is a second-order cone constraint whose feasible set is convex, it is enough to prove that the function $\mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w}$ is concave, which is the case if $F^{-1}_{(w)}(1-p)$ is smaller than or equal to 0.

Since the probability distribution of ξ is symmetric, the probability distribution F of the normalized random variable ψ is centrally symmetric. It follows that $F_{(w)}(0) = 0.5$ (or, equivalently, that $F^{-1}_{(w)}(0.5) = 0$). This, combined with the fact that any cumulative distribution function is increasing, implies that $F_0^{-1}(1-p), p \in [0.5, 1)$ is at most equal to 0, which was set out to prove. ■

Clearly, problem (7) minimizes a convex quadratic function over a second-order cone and some linear constraints, and is therefore a convex, second-order cone problem.

b) Positively skewed probability distributions

The skewness is a measure of the asymmetry of a probability distribution of a real-valued random variable [2], and is computed as

$$skew(\xi) = \frac{E[\xi - \mu]^3}{\sigma^3} \ ,$$

where μ and σ are respectively the mean and standard deviation of ξ.

The probability distribution F of a random variable is said to be right-skewed or to have positive skewness (left-skewed or negative skewness, respectively) if the right, upper value (left, lower value, resp.) tail is longer or fatter than the left, lower value (right, upper value, resp.), or, stated differently, if its median m is strictly smaller (larger, resp.) than its mean μ.

Definition 2.3 *The probability distribution of an r-variate random vector ξ has positive skewness if*

$$P(0 \geq \psi) \geq P(m \geq \psi) \Leftrightarrow F^{-1}(\alpha) \leq 0, \ \alpha \leq 0.5$$

where $E[\psi] = E[\xi - \mu] = 0$ and $F(m) = P(m \geq \psi) = 0.5$.

Theorem 2.4 *If $p \in [0.5, 1)$ and if the probability distribution of $\xi^T w$ has positive skewness, the deterministic equivalent $\mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R$ of the probabilistic constraint $\mathcal{P}(\xi^T w \geq R) \geq p$ is a second-order cone constraint.*

Proof: As mentioned above, $\mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R$ is a second-order cone constraint if $F^{-1}_{(w)}(1-p)(p \geq 0.5)$ is smaller than or equal to 0.

This follows immediately from Definition 2.3:

$$0 > F^{-1}_{(w)}(1-p), 1-p \leq 0.5$$

for the probability distribution $F_{(w)}$ of the normalized random variable ψ has positive skewness. ■

The exact value of the quantile $F^{-1}_{(w)}(1-p)$ can be derived for some probability distributions. If we assume the returns of the risky assets to be normally distributed, then the normalized portfolio return ψ has a standard normal cumulative distribution function

$$\phi(\psi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\psi} e^{-t^2/2} dt \ ,$$

and the numerical value of its quantile $\phi^{-1}(p)$ is known. The same applies if the normalized portfolio return is uniformly distributed in an ellipsoid $\Omega = \{\omega = \mathcal{Q}z : \|z\| \leq 1\}$ with $\|z\|$ being the Euclidian norm of z.

2.2.2 Quantile approximation

The exact value of the $(1-p)$-quantile $F^{-1}_{(w)}(1-p)$ cannot be derived for each probability distribution $F_{(w)}$ which therefore impedes the derivation of the exact deterministic equivalent of the probabilistic constraint (3) in (4). In this section, using variants of Chebychev's inequality, we derive convex approximations of (3) for different classes of probability distributions. Such approximations are popular in the robust optimization literature [5], and differ in terms of their conservativeness.

Theorem 2.5 *The second-order cone constraint*

$$\mu^T w - \sqrt{\frac{p}{1-p}} \sqrt{w^T \Sigma w} \geq R$$

is a valid approximation of the probabilistic constraint

$$\mathcal{P}\left(\xi^T w \geq R\right) \geq p \tag{8}$$

when the portfolio return follows any probability distribution defined by its first two moments μ and σ^2.

Proof: Let us consider the random variable Y such that $Y^T w = (2\,\mu^T - \xi^T)w$: $Y^T w$ has the same mean and variance as $\xi^T w$.
Applying Chebychev's inequality, we obtain

$$\mathcal{P}\left(Y^T w - \mu^T w > \mu^T w - R\right) \leq \begin{cases} \frac{1}{1+\frac{(\mu^T w - R)^2}{w^T \Sigma w}} = \frac{w^T \Sigma w}{w^T \Sigma w + (\mu^T w - R)^2} & \text{if } \mu^T w \geq R \\ 1 & \text{otherwise} \end{cases} . \tag{9}$$

Clearly,

$$\mathcal{P}\left(Y^T w - \mu^T w > \mu^T w - R\right) = \mathcal{P}\left(\mu^T w - Y^T w < R - \mu^T w\right) = \mathcal{P}\left(\xi^T w - \mu^T w < R - \mu^T w\right) \ .$$

This, combined with (9), successively implies that

$$\mathcal{P}\left(\xi^T w - \mu^T w < R - \mu^T w\right) \leq \frac{w^T \Sigma w}{w^T \Sigma w + (\mu^T w - R)^2}$$

$$1 - \mathcal{P}\left(\xi^T w - \mu^T w \geq R - \mu^T w\right) \leq \frac{w^T \Sigma w}{w^T \Sigma w + (\mu^T w - R)^2}$$

$$\mathcal{P}\left(\xi^T w - \mu^T w \geq R - \mu^T w\right) \geq 1 - \frac{w^T \Sigma w}{w^T \Sigma w + (\mu^T w - R)^2} \tag{10}$$

Therefore,

$$1 - \frac{w^t \Sigma w}{w^t \Sigma w + (\mu^T w - R)^2} \geq p$$

is sufficient for constraint (8) to hold. The expression above can be successively rewritten as:

$$\begin{aligned}
&(1-p)\ \left(w^T \Sigma w + (\mu^T w - R)^2\right) \geq w^T \Sigma w \\
&(1-p)\ (\mu^T w - R)^2 \geq p\, w^T \Sigma w \\
&\mu^T w - \sqrt{\frac{p}{1-p}}\ \sqrt{w^T \Sigma w} \geq R
\end{aligned}$$

which was set out to prove. ■

A tighter approximation can be obtained if the probability distribution is symmetric.

Theorem 2.6 *The second-order cone constraint*

$$\mu^T w - \sqrt{\frac{1}{2(1-p)}} \sqrt{w^T \Sigma w} \geq R$$

is a valid approximation of the probabilistic constraint

$$\mathcal{P}\left(\xi^T w \geq R\right) \geq p$$

when the portfolio return has a symmetric probability distribution.

Proof: Chebychev's inequality for symmetric probability distributions is formulated as follows:

$$\mathcal{P}\left(\xi^T w - \mu^T w > \mu^T w - R\right) \leq \begin{cases} 0.5 \cdot \min\left[1, \frac{w^T \Sigma w}{(\mu^T w - R)^2}\right] & \text{if } \mu^T w \geq R \\ 1 & \text{otherwise} \end{cases}, \tag{11}$$

where the expression $min[a, b]$ returns the minimum value of a and b

Consequently, we have that

$$1 - \frac{1}{2} \frac{w^T \Sigma w}{(\mu^T w - R)^2} \leq P\left(\xi^T w - \mu^T w \leq \mu^T w - R\right)\ ,$$

and, using the same variable substitution approach as above, we obtain

$$P\left(\xi^T w - \mu^T w \geq R - \mu^T w\right) \geq 1 - \frac{1}{2} \frac{w^T \Sigma w}{(\mu^T w - R)^2}\ . \tag{12}$$

Therefore,

$$1 - \frac{1}{2} \frac{w^T \Sigma w}{(\mu^T w - R)^2} \geq p$$

is a sufficient condition for $\mathcal{P}\left(\xi^T w \geq R\right) \geq p$ to hold true. Consequently,

$$\begin{aligned}
&2\ (\mu^T w - R)^2 \geq \frac{w^T \Sigma w}{1-p} \\
&\mu^T w - \sqrt{\frac{1}{2(1-p)}}\ \sqrt{w^T \Sigma w} \geq R
\end{aligned}$$

which was set out to prove. ■

2.3 Integrality constraints for stock market restrictions

We now propose extensions of problem (7) in order to take into account real-life stock market restrictions. These are modeled through the introduction of integer decision variables in (7), and pertain to the prevention from holding small positions (Section 2.3.1), to the requirement of purchasing shares by batch of a certain size (Section 2.3.2), and to the investment in a predefined minimal number of industrial sectors (Section 2.3.3).

This leads to the formulation of a discrete optimization problem including a probabilistic constraint. The relaxation of the integrality conditions results in a convex optimization problem whose deterministic equivalent is a second-order cone problem. The deterministic equivalent of the discrete optimization problem reads:

$$\begin{aligned} & \min w^T \Sigma w \\ \text{subject to } & \mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R \\ & g_j(w,y) \leq 0,\ j = 1,\ldots,m \\ & w \in \mathcal{R}^{r+1}_{+} \\ & y \in \mathcal{Z}_{+} \end{aligned} \quad . \tag{13}$$

Problem (13) minimizes the volatility of the portfolio over a convex feasible set determined by the second-order cone constraint on the expected return and m deterministic constraints $g_j(w,y) \leq 0$. The decision variables y are integer-valued.

2.3.1 Buy-in threshold constraints

In this section, we introduce constraints that prevent investors from holding very small active positions. The rationale for this hinges on the fact that such small positions have very limited impact on the total performance of the portfolio [54], but trigger some tracking and monitoring costs. Certain portfolio selection models, such as the Markowitz model, are known for occasionally returning an optimal portfolio containing very small investments in a (large) number of securities. Such a portfolio is in practice very difficult to justify due to the costs of establishing and maintaining it (brokerage fees, bid-ask spreads, etc.), and the usually poor liquidity of small positions. In order to avoid this, constraints preventing from holding an active position representing strictly less than a prescribed proportion w_{min} of the available capital are useful. To model such constraints, we introduce r extra binary variables $\delta_j \in \{0,1\}, j = 1,\ldots,r$ taking value 1 if the investor detains shares of asset j (i.e., $w_j > 0$):

$$w_j \leq \delta_j \ , \ j = 1,\ldots,r \ . \tag{14}$$

Small investments are avoided by introducing the following constraints:

$$w_{min}\, \delta_j \leq w_j \ , \ j = 1,\ldots r \ . \tag{15}$$

With these additional variables and constraints, problem (13) becomes

$$\begin{aligned}
&\min w^T \Sigma w \\
&\text{subject to } \mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R \\
&w_0 + \sum_{j=1}^{r} w_j = 1 \\
&w_j \leq \delta_j \ , \ j = 1, \ldots, r \\
&w_{min}\, \delta_j \leq w_j \ , \ j = 1, \ldots r \\
&\delta \in \{0, 1\}^r \\
&w \in \mathcal{R}_+^{r+1}
\end{aligned} \quad . \tag{16}$$

2.3.2 Round lot purchasing constraints

Large institutional investors usually purchase large (i.e., even lot) blocks of individual financial assets. This is primarily because such blocks are more easily traded than smaller (i.e., odd lot) holdings, but also for liquidity reasons, i.e., to avoid the risk of getting stuck with a small, poorly liquid holding of a financial asset. Another reason to buy stocks by lots of large quantity is that, often, brokers require a premium for odd lot trades because they may have to split an even lot which would leave them with the remaining odd lot part. The effect of the round lot constraints on the structure of the portfolio is very important when assets whose prices are large relative to the size of the trade are involved. In that case it is very important to have a portfolio construction approach that effectively handles the round lot constraints within the optimization procedure. This is what motivates the construction of portfolio models including *round lot* constraints that require the purchase of shares by *batches* or *lots* of M stocks.

To each risky asset j, we associate a general integer variable γ_j, and a round lot constraint

$$x_j = \gamma_j M \ , \ j = 1, \ldots, r \tag{17}$$

imposing that the number x_j of shares of asset j in the portfolio is a multiple of M. Denoting by p_j the face value of stock j and by K the available capital, we have $x_j = \frac{w_j K}{p_j}$, and we reformulate (17) as

$$w_j = \frac{p_j \gamma_j M}{K}, \ j = 1, \ldots, r \ .$$

Problem (13) becomes a second-order cone problem with general integer decision variables

$$\begin{aligned}
&\min w^T \Sigma w \\
&\text{subject to } \mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R \\
&w_0 + \sum_{j=1}^{r} w_j = 1 \\
&w_j = \frac{p_j \gamma_j M}{K}, \ j = 1, \ldots, r \\
&\gamma \in \mathcal{Z}_+^r \\
&w \in \mathcal{R}_+^{r+1}
\end{aligned} \quad . \tag{18}$$

2.3.3 Diversification constraints

Many institutional investors have limitations on the allowable exposure to risky investments. Very often, such limits are defined by an upper bound on the maximum percentage of the portfolio value that may be invested in certain categories of financial assets, and/or by the requirement to invest in a predefined minimum number of asset categories or industrial sectors. In this section, we consider constraints that force the investor to diversify its portfolio by purchasing assets in at least L_{min} different economic sectors. Every asset j is linked with an economy sector k, so that the sets $S_k, k = 1, \ldots, L$ of assets affiliated with a sector k form an exact partition of $\{1, \ldots, r\}$. We associate a binary variable $\zeta_k \in \{0, 1\}$ with each economic sector k: ζ_k is equal to 1 if and only if the investment in sector k ($\sum_{j \in S_k} w_j$) is above a minimum pre-defined level $s_{\min}$:

$$s_{min}\,\zeta_k \leq \sum_{j \in S_k} w_j \leq s_{min} + (1 - s_{min})\ \zeta_k\ .$$

In addition to the constraint above we must add a *cardinality* constraint to satisfy the diversification requirement.

The diversification condition requires to detain "representative" positions in at least L_{min} sectors. Note that the constraints above do not consider a very small position in a sector k (i.e., $\leq s_{min}$) as contributing to the diversification of the portfolio. The probabilistic Markowitz model with diversification constraint reads:

$$\begin{aligned} & \min\, w^T \Sigma w \\ \text{subject to } & \mu^T w + F^{-1}_{(w)}(1-p)\sqrt{w^T \Sigma w} \geq R \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & s_{\min}\,\zeta_k \leq \sum_{i \in S_k} w_k \leq s_{min} + (1 - s_{min})\,\zeta_k, k = 1, \ldots, L \\ & \sum_{k=1}^{L} \zeta_k \geq L_{min} \\ & \zeta \in \{0,1\}^L \\ & w \in \mathcal{R}_+^{r+1} \end{aligned} \quad . \tag{19}$$

3 Solution Method

In this paper, we develop an *exact* mixed-integer non-linear programming solution method for portfolio optimization problems subject to the joint enforcement of probabilistic constraint on the expected portfolio return and integer constraints representative of trading mechanisms. More precisely, we rely on a non-linear branch-and-bound algorithm that we complement with new branching rules, namely the *idiosyncratic risk* and *portfolio risk* branching rules.

The non-linear branch-and-bound method is a method aimed at solving problem of the general form

$$\begin{aligned} \min\ & f(x) \\ \text{subject to}\ & g_i(x) \leq 0,\ \forall i = 1, \ldots, m \\ & x_i \in \mathcal{Z},\ \forall i \in I \\ & x \in \mathbb{R}^n \end{aligned},$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}^m$ are at least once continuously differentiable convex functions.

In the following, we give a brief reminder of the classical branch-and-bound algorithm and then describe two new branching rules which are suitable for the considered portfolio optimization problems.

3.1 Non-linear branch-and-bound algorithm

The branch-and-bound algorithm solves problems of the form (13) by performing an implicit enumeration through a tree search. The algorithm starts by solving the continuous relaxation where all integrality requirements have been removed. We denote by (w^*, y^*) the optimal solution of this continuous relaxation. If y^* is integer valued, then (w^*, y^*) is the optimal solution and the problem is solved. Otherwise, at least one of the integer variables (y_i) has a non integer value. Such a variable is chosen for *branching*: two sub-problems (or nodes) are created where the upper and lower bounds on $y_{\hat{i}}$ are set to $\lfloor y^*_{\hat{i}} \rfloor$ and $\lceil y^*_{\hat{i}} \rceil$, respectively, and the two sub-problems are put in a list of open nodes.

Then, at each subsequent iteration of the algorithm, a sub-problem is chosen from the list of open nodes, and the continuous relaxation of the current node is solved providing a lower bound. The enumeration at the current node can be stopped, or stated differently, the node is said to be fathomed or pruned, if any of the three following conditions happen:

- the continuous relaxation is infeasible (pruning by infeasibility);
- the optimal solution of the continuous relaxation is not better than the value of the best integer feasible solution found so far (pruning by bounds);
- the optimal solution of the continuous relaxation is integer feasible (pruning by optimality).

If the optimal solution of the continuous relaxation solution (w^*, y^*) cannot be pruned, then at least one of the integer variables (y_i) has a non integer value ($y^*_i \notin \mathcal{Z}$) in the optimal solution. One of the integer infeasible variables $y_{\hat{i}}$ is then chosen for *branching*, and two new sub-problems are thus added to the list of open nodes. By iterating the process a search tree is created and the algorithm continues until the list of open sub-problems is empty.

One of the key ingredients of the branch-and-bound procedure is the choice of the variable to branch-on. The classical rule is to choose the variable that has the largest fractional part, but this rule is often not very efficient. In this paper, we present two new rules specifically adapted to the portfolio optimization problems presented in Section 2. These two rules are respectively called *idiosyncratic risk* and *portfolio risk* branching and are described in the next sections.

3.2 Static branching rule: Idiosyncratic Risk Branching

The idiosyncratic risk branching rule is a static branching rule in which branching priorities are determined *a priori* (i.e., before the optimization is started).

For each integer decision variable, the branching priority is given by an integer π_i. At each node, the variable chosen for branching is the one, among the integer constrained variables with fractional value in the optimal solution of the current continuous relaxation, that has the highest priority. In case of a tie (i.e., when several candidate variables for branching have the same priority), the variable selected for branching is the one, among those with highest priority, that is the most fractional in the continuous relaxation.

It is important to recall that, in the optimization problems with buy-in threshold constraints (16) and with round lot constraints (18), there is a mapping between assets and integer decision variables: to each asset j corresponds a unique integer decision variable γ_j in (16) and δ_j in (18). In the context of mean-variance portfolio optimization problems, we propose to give the highest priority to the integer decision variable associated with the asset whose return has the greatest variance. We refer thereafter to this branching procedure as the *idiosyncratic risk branching* procedure. The intuition behind these priorities is that the asset with the largest variance is the one which has the most significant impact on the overall risk of the portfolio. Therefore, if the variance is the largest, the two sub-problems resulting from the branching are more likely to have an optimal value differing substantially from that of the parent node.

For the problems with diversification constraints, each integer decision variable is associated with a specific industrial sector. To each binary variable, and therefore to each sector, we assign a branching priority which is an increasing function of the sum of the variances of each asset stock related to the considered sector.

3.3 Dynamic branching rule: Portfolio Risk Branching

The portfolio risk branching rule is a dynamic branching rule, in which the branching priorities change at each node and tributary of the structure of the portfolio at the current node. Clearly, the branching variable is determined by relying upon a *dynamic*, *integrated* risk approach. The dynamics of the branching rule stems from the revision of the branching priorities at each node in the search tree, while its integrated risk approach derives from the fact that the branching priorities are a function of the specific contribution of each variable (asset) to the *overall* risk of the portfolio.

The dynamic feature is relevant since, in the course of the optimization process, a new optimal portfolio can potentially be constructed at each node in the branch-and-bound tree. Therefore, an iterative (at each node) evaluation of the contribution of each variable to the variance of the portfolio is desirable. As it will be detailed in the next subsections, it is possible to establish a direct correspondence between an integer decision variable and an asset. At each node in the branch-and-bound tree, we consider each integer variable whose optimal value in the current continuous relaxation is not integer feasible. For each such variable, we evaluate how the restoration of the integrality condition impacts (increases) the variance of the current portfolio. The variable whose integer feasibility restoration has the largest impact on the variance receives the highest priority, and is the one with respect to which we branch.

To carry out this evaluation, we approximate the problem at hand by a more simple disjunctive program with quadratic objective function and linear equality constraints which takes into account the integrality of only one variable:

$$\begin{aligned} & \min f(w) = w^T \Sigma w \\ & \text{subject to } Aw = b, \\ & \quad (w_i \le \underline{w}_i) \vee (w_i \ge \overline{w}_i)\,,\ i \in 1, \ldots, r \\ & \quad w \in \mathcal{R}^r \end{aligned} \quad . \tag{20}$$

Clearly, the problem above, and therefore the evaluation of the impact of the integer feasibility restoration, are obtained by omitting the non-linear term in the portfolio return constraint and relaxing the bounds on the variables.

In the next subsections, we give a precise description of how this approximation is obtained for each variant of the probabilistic Markowitz problem. Prior to this, we explain how the branching rule is applied in the general setting of (20).

Let w^* be the (continuous) optimal solution of (20), and let $\mathcal{L}_\lambda(w)$ be the Lagrangian function:

$$\mathcal{L}_\lambda(w) = f(w) + \lambda^T (Aw - b)\ . \tag{21}$$

We estimate the change in the objective value of (20) through the Lagrangian function. A movement of $\delta \in \mathcal{R}^r$ from w^* induces the following change in (21):

$$\begin{aligned} \mathcal{L}_\lambda(w^* + \delta) - \mathcal{L}_\lambda(w^*) &= (w^* + \delta)^T \Sigma (w^* + \delta) - {w^*}^T \Sigma w^* + \lambda^T (A\delta) \\ &= \delta^T \Sigma \delta + (2{w^*}^T \Sigma + \lambda^T A)\delta\ . \end{aligned}$$

Since w^* is optimal, it satisfies the KKT conditions:

$$\begin{aligned} 2{w^*}^T \Sigma + \lambda^T A &= 0 \\ \lambda(Aw^* - b) &= 0 \end{aligned}$$

which implies that $\mathcal{L}_\lambda(w^* + \delta) - \mathcal{L}_\lambda(w^*) = \delta^T \Sigma \delta$.

Let us consider a variable w_i with value w_i^*, such that $w_i^* \in [\underline{w}_i, \overline{w}_i]$. Branching on w_i creates two nodes: in each of them we add one of the constraints $w_i \le \underline{w}_i$ and $w_i \ge \overline{w}_i$. Using the procedure described above, we estimate the change in the Lagrangian of (20) by computing the two estimates δ_i^- and δ_i^+ defined by

$$\begin{aligned} \delta_i^- &= (w_i^* - \underline{w}_i) e_i^T \Sigma (w_i^* - \underline{w}_i) e_i = (w_i^* - \underline{w}_i)^2 \sigma_{ii} \\ \delta_i^+ &= (\overline{w}_i - w_i^*) e_i^T \Sigma (\overline{w}_i - w_i^*) e_i = (\overline{w}_i - w_i^*)^2 \sigma_{ii} \end{aligned} \tag{22}$$

where e_i is a vector whose components are all equal to 0 but the $i-$th one which is equal to 1.

By analogy to mixed-integer programming [37], we then combine these two estimates to obtain the *score* of variable w_i by taking a linear combination of the minimum and the maximum of the two [37]:

$$\delta_i = L \min(\delta_i^-, \delta_i^+) + U \max(\delta_i^-, \delta_i^+)\ . \tag{23}$$

We set the values of L to 1 and U to 2.

We calculate δ_i for all integer variables with fractional values in the optimal solution of the continuous relaxation, and we select as branching variable the one which has the highest score:

$$\hat{i} = \arg \max_{\{i: w_i^* \in (\underline{w}_i, \overline{w}_i)\}} \delta_i \ .$$

The quality of the branching scheme depends on the quality of the relaxation (20) with respect to the original problem. For the problems handled in this paper, it is easy to build such relaxations, and the computational experiments indicate that they are of good quality.

3.3.1 Problem with buy-in threshold constraints

In this section, we discuss the implementation of the dynamic portfolio risk branching rule in problem (16) in which the constraints (14) and (15) define the minimum proportion of available wealth K that must be invested in any active position.

In this case, we use the following formulation:

$$\begin{aligned} & \min w^T \Sigma w \\ \text{subject to } & \mu^T w + F_{(w)}^{-1}(1-p)\sqrt{w^T \Sigma w} \geq R \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & (w_i \leq 0) \vee (w_i \geq w_{\min}) \\ & w \in \mathcal{R}_+^{r+1} \end{aligned} \quad . \tag{24}$$

Note that this formulation is strictly equivalent to (16): imposing the condition $w_i \leq 0$ is equivalent to setting γ_i equal to 0 in (16), and imposing $w_i \geq w_{\min}$ is equivalent to setting γ_i equal to 1. The continuous relaxation is obtained by removing the disjunctive constraints.

The selection of the branching variable is performed by applying the scheme described in Section 3.3 to the following relaxation

$$\begin{aligned} & \min w^T \Sigma w \\ \text{subject to } & \mu^T w = R \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & (w_i \leq 0) \vee (w_i \geq w_{\min}) \\ & w \in \mathcal{R}^{r+1} \end{aligned}$$

of problem (24). The relaxation is obtained by transforming the portfolio return constraint into an equality constraint from which the non-linear component is dropped, and by removing the non-negativity constraints.

3.3.2 Problem with round lot constraints

The constraint $\gamma_i = \frac{K}{Mp_i} w_i$ establishes a direct correspondence between the continuous variables w_j and the integer ones γ_j in portfolio optimization problems with round lot constraints (18). Therefore, for a particular value of w^*, we use the following relaxation

$$\begin{aligned} \min\ & w^T \Sigma w \\ \text{subject to } & \mu^T w = R \\ & w_0 + \sum_{j=1}^{r} w_j = 1 \\ & (w_i \leq \left\lfloor \frac{K}{Mp_i} w_i^* \right\rfloor) \vee (w_i \leq \left\lceil \frac{K}{Mp_i} w_i^* \right\rceil) \end{aligned} \tag{25}$$

to select the branching variable in portfolio optimization problems with round lot constraints (18).

3.4 Analogy between branching strategies and criteria for credit risk

An interesting analogy can be established between the two branching rules proposed in this paper and the concepts of *marginal* and *standalone* risk of a security proposed by JP Morgan in its CreditMetrics framework [44] for quantifying the credit risk of portfolios. More precisely, there is a direct link between the portfolio risk branching rule and the marginal risk, on one hand, and between the idiosyncratic risk branching rule and the standalone risk, on the other hand.

CreditMetrics recommendation to hold a financial instrument is based on its marginal risk defined as the marginal increase in the portfolio risk that would result from adding this instrument to it. This approach takes into account the co-movements between the instrument considered for incorporation in the portfolio and those already included. The portfolio risk branching rule gives the highest branching priority to the asset and the associated integer variable for which the restoration of the integrality condition would most affect the risk of the portfolio. Both concepts are clearly linked to the diversification axiom and address the concentration risk, which is the accrued portfolio risk due to an exposure to one obligor or groups of correlated obligors (e.g., by industry, by location, etc.). The two approaches measure the portfolio volatility of an instrument or position and assign branching priorities (portfolio risk), or decide the inclusion of the asset (marginal risk) on that basis.

The standalone risk of an instrument is measured by its standard deviation and is independent of the correlation between the instrument and the other assets in the portfolio. Similarly, the idiosyncratic risk branching assigns the highest priority with respect to the integer variable associated with the asset which has the highest volatility, i.e. standard deviation.

4 Computational results

4.1 Test problems

To build the test bed for our approach, we use the daily return data of more than 600 stocks that have been part of Standard&Poor's 500 index between 1990 and 2004. The data accounts for the splits that the considered stocks have undergone in the period indicated above. In order to test the computational tractability of the solution method, we approximate the probability distribution of the expected returns (which is usually estimated through proprietary models by financial institutions [10]; see also the robust statistics and optimization literature) by the probability distribution of the returns. Using historical data, we compute the geometric mean and matrix of variance-covariance of the returns. This has no bearing on the computational results of our solution method which can be used for any probability distribution characterized by its first two moments.

Using those data, we build 36 portfolio optimization instances of various sizes (12 problems with 50 assets, 12 with 100 and 12 with 200) by randomly selecting the assets included in the problems. For each problem instance, we formulate three models corresponding to the trading constraints (buy-in threshold, round lot purchase, and diversification) considered in this paper. To model the problems with diversification constraints, we use the Global Industry Classification Standard (GICS) [57] developed by Standard&Poor's and Morgan Stanley Capital International to identify the industrial sector to which each company belongs. The GICS structure consists of 10 Sectors, 24 Industry Groups, 67 Industries and 147 Sub-Industries. The present study allocates each company to one of the 67 industries. The data come from the CRSP database and were obtained using the Wharton Research Database Service.

In each problem instance, the prescribed return level R is set equal to 7%, the fixed return of the money market is equal to 2% and the prescribed reliability level p, by which the investor wants the expected portfolio return to exceed the prescribed return level, is set to 85%. The asset returns are assumed to follow a normal distribution. The problem instances are modeled by using the AMPL modeling language. Table 1 reports the numbers of continuous and integer decision variables per type of models and per number of considered assets.

Table 1: Size of optimization models:
Number of variables n, non-linear variables n_{nl}, integer variables n_i, linear constraints m_l, non-linear constraint m_{nl} and non-zeroes in the Jacobian n_{jac}.

Models with	Number of Assets																	
	50						100						200					
	n	n_{nl}	n_i	m_l	m_{nl}	n_{jac}	n	n_{nl}	n_i	m_l	m_{nl}	n_{jac}	n	n_{nl}	n_i	m_l	m_{nl}	n_{jac}
Buy in threshold constraints	101	50	50	101	1	300	201	100	100	201	1	600	401	200	200	401	1	1200
Round lot Constraints	101	50	50	51	1	200	201	100	100	201	1	400	401	200	200	201	1	800
Diversification Constraints	82	50	32	67	1	293	148	100	48	99	1	538	264	200	64	129	1	982

In our experiments, we compare the results obtained with the standard branch-and-bound algorithm of the three solvers `MINLP_BB` [34], `CPLEX 10.1` [28] and `Bonmin` [7] to the results obtained with our specialized branch-and-bound algorithms implemented within the `Bonmin` framework.

`MINLP_BB` implements a branch-and-bound method for general non-linear programs and uses the sequential quadratic trust region algorithm called `filterSQP` [22] to solve the continuous relaxations. `CPLEX 10.1` is a commercial code that uses a branch-and-bound approach and that utilizes an interior point algorithm to solve second-order cone optimization problems. `Bonmin` [7, 3] is an open-source solver (available under the Common Public License) designed to solve to optimality general convex MINLPs. Among the several algorithms implemented within `Bonmin`, we have chosen to use, based on preliminary tests, a branch-and-bound algorithm that employs the general purpose interior point solver `Ipopt` [60] to solve the continuous relaxations and the branch-and-cut code `Cbc` to manage the tree search.

Some of the main differences between the three codes are:

- `MINLP_BB` uses an active set method for solving the continuous relaxation, while `Bonmin` and `CPLEX 10.1` use interior point algorithms (`MINLP_BB` has better warm-starting capabilities but needs more memory to solve large problems);

- `MINLP_BB` and `Bonmin` use general non-linear programming methods to solve the continuous relaxations while `CPLEX 10.1` uses a method dedicated to second-order cone programming;

- `MINLP_BB` uses the depth-first search strategy for choosing the next node to process in the tree search (i.e., the next node to be processed is the the deepest one), while `Bonmin`, by default, uses the best-bound approach (i.e., the next node to be processed is the one whose parent provides the smallest lower bound);

- `CPLEX 10.1` is the only solver to have advanced heuristic method for finding integer feasible solutions.

Since `Bonmin` is an open-source code, it can be very conveniently used to implement modifications of the general algorithm. We used this feature to implement the branching rules devised in Section 3. The idiosyncratic risk branching is implemented through the definition of branching priorities, while the portfolio risk branching is implemented in two specific branching rules for problems containing buy-in threshold constraints and round-lot constraints.

All tests were performed on an IBM IntellistationZ Pro with an Intel Xeon 3.2GHz CPU, 2 gigabytes of RAM and running Linux Fedora Core 3.

4.2 Evaluation of solution approaches

4.2.1 Model with buy-in threshold constraints

In this section, we analyze the computational results obtained for the problem instances containing buy-in threshold constraints. The experiments have been conducted by setting the minimum fraction of wealth ($w_{\min}$) to be invested in an asset (should the investor decide to include that asset in his portfolio) equal to 2%, 3% and 5% for the instances with 50, 100 and 200 stocks, respectively.

Table 2 reports the results obtained with the five algorithmic approaches listed below on the 36 problem instances with buy-in threshold constraints:

- `Bonmin`'s branch-and-bound algorithm with branching performed on the most fractional integer variable (i.e., the default branching rule in `Bonmin`),

- `Bonmin`'s branch-and-bound algorithm with the idiosyncratic risk branching rule (Section 3.2),

- `Bonmin`'s branch-and-bound algorithm with the portfolio risk branching rule (Section 3.3.1),

- `MINLP_BB`'s branch-and-bound algorithm,

- the `CPLEX 10.1` solver.

Table 2: Computational results for problems with buy-in thresholds constraints

	Most Fractional			Idiosyncratic Risk			Portfolio Risk			MINLP_BB			CPLEX 10.1		
Name	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes
050_01	*	97.4	1250	*	10.2	148	*	11.4	142	*	90.9	478	*	110.97	1781
050_02	*	23.9	161	*	2.7	39	*	2.3	30	*	30.9	506	*	11.21	138
050_03	*	62.9	860	*	7.1	114	*	6.9	96	*	30.3	2140	*	66.72	1126
050_04	*	31.4	433	*	5.2	87	*	6.2	86	*	30.2	1080	*	31.99	494
050_05	*	2194.7	167	*	5.0	81	*	4.3	58	*	30.8	312	*	11.96	138
050_06	*	40.1	563	*	23.7	393	*	22.1	316	*	30.5	1746	*	49.85	791
050_07	*	36.3	384	*	8.5	129	*	7.2	98	*	30.3	682	*	21.34	314
050_08	*	107.5	1438	*	6.5	109	*	6.1	86	*	90.4	3398	*	98.84	1663
050_09	*	49.7	224	*	3.6	55	*	4.5	52	*	30.2	492	*	14.52	192
050_10	*	37.5	384	*	6.7	98	*	8.4	96	*	30.4	576	*	22	346
050_11	*	43.9	527	*	7.4	123	*	8.8	104	*	30.7	2294	*	48.05	760
050_12	*	6.9	62	*	7.2	58	*	1.7	18	*	30.6	214	*	6.47	52
100_01	*	99.0	257	*	14.3	47	*	7.1	20	*	30.2	440	*	53.95	142
100_02	*	60.1	212	*	17.9	75	*	11.0	34	*	90.7	892	*	78.36	164
100_03	*	201.5	696	*	12.7	49	*	11.7	38	*	210.5	1788	*	184.08	504
100_04	*	40.9	147	*	16.0	64	*	11.6	36	*	90.3	1116	*	62.66	149
100_05	*	2993.1	9865	*	301.3	1201	*	360.2	1234	*	7890.9	56994	*	2712.09	9498
100_06	*	435.1	1560	*	89.7	368	*	60.2	206	*	1050.3	8396	*	464.01	1425
100_07	*	122.5	454	*	29.4	125	*	23.0	82	*	150.2	1266	*	106.56	302
100_08	*	59.8	218	*	22.4	95	*	15.1	54	*	90.1	804	*	69.48	168
100_09	*	1186.7	4321	*	147.7	575	*	101.5	362	*	1170.7	8872	*	1216.34	3915
100_10	*	150.6	539	*	24.5	85	*	17.4	62	*	150.4	1262	*	142.66	370
100_11	*	116.5	472	*	16.5	59	*	7.6	24	*	90.1	698	*	63.2	176
100_12	*	716.9	2487	*	108.8	412	*	97.6	348	*	870.6	6778	*	834.54	2660
200_01	NS	>10800	9342.00	*	488.46	388.00	*	413.30	358.00	13.561%	>10800	13100	15.619%	>10800	4338
200_02	NS	>10800	8816.00	*	2054.08	1434.00	*	1768.18	1462.00	5.690%	>10800	17400	8.875%	>10800	4487
200_03	NS	>10800	8935.00	*	1092.79	785.00	*	636.60	536.00	1.322%	>10800	23400	4.036%	>10800	4600
200_04	*	655.83	509.00	*	152.19	103.00	*	90.12	72.00	*	750.61	2230	*	1323.43	409
200_05	NS	>10800	9249.00	*	1960.07	1454.00	*	1766.89	1540.00	12.261%	>10800	18100	9.906%	>10800	4742
200_06	NS	>10800	9051.00	*	483.93	347.00	*	357.01	306.00	12.321%	>10800	17700	11.555%	>10800	5158
200_07	NS	>10800	8994.00	*	2080.34	1431.00	*	1252.50	1066.00	7.194%	>10800	17600	6.067%	>10800	5197
200_08	NS	>10800	9078.00	*	3240.16	2203.00	*	2149.53	1812.00	2.424%	>10800	17900	2.996%	>10800	5352
200_09	NS	>10800	9196.00	*	1493.92	1105.00	*	1801.83	1544.00	4.584%	>10800	17800	6.826%	>10800	4789
200_10	*	1891.71	1603.00	*	342.05	253.00	*	269.86	232.00	*	2610.87	7314	*	5228.77	1705
200_11	*	1539.33	1321.00	*	336.25	253.00	*	307.76	260.00	*	1470.52	4668	*	3671.4	1316
200_12	NS	>10800	8867.00	*	1681.56	1186.00	*	1026.66	832.00	4.142%	>10800	18000	3.211%	>10800	5184

The above solution approaches will thereafter be referred to as *MF, IR, PR, MBB*, and *CP*, respectively.

For each ”combination” of problem instance and solution approach, Table 2 reports

- the quality of the best obtained solution (columns 2, 5, 8, 11, 14). We use the acronym *NS* to indicate that no feasible integer was found. We report the value of the mixed-integer optimality gap when the best integer solution found is not proven to be optimal by the algorithm, and use the symbol ”*” when the optimality is proven;

- the computing time (in CPU seconds) needed to solve the problem to optimality (columns 3, 6, 9, 12, 15). If this latter cannot be found within the allowed computing time (3 hours), the entry in the table reads ”>10800”;

- the number of explored nodes in the branch-and-bound tree (columns 4, 7, 10, 13, 16).

First, we comment on the accuracy of the found solutions. It is well known that the structure of the variance-covariance matrix of returns often leads to numerical difficulties [6]. While we cannot establish the optimality of the obtained solutions (outside of the tolerances of the solvers), we can compare the values of the optimal solutions obtained with `Bonmin`, `MINLP_BB` and `CPLEX 10.1`. We recall that those solvers are based on very different continuous non-linear programming methods. We observe that the relative difference between the optimal solutions found by `Bonmin`, `MINLP_BB` and `CPLEX 10.1` are in the order of 10^{-4} except for problem 050_1 where it is $8.56 * 10^{-3}$. We note that the solution found by the three variants of `Bonmin` are always identical for these instances as well as for all the other instances in the paper. It is also worth pointing out that the solution claimed by `Bonmin` always has a better objective value than the one claimed by `MINLP_BB` and `CPLEX 10.1` on these problems.

The instances with 50 and 100 assets do not really allow us to discriminate the four solution approaches in terms of the quality of the solution. Indeed, Figure 1 shows that the optimal solution is found by each approach for every 50-stock and 100-stock problem instance.

For the most difficult problems containing 200 stocks, the solution approaches *IR* and *PR* utilizing the two new branching rules clearly dominate *MP*, *MBB* and *CP*. The former two approaches solve all the instances to optimality, while the latter three solve only 25% of those instances to optimality. It is also worth noting that *MP* does not find any integer feasible solution when it cannot find the optimal one, while *MBB* and *CP* always find an integer feasible solution, and the average value of the final optimality gap is equal to 7.06% (7.68%) for *MBB* (*CP*). This might be due to the node selection rule used in *MP* (best-bound) and to the fact that no primal heuristic is implemented in `Bonmin`.

Another interesting point is to compare the time to solve a node by each of the three approaches. For the three branch-and-bound algorithms (`MINLP_BB`, `CPLEX 10.1` and `Bonmin`), the largest fraction of the node processing time is spent solving the continuous relaxation. Since `filterSQP` is an active set method, it has the clear advantage of having more efficient warm-start capabilities, and this is corroborated by the experiments: `filterSQP` needs on average 0.24 sec./node (on the test set) versus 0.52 sec./node for `Bonmin` and 0.95 sec./node for `CPLEX 10.1`. Both `CPLEX 10.1` and `Bonmin` use interior point methods to solve the continuous relaxation. Provided that `CPLEX 10.1`'s interior point method is specialized for the solution of second-order cone problems, it is surprising to observe that `Bonmin` is, on average, almost twice faster than `CPLEX 10.1` per node. This might be due to the fact that `CPLEX 10.1` applies more integer programming methods (primal heuristic, bound tightening, etc.) than `Bonmin`.

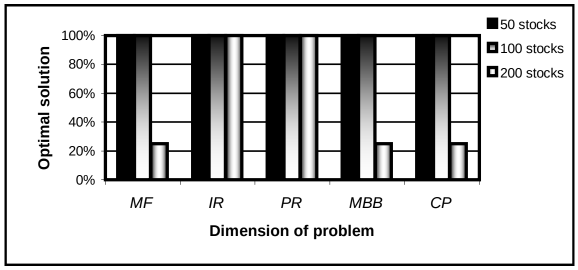

Figure 1: Quality of solution for problems with buy-in constraints

Figures 2 and 3 display the average total computing time for each combination of solution approach and size of problem instance.

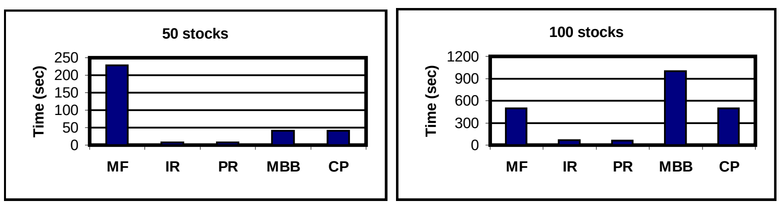

Figure 2: Average computing time for 50-stock and 100-stock instances with buy-in constraints

In Figure 3, the left-hand side graph shows the average time computed over 200-stock instances, while the right-hand side one shows the average time computed over the only instances that could be solved to optimality by every solution approach. It is clear that the *IR* and *PR* solution approaches, relying respectively on the idiosyncratic and portfolio risk branching rules, are, regardless of the size of the problem, much faster than *MP*, *MBB* and *CP*. The *PR* solution approach is slightly faster than *IR*, and is on average:

- more than 5 (respectively, 16 and 8) times faster than *MBB* on the 50-stock (respectively, 100- and 200-stock) instances.
- more than 5 (respectively, 8 and 9) times faster than *CP* on the 50-stock (respectively, 100- and 200-stock) instances.

Figure 4 shows the evolution of the average computing time (for all instances on the right-hand side, for instances solved to optimality on the left-hand side). We can see that *PR* and *IR* scale very well: the rhythm at which their average computing time increases is very reasonable, therefore indicating their applicability to problems of larger size. This must be contrasted to the *MF*, *MBB* and *CP* approaches for which the computing time seems to increase exponentially with the number of assets.

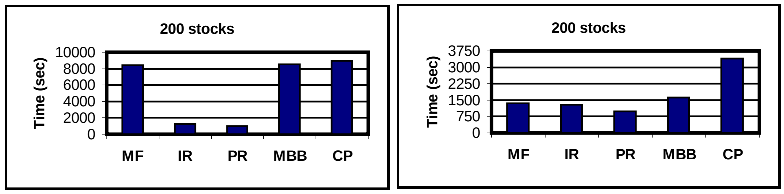

Figure 3: Average computing times for 200-stock instances with buy-in constraints

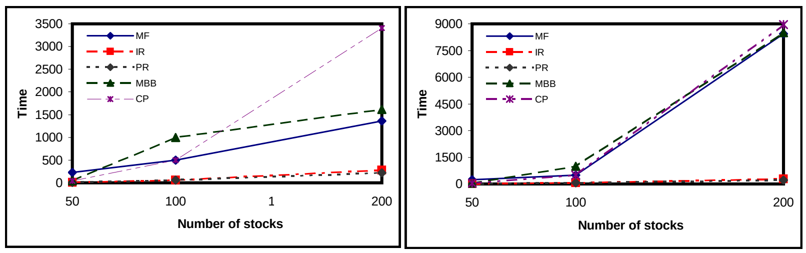

Figure 4: Buy-in constraints: computing time as a function of dimensionality

4.2.2 Model with round lot constraints

Table 3 reports the computational results for the 36 problem instances with round lot constraints and in which the investor is constrained to buy shares by multiples of M, set equal to 100 in our experiments. Table 3 provides the same outputs (optimality gap, CPU time, number of nodes) and uses the same notations as those in Table 2. The following five integer solution methods have been tested:

- `Bonmin`'s branch-and-bound algorithm with branching on the most fractional integer variable,
- `Bonmin`'s branch-and-bound algorithm with the idiosyncratic risk branching rule (Section 3.2),
- `Bonmin`'s branch-and-bound algorithm with the portfolio risk branching rule (Section 3.3.2),
- `MINLP_BB`'s branch-and-bound algorithm,
- the `CPLEX 10.1` solver.

Figure 5 shows that the *PR* solution approach using the dynamic portfolio risk branching rule is by far the most robust method for problems with round lot constraints. The *PR* method is the only one solving to optimality *all* 100-assets instances, while *CP* (respectively, *MBB*, *IR*, *MF*) finds the optimal solution for 66.67% (respectively, 58.33%, 58.33%, 66.67%) of those instances. Even more striking is the fact the *PR* method solves to optimality 83% of the 200-asset problem instances, while *none* of the three other methods can solve to optimality *any* of the 200-asset instances. A few additional comments are in order. First, the *MF* approach does not find any feasible integer solution for any of the problem instances that it cannot solve to optimality (i.e., 33% and 100% of the 100-stock and 200-stock instances, respectively). The *IR* does not find any integer feasible solution for any of the 200-problem instances. This is again, most probably, due to the fact that `Bonmin` does not have heuristic methods.

Table 3: Computational results for problems with round lots constraints

	Most Fractional			Idiosyncratic Risk			Portfolio Risk			MINLP_BB			CPLEX 10.1		
Name	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes
050_01	*	455.32	6888	*	1227.34	13576	*	70.38	1202	*	101.42	5154	*	483.98	9517
050_02	*	214.81	3002	NS	10800	35026	*	34.56	560	*	38.45	2958	*	314.26	5470
050_03	*	988.17	14976	*	230.35	3299	*	136.05	2304	*	193.77	17744	*	711.76	12511
050_04	*	55.97	870	*	220.78	1245	*	20.57	382	*	12.2	1004	*	127.64	2586
050_05	*	148.16	1990	*	114.46	1623	*	49	856	*	21.47	1700	*	63.05	1611
050_06	*	415.44	6757	*	690.34	10432	*	39.96	860	*	103.46	7224	*	421.07	7803
050_07	*	2455.82	32870	*	317.15	2211	*	41.13	812	*	551.7	42108	*	2052.81	39989
050_08	*	2154.31	17064	*	55.67	836	*	414.97	1437	*	394.73	28082	*	998.62	19527
050_09	*	192.89	2646	*	943.3	9640	*	46.75	868	*	6.41	710	*	251.45	4867
050_10	*	364.39	3493	*	41.6	684	*	158.22	622	*	0.6	68	*	64.21	1920
050_11	*	149.53	2143	*	292.27	2859	*	20.12	408	*	33.36	1994	*	165.38	3898
050_12	*	77.25	1090	*	85.3	444	*	11.08	202	*	16.27	1104	*	56.03	1731
100_01	*	6665.12	17858	*	1654.4	5614	*	603.62	2062	*	1217.6	15030	*	2908.47	11090
100_02	NS	>10800	45827	NS	>10800	18370	*	654.59	2662	0.000%	>10800	84600	0.090%	>10800	40000
100_03	*	1124.4	4554	*	2295.49	3954	*	69.6	274	*	407.77	5614	*	3365.43	11591
100_04	*	1609.51	5267	*	660.25	1144	*	786.76	2333	*	154.97	2014	*	2535.39	9957
100_05	NS	>10800	42014	NS	>10800	34790	*	1416.25	5848	0.503%	>10800	94700	0.300%	> 10800	42000
100_06	NS	>10800	33462	*	5419.46	9120	*	1061.64	4176	0.000%	>10800	99700	0.180%	>10800	41000
100_07	*	5242.64	16573	NS	>10800	36278	*	171.02	714	*	1691.5	21866	*	5436.22	23086
100_08	*	732.54	2784	NS	>10800	36215	*	113.71	520	*	704.72	10136	*	3246.26	13047
100_09	*	2632.15	9770	*	4258.54	7196	*	106.84	452	*	3271.11	47086	0.020%	>10800	42000
100_10	*	6222.88	22463	*	3344.64	5830	*	617.29	2430	*	2502.73	29108	*	6771.1	28450
100_11	*	1841.28	6570	*	5345.76	21646	*	563.5	2318	*	82.29	1224	*	1674.96	6983
100_12	NS	>10800	41882	NS	>10800	30733	*	273.88	1006	0.274%	>10800	100300	0.420%	>10800	43000
200_01	NS	>10800	8856	NS	>10800	4805	*	5169.96	4226	2.880%	>10800	24400	3.000%	>10800	6000
200_02	NS	>10800	7249	NS	>10800	4690	NS	>10800	7498	0.849%	>10800	19100	1.151%	>10800	4132
200_03	NS	>10800	9038	NS	>10800	4648	*	1677.1	1428	0.000%	>10800	23900	0.424%	>10800	4500
200_04	NS	>10800	8665	NS	>10800	4432	*	2729.16	2132	0.602%	>10800	18900	0.412%	>10800	4498
200_05	NS	>10800	8746	NS	>10800	4867	*	3127.63	2414	1.420%	>10800	24400	1.384%	>10800	3501
200_06	NS	>10800	7850	NS	>10800	4723	*	9116.37	7074	2.027%	>10800	23900	2.127%	>10800	4400
200_07	NS	>10800	8286	NS	>10800	4836	*	8206.51	6128	0.947%	>10800	24000	1.572%	>10800	4100
200_08	NS	>10800	8771	NS	>10800	5027	NS	>10800	9139	2.000%	>10800	24300	0.952%	>10800	4601
200_09	NS	>10800	8652	NS	>10800	5129	*	3205.84	2556	2.918%	>10800	19300	2.676%	>10800	3852
200_10	NS	>10800	7695	NS	>10800	4362	*	2950.6	2054	0.648%	>10800	18800	0.873%	>10800	3639
200_11	NS	>10800	8378	NS	>10800	5273	*	1203.23	934	NS	>10800	29147	2.066%	>10800	3801
200_12	NS	>10800	8380	NS	>10800	4602	*	7616.07	6016	0.662%	>10800	24100	0.359%	>10800	3562

The average optimality gap amounts to

- 0.194% (100-stock instances) and 1.359% (200-stock instances) with `MINLP_BB`;
- 0.202% (100-stock instances) and 1.416% (200-stock instances) with `CPLEX 10.1`.

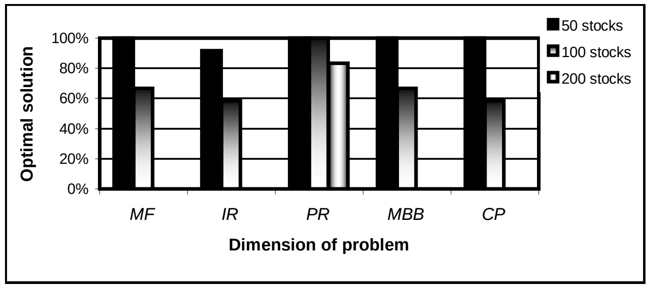

Figure 5: Quality of solution for problems with round lot constraints

Figure 6 shows that *PR* is not only the most robust but also the fastest regardless of the dimensionality of the problem. The average computing times (i.e., irrespective of whether one considers all instances [left-hand side in Figure 6], or only those solved to optimality by all approaches [right-hand side in Figure 6]) of *PR* are significantly lower than those of the other methods. The difference in speed between *PR* and any of the other three methods increases with the size of the problem; indeed, *PR* is

- 5.47 (respectively, 1.41, 4.41, 7.35) times faster than `CPLEX 10.1` (respectively, *MBB*, *IR*, *MF*) on the 50-stock instances;
- 8.55 (instances solved to optimality) and 12.41 (all instances) times faster than `CPLEX 10.1` on the 100-stock instances;
- 2.33 (instances solved to optimality) and 8.26 (all instances) times faster than *MBB* on the 100-stock instances;
- 6.24 (instances solved to optimality) and 10.75 (all instances) times faster than *MF* on the 100-stock instances;
- 9.06 (instances solved to optimality) and 11.95 (all instances) times faster than *IR* on the 100-stock instances.

No speed comparison can be drawn for the 200-stock instances since *PR* is the only method solving some (83%) of them to optimality.

Finally, we note that the relative difference between the optimal values found by `Bonmin`, `CPLEX 10.1` and `MINLP_BB` is always smaller than 10^{-4}.

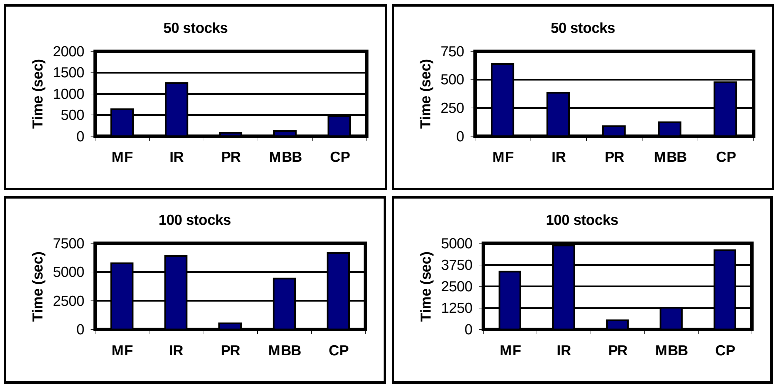

Figure 6: Average computing time for instances with round lot constraints

4.2.3 Model with diversification constraints

The results displayed in Table 4 are related to the 36 problem instances with cardinality-type diversification constraints.

The results have been obtained by setting L_{min} (the minimum number of sectors in which the investor must allocate his capital) to 10, 15 and 20 for the problem instances comprising 50, 100 and 200 assets, respectively, and by setting s_{min} (minimal position in any of the K_{min} sectors) to 1% for all problem instances. The results obtained with the following four integer solution methods

- `Bonmin`'s branch-and-bound algorithm with branching on the most fractional integer variable,
- `Bonmin`'s branch-and-bound algorithm with the idiosyncratic risk branching rule (Section 3.2),
- `MINLP_BB`'s branch-and-bound algorithm,
- the `CPLEX 10.1` solver.

are given in Table 4.

The results in Table 4 indicate that the four methods above solve to optimality all 36 instances in very limited computing time. On the largest problem instances, the average computing times for the slowest and fastest methods (respectively *CP* and *MBB*) are equal to 429 sec and 69 sec. Clearly, the problems with diversification constraints appear the easiest to solve.

4.3 Impact of integer trading constraints

We discuss below the impact on the various types of integer trading constraints. In particular, we analyze

- the difficulty of solving the problem associated with each type of constraints. The difficulty is evaluated with respect to the average computing time per type of models and for each problem size (50, 100, 200 stocks). Figure 7 shows that the computational time is an increasing function in the number of stocks, and highlights the following hierarchy in terms of problem complexity: problem with (1) cardinality, (2) buy-in threshold, and (3) round lot constraints.

Table 4: Computational results for problems with diversification constraints

Name	Most Fractional			Idiosyncratic Risk			MINLP_BB			CPLEX 10.1		
	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes	Final Gap	Time	#nodes
50_01	*	18.37	322	*	2.30	33	*	4.184	176	*	23.99	342
50_02	*	0.76	11	*	2.34	23	*	0.244	12	*	2.92	10
50_03	*	1.85	34	*	2.34	38	*	1.096	82	*	3.89	34
50_04	*	1.28	27	*	1.62	27	*	0.632	70	*	2.74	10
50_05	*	1.58	26	*	1.88	27	*	0.272	18	*	3.26	26
50_06	*	2.25	44	*	2.34	37	*	1.436	110	*	4.74	42
50_07	*	3.84	66	*	3.02	37	*	1.516	114	*	6.37	66
50_08	*	10.55	216	*	4.24	67	*	3.552	246	*	16	244
50_09	*	1.90	34	*	15.57	33	*	1.816	144	*	4.43	34
50_10	*	8.72	90	*	2.36	31	*	1.428	106	*	8.53	90
50_11	*	1.97	36	*	3.02	47	*	1.008	64	*	5.54	52
50_12	*	1.79	27	*	2.66	33	*	0.176	10	*	1.94	4
100_01	*	4.10	12	*	10.76	24	*	2.536	22	*	16.57	17
100_02	*	9.45	38	*	12.90	37	*	3.42	40	*	20.03	38
100_03	*	11.27	39	*	13.34	39	*	4.056	54	*	20.25	38
100_04	*	1.76	5	*	1.63	3	*	0.54	4	*	7.65	2
100_05	*	21.98	79	*	24.30	67	*	16.749	162	*	46.86	117
100_06	*	17.21	64	*	23.73	67	*	16.489	182	*	30.15	64
100_07	*	2.86	11	*	6.13	17	*	0.808	6	*	9.14	4
100_08	*	19.78	77	*	13.96	41	*	4.164	40	*	24.57	43
100_09	*	7.61	27	*	24.53	65	*	3.532	28	*	14.61	20
100_10	*	11.70	43	*	13.74	37	*	4.576	42	*	19.58	16
100_11	*	7.94	37	*	12.44	39	*	5.78	82	*	13.42	10
100_12	*	20.15	81	*	11.84	33	*	14.657	130	*	32.5	76
200_01	*	92.41	89	*	77.63	49	*	43.095	74	*	251.61	68
200_02	*	178.43	163	*	167.92	101	*	123.1	226	*	599.88	163
200_03	*	83.82	75	*	128.27	81	*	29.23	58	*	282.74	64
200_04	*	93.50	70	*	76.43	39	*	51.823	84	*	282.88	81
200_05	*	170.03	149	*	134.40	83	*	84.557	156	*	595.06	146
200_06	*	64.90	61	*	69.39	43	*	36.658	62	*	225.39	58
200_07	*	46.18	41	*	95.21	59	*	23.905	52	*	221.81	38
200_08	*	248.74	236	*	140.52	89	*	137.233	328	*	808.5	234
200_09	*	293.05	251	*	265.64	159	*	163.938	274	*	1039.41	270
200_10	*	19.39	15	*	39.97	23	*	7.388	12	*	80.27	10
200_11	*	190.53	175	*	171.14	107	*	113.751	232	*	663.95	171
200_12	*	24.06	21	*	53.02	29	*	8.329	16	*	83.58	10

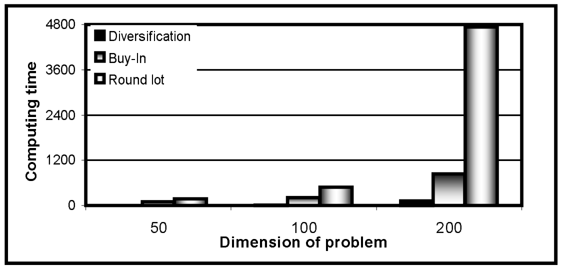

Figure 7: Average computing time per model type and problem dimension

The largest problems (i.e., 200-stock instances) with diversification constraints require less computing time on average than the least complex (i.e., 50-stock instances) problems with round lot constraints. The accrued complexity of these latter is due to the presence of general integer variables which implicitly require the detention of an integer number of shares of any asset included in the optimal portfolio.

- the impact of the buy-in threshold constraints. Table 5 presents detailed results about the composition of the optimal portfolio for each combination of model type (without integer constraints, with diversification, round lot and buy-in threshold constraints) and problem size. The notation N^P and N^{SP} respectively denote the average number of positions in the optimal portfolio and the average number of positions which are greater than the threshold imposed by the buy-in constraints. The threshold $w_{\min}$ is equal to 2%, 3% and 5% for the 50-, 100- and 200-stock instances, respectively. Table 5 shows that the buy-in constraints drastically change the structure of the optimal portfolio. The optimal portfolio with buy-in constraints is less diversified than the optimal portfolio obtained with any of the other three approaches. The optimal portfolio with buy-in constraints has positions in 16, 24 and 10 assets for 50-, 100-, and 200-stock instances, respectively. These numbers must be contrasted to those of the optimal portfolios without any integer constraints (24, 30, 34), with diversification constraints (26, 37, 41), and with round lot constraints (24, 28, 30).

Table 5: Concentration effect of buy-in threshold constraint

		No integer constraint (7)	Diversification constraint (19)	Round lot constraint (18)	Buy-in threshold constraint (16)
50-stock	N^P	24	26	24	16
instances	N^{SP}	14	16	13	0
($w_{min} = 0.02\%$)	N^{SP}/N^P	58.33%	61.54%	54.17%	0%
100-stock	N^P	30	37	28	24
instances	N^{SP}	13	21	11	0
($w_{min} = 0.03\%$)	N^{SP}/N^P	43.33%	56.76%	39.29%	0%
200-stock	N^P	34	41	30	10
instances	N^{SP}	32	39	28	0
($w_{min} = 0.05\%$)	N^{SP}/N^P	94.12%	95.12%	93.33%	0%

- the impact of the diversification constraints. In addition to constraining the holding of positions in a pre-defined number of industrial sectors, another effect of the diversification constraints, as shown by Table 5, is that the investor detains positions in a larger number of assets (at least, on average, 20.5% of the available assets) and detains a larger number of small positions (at least, on average, 56.76%).

- the impact of the round lot constraints. The effect of the requirement to buy shares by large lots is to limit the number of active positions. This number is smaller than that for the model without integer constraints and with diversification constraints.

5 Application by finance industry

The flexibility of our solution framework and its relevance for asset managers can be illustrated by its utilization by the Private Banking Group of an international bank (ING) with over 1050 billion US$ in total assets.

Within its new "Absolute Return" investment program designed to individuals, the Private Banking Group proposes investments in so-called Fund-of-Funds (FoF). To build these FoFs, investment managers at ING identified the funds to be included in the market universe for the FoFs. The funds belong to seven fund categories (short-term deposits, currencies, equity funds, bond funds, commodity funds, real estate funds, and specialized funds) which are themselves divided into sub-categories (Table 6).

Table 6: Asset classes and sub-classes

Classes	Short-term Deposits	Currencies	Equity Funds	Bond Funds	Commodity Funds	Real Estate Funds	Specialized Funds
Sub-Classes		US$ Euro Japanese Yen	North America Europe Asia	Government Inflation-Linked Investment-Grade Corporate High Yield Corporate Structured Credits Convertible Emerging Market	Energy Metals Agricultural Live Stock	North America Europe Asia	Equity Hedge Directional Trading Event Driven Relative Value

The FoF positions are determined with the help of an optimization model which uses the risk measure discussed in this paper (i.e., the minimization of the risk of the portfolio provided that the FoF expected return exceeds a threshold return level with a certain probability) and the associated optimization model has the same characteristics as (16). In particular, it includes buy-in threshold constraints imposing a lower bound on each individual position, the budget constraint, and the no short-selling constraints plus additional diversification, liquidity and currency constraints modeled as linear ones. *Class* and *subclass* diversification constraints impose a lower and an upper bounds on the quantity of the capital invested per class and subclass. *Currency* constraints limit the amount invested in assets traded in each currency (US$, Euro, Japanese Yen). *Liquidity constraints* ensure that a minimal proportion of the capital is invested in assets with weekly or monthly liquidities. The resulting model is solved using the portfolio risk branch-and-bound algorithm described in this paper. Four long-only absolute return fund-of-funds (`Serenity VSX 10.I CAP EUR`, `Serenity VSX 10.I CAP USD`, `Serenity VSX 5.I CAP EUR` and `Serenity VSX 5.I CAP`

USD), which differ in terms of the threshold return level and the probability by which their expected return exceeds it, have been constructed using our approach.

6 Conclusion

In this paper, we study the probabilistic Markowitz mean-variance portfolio optimization model with integer-based trading constraints. We consider real-world trading constraints, such as the need to diversify the investments in a number of industrial sectors, the non-profitability of holding small positions, or the constraint of buying stocks by lots, which are modeled with integer variables. We account for the uncertainty in the estimation of the expected asset return through the introduction of a stochastic constraint ensuring that the expected return of the portfolio exceeds the prescribed return with a high confidence level. We derive stochastic integer formulations for each type of trading constraints, show under which conditions their continuous relaxations are convex, taking the form of second-order integer programming problems, and develop exact solution techniques.

A key contribution of this paper is that it develops an *exact* solution approach for portfolio optimization problems in which uncertainty in the estimate of the expected return and real-life market restrictions modeled with integer constraints are *simultaneously* considered. The joint presence of integrality restrictions and of a non-linear, probabilistic constraint explains the complexity of solving such problems, for which very few solvers can be efficiently used.

The proposed solution approach is based on two new branching strategies that we implement in a non-linear branch-and-bound algorithm. The first one is a static branching rule, called idiosyncratic risk branching, while the second one is an integrated, dynamic branching rule, called portfolio risk branching. The latter updates, at each node in the branch-and-bound tree, the branching priorities given to the integer variables depending on their impact on the variance of the portfolio.

We evaluate the efficacy of five exact integer solution approaches on 36 problem instances containing up to 200 assets and constructed using the stocks included in the S&P 500 Index. We have not found any other computational study that considers so many assets for a stochastic portfolio optimization model subject to integer constraints. Computational results show that the solution approach using the portfolio risk branching rule is the most performing one, both in terms of speed and robustness (i.e., percentage of problems solved to optimality), and that it scales well. The results attest the marked superiority of our approach with respect to the MINLP_BB and CPLEX 10.1 solvers, and highlight the computational contribution of our approach. Another recent computational study [59] has also shown the limited efficiency of the CPLEX 10.1 solver to handle mixed integer conic optimization problems. Those results clearly suggest the importance and need of developing efficient solution methods for such optimization problems.

We observe that, for credit risk (marginal risk rule developed by J.P. Morgan [44]) as well as for portfolio selection (portfolio branching strategy), it is preferable to adopt an integrated approach considering the composition of the entire portfolio and accounting for the diversification axiom. We also derive a hierarchy of the integer trading constraints and give insights about the impact (concentration effect) of the buy-in threshold and diversification constraints. Finally, the relevance and interest of the proposed approach are shown through its implementation for the construction by a major financial group of four long-only absolute return fund-of-funds constructed.

The algorithmic results presented in this paper pave the way for multiple extensions. We would like to point out and test the applicability of the solution approach to other risk measures, in particular value-at-risk and first-order stochastic dominance. The scalability of the proposed solution approaches could lead to their application to problems of larger dimension. The running time could be further reduced by relying on a second-order cone programming solver to optimize the continuous relaxations of the second-order cone problems (i.e., polynomial running time) at each node in the branch-and-bound tree. Branch-and-cut solution approaches could also be considered. Other trading constraints (i.e., "transaction cost", "tax lot", "maximum number of transaction" constraints, etc.) leading to the formulation of other types of second-order cone problems with integer variables deserve attention.

Note that buy w_j^+ or sell w_j^- rebalancing decisions [42] modeled through balance constraints

$$w_j^0 + w_j^+ + w_j^- = w_j \ ,$$

where w_j^0 is the initial position in asset j, and coupled with a turnover constraint

$$\sum_{j=0}^{r} (w_j^+ + w_j^-) \leq t \ ,$$

where t is the turnover upper bound, can be handled by the proposed solution method. The relaxation of the integrality conditions gives a deterministic equivalent which is also a second-order cone optimization problem. The same observation applies to stochastic integer portfolio optimization problems of the same form subject to proportional transaction costs a_j (in addition to the rebalancing constraints). The transaction costs for rebalancing the initial positions is given by $\sum_{j=1}^{r} a_j \left(w_j^+ + w_j^- \right)$, the budget constraint becomes

$$\sum_{j=0}^{r} w_j + \sum_{j=0}^{r} a_j \left(w_j^+ + w_j^- \right) = 1 \ ,$$

and the constraint (6) now requires the expected return of the portfolio, after payment of the transaction costs, to be greater or equal to the prescribed return level

$$\mu^T w + F_{(w)}^{-1}(1-p)\sqrt{w^T \Sigma w} - \sum_{j=0}^{r} a_j \left(w_j^+ + w_j^- \right) \geq R \ .$$

Our future research will be devoted to other forms of transaction costs involving integer decision variables (application of fixed transaction cost component if any transaction regarding an asset j is carried out), modelled as concave functions, and to market impact models in which, in addition to the transaction costs themselves (e.g., brokerage commissions), the difference between the transaction price and what the market price would have been in the absence of the transaction is taken into account. Another line of future research relates to the study of portfolio selection problems satisfying the same risk criterion and the same trading constraints, but in which the risk of assets is estimated through a factor risk model [11, 42].

References

[1] Artzner P., Delbaen F., Eber J.-M., Heath D. 1999. Coherent Measures of Risk. *Mathematical Finance* 9 (3), 203-228.

[2] Avérous J., Meste M. 1997. Skewness for Multivariate Distributions: Two Approaches. *The Annals of Statistics* 25 (5), 1984-1997.

[3] Basic Open Source Solver for Mixed Integer Nonlinear Programming: *https://projects.coin-or.org/*`Bonmin`

[4] Bawa V.S., Brown S.J., Klein R.W. 1979. *Estimation Risk and Optimal Portfolio Choice*. North-Holland. Amsterdam, The Netherlands.

[5] Ben-Tal A., Nemirovski A. 2002. Robust Optimization - Methodology and Applications. *Mathematical Programming* 92, 453-480.

[6] Bienstock D. 1996. Computational Study of a Family of Mixed-Integer Quadratic Programming Problems. *Mathematical Programming* 75, 121-140.

[7] Bonami P., Biegler L.T., Conn A.R., Cornuéjols G., Grossmann I.E., Laird C.D., Lee J., Lodi A., Margot F., Sawaya N., Wachter A. 2008. An Algorithmic Framework for Convex Mixed-integer Nonlinear Programs. *Discrete Optimization*. In Press.

[8] Broadie M. 1993. Computing Efficient Frontiers Using Estimated Parameters. *Annals of Operations Research* 45, 21-58.

[9] Byrne P., Lee S. 2004. Different Risk Measures: Different Portfolio Compositions? *Journal of Property Investment & Finance* 22 (6), 501-511.

[10] Ceria S., Stubbs R.A. 2006. Incorporating Estimation Errors into Portfolio Selection: Portfolio Construction. *Journal of Asset Management* 7 (2), 109-127.

[11] Chan L.C.K., Karceski J., Lakonishok J. 1999. On Portfolio Optimization: Forecasting Covariances and Choosing the Risk Model. *The Review of Financial Studies* 12 (5), 937-974.

[12] Charnes A., Cooper W.W. 1959. Chance Constrained Programming. *Management Science* 6 (1), 73-79.

[13] Cheon M.-S., Ahmed S., Al-Khayyal F. 2006. A Branch-Reduce-Cut Algorithm for the Global Optimization of Probabilistically Constrained Linear Programs. *Mathematical Programming* 108 (2-3), 617-634.

[14] Chopra V., Ziemba W.T. 1993. The Effects of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice. *Journal of Portfolio Management* 19, 6-11.

[15] Corazza M., Favaretto D. 2007. On the Existence of Solutions to the Quadratic Mixed-Integer Mean-Variance Portfolio Selection Problem. *European Journal of Operational Research* 176, 1947-1960.

[16] Cornuejóls G., Tütüncü R. 2007. *Optimization Methods in Finance*. Cambridge University Press.

[17] Costa O.L.V., Paiva A.C. 2002. Robust Portfolio Selection Using Linear-Matrix Inequality. *Journal of Economic Dynamics & Control* 26, 889-909.

[18] Dentcheva D., Ruszczyński A. 2003. Optimization with Stochastic Dominance Constraints. *SIAM Journal on Optimization* 14, 548-566.

[19] Dentcheva D., Ruszczyński A. 2006. Portfolio Optimization with Stochastic Dominance Constraints. *Journal of Banking and Finance* 30 (2), 433-451.

[20] El Ghaoui L., Oks M., Oustry F. 2003. Worst-Case Value-at-Risk and Robust Portfolio Optimization: A Conic Programming Approach. *Operations Research* 51 (4), 543-556.

[21] Feinstein C.D., Thapa M.N. 1993. Notes: A Reformulation of a Mean-Absolute Deviation Portfolio Optimization Model. *Management Science* 39, 1552-1553.

[22] Fletcher R., Leyffer, S. 1998. User Manual for FilterSQP. *Numerical Analysis Report NA/181*, Dundee University.

[23] Föllmer H., Schied A. 2002. Convex Measures of Risk and Trading Constraints. *Finance and Stochastics* 6, 429-447.

[24] Frangioni A., Gentile C. 2006. Perspective Cuts for a Class of Convex 0-1 Mixed Integer Programs. *Mathematical Programming* 106,225-236.

[25] Goldfarb D., Iyengar G. 2003. Robust Portfolio Selection Problems. *Mathematics of Operations Research* 28, 1-38.

[26] Hanoch G., Levy H. 1969. The Efficiency Analysis of Choices Involving Risk. *Review of Economic Studies* 36, 335-346.

[27] Henrion R., Strugarek C. 2008. Convexity of Chance Constraints with Independent Random Variables. *Computational Optimization and Applications*. In Press.

[28] ILOG `CPLEX 10.1` Division. 2006. *ILOG* `CPLEX 10.1` *10.0: Reference Manual*, Incline Village, NV, `http://www.ilog.com`.

[29] Jobst N.J., Horniman M.D., Lucas C.A., Mitra G. 2001. Computational Aspects of Alternative Portfolio Selection Models in the Presence of Discrete Asset Choice Constraints. *Quantitative Finance* 1, 1-13.

[30] Kataoka S. 1963. A Stochastic Programming Model. *Econometrica* 31 (1-2), 181-196.

[31] Kellerer H., Mansini R.W., Speranza M.G. 2000. Selecting Portfolios with Fixed Costs and Minimum Transaction Lots. *Annals of Operations Research* 99, 287-304.

[32] Konno H., Yamamoto R. 1991. Mean-Absolute Deviation Portfolio Optimization Model and its Applications to Tokyo Stock Market. *Management Science* 37, 519-531.

[33] Konno H., Yamamoto R. 2005. Integer Programming Approaches in Mean-Risk Models. *Computational Management Science* 2 (4), 339-351.

[34] Leyffer S. 1999. *User Manual for* `MINLP_BB`, University of Dundee.

[35] Li D., Chan T.F., Ng W.L. 1998. Safety-First Dynamic Portfolio Selection. *Dynamics of Continuous, Discrete and Impulsive Systems* 4, 585-600.

[36] Li D., Sun X. L., Wang J. 2006. Optimal Lot Solution to Cardinality Constrained Mean-Variance Formulation for Portfolio Selection. *Mathematical Finance* 16 (1), 83-101.

[37] Linderoth J.T., Savelsbergh M.V.P. 1999. A Computational Study of Branch and Bound Search Strategies for Mixed Integer Programming. *INFORMS Journal on Computing* 11, 173-187.

[38] Mansini R.W., Ogryczak W., Speranza M.G. 2003. LP Solvable Models for Portfolio Optimization: A Classification and Computational Comparison. *IMA Journal of Management Mathematics* 14, 187-220.

[39] Mansini R.W., Speranza M.G. 1999. Heuristic Algorithms for the Portfolio Selection Problem with Minimum Transaction Lots. *European Journal of Operational Research* 114, 219-233.

[40] Markowitz H.M. 1952. Portfolio Selection. *The Journal of Finance* 7, 77-91.

[41] Marshall A., Olkin I. 1979. *Inequalities: Theory of Majorization and its Applications.* Academic Press, New York.

[42] Mitra G., Ellison F., Scowcroft A. 2007. Quadratic Programming for Portfolio Planning: Insights into Algorithmic and Computation Issues. Part II: Processing of Portfolio Planning Models with Discrete Constraints. *Journal of Asset Management* 8, 249-258.

[43] Morgan Guaranty. 1994. RiskMetrics. *Technical Report*. New York, Morgan Guaranty.

[44] J.P. Morgan. 1997. CreditMetrics^{TM}. *Technical Report*. New York, J.P. Morgan.

[45] Mulvey J., Erkan H.G. 2003. Decentralized Risk Management for Global P/C Insurance Companies. *Working Paper*. In: *Applications of Stochastic Programming*. Eds: Wallace S.W., Ziemba W.T. MPS-SIAM Series on Optimization, Philadelphia, PA, 503-530.

[46] Ogryczak W., Ruszczyński A. 1999. From Stochastic Dominance to Mean-Risk Models: Semi-Deviations as Risk Measures. *European Journal of Operational Research* 116, 33-50.

[47] van de Panne C., Popp W. 1963. Minimum Cost Cattle Feed under Probabilistic Constraints. *Management Science* 9, 405-430.

[48] Prékopa A. 1995. *Stochastic Programming*. Kluwer. Boston, MA.

[49] Prékopa A., Subasi M. 2007. A Convexity Theorem in Programming Under Probabilistic Constraints. *RUTCOR Research Paper* 32-2007.

[50] Rockafellar R.T., Uryasev S. 2000. Optimization of Conditional Value-at-Risk. *Journal of Risk* 2, 21-41.

[51] Roy A.D. 1952. Safety First and The Holding Assets. *Econometrica* 20, 431-449.

[52] Ruszczyński A., Shapiro A. 2003. *Stochastic Programming: Handbook in Operations Research and Management Science 10*. Elsevier.

[53] Sharpe W. 1966. Mutual Fund Performance. *Journal of Business* 39, 119-138.

[54] Scherer B., Martin D. 2005. *Introduction to Modern Portfolio Optimization*. Springer, New York.

[55] Schultz R., Tiedemann S. 2004. Conditional Value-at-Risk in Stochastic Programs with Mixed-Integer Recourse. *Mathematical Programming, Series B* 105, 365-386.

[56] Schultz R., Tiedemann S. 2006. Risk Aversion via Excess Probabilities in Stochastic Programs with Mixed-Integer Recourse. *SIAM Journal on Optimization* 14, 115-138.

[57] Standard&Poor's. 2005. S&P and MSCI Barra Announce Revisions to the Global Industry Classification Standard (GICS). *S&P Technical Report*.

[58] Tütüncü R. H., Koenig M. 2004. Robust Asset Allocation. *Annals of Operations Research* 132, 157-187.

[59] Vielma J.P., Ahmed S., Nemhauser G.L. 2008. A Lifted Linear Programming Branch-and-Bound Algorithm for Mixed Integer Conic Quadratic Programs. *INFORMS Journal of Computing*. In Press.

[60] Wächter A., Biegler L.T. 2006. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming* 106 (1), 25-57.