



Universidad
Carlos III de Madrid



This is a postprint version of the following published document:

García, S., Quintana, D., Galván, I., Isasi, P. (2014). Extended mean-variance model for reliable evolutionary portfolio optimization. *AI Communications*, vol. 27, no. 3, pp. 315-324. Available in <http://dx.doi.org/10.3233/AIC-140600>.

© IOS Press

Extended Mean-Variance Model for Reliable Evolutionary Portfolio Optimization

Sandra García^a, David Quintana^{a,*},
Inés M. Galván^a and Pedro Isasi^b

^a *Department of Computer Science,
Universidad Carlos III de Madrid,
Av. de la Universidad, 30
Leganés, 28910, Spain.*

E-mail: {sgrodrig,dquintan,igalvan}@inf.uc3m.es

^b *E-mail: isasi@ia.uc3m.es*

Real world optimization of financial portfolios pose a challenging multiobjective problem that can be tackled using Evolutionary Algorithms. The fact that the optimization process is subject to the presence of uncertainty concerning asset returns is likely to lead to unreliable solutions. This work suggests extending the classic mean-variance optimization problem with a third explicit robustness objective. This results on sets of portfolios that can be subsequently grouped together according to their reliability. This additional information allows for a better informed decision making regarding asset allocation.

Keywords: Multiobjective Evolutionary Algorithms, Financial Portfolio Optimization, Robustness

1. Introduction

Asset allocation has been one of the core lines on financial research for decades. The problem can be succinctly described as finding the best combinations of financial assets according to specific criteria set by the decision maker. Solving the problem under real-world constraints is complicated and subject to uncertainty. The main contribution of this paper is presenting a new approach compatible with solutions based on evolutionary algorithms that mitigates the latter factor.

At the core of Modern Portfolio Theory lies the seminal work of Markowitz [9,10]. This author suggested a mean-variance optimization framework

that defines the portfolio optimization as multiobjective problem where investors try to both minimize risk and maximize return at the same time. Given the opposing nature of these objectives, the solution does not consist of a single asset allocation, but a range of them that define a Pareto front. In this context, this front is called "Efficient frontier". The portfolios included in the efficient frontier define a curve similar to fig.1, plotted in the *risk-return* space of all possible portfolios. Once this set has been identified, is up to the decision maker to choose the portfolio that suits him better according to his preferences.

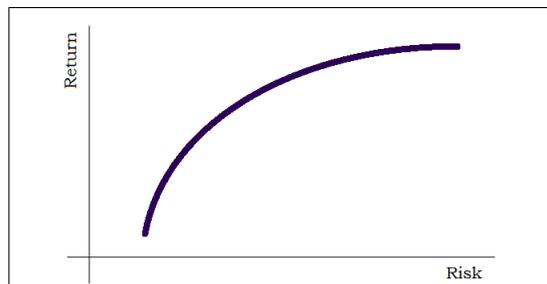


Fig. 1. Efficient frontier

The problem, as presented by Markowitz, relies on a set of simplifying assumptions that make it solvable using quadratic programming (QP). Unfortunately, real world portfolio optimization is hardly compatible with most of them, adding a great deal of complexity to the task. This has led to two lines of work, one coming for operations research relying on mathematical tools like cone programming and a second one based on heuristics. Even though the former leads to fast and elegant solutions, it ends up facing the mentioned problem: at some point, the mathematical complexity of the real world problem is extremely difficult to handle. For this reason, heuristics, such as the ones falling under the umbrella of evolutionary computation, are gaining popularity among researchers and practitioners.

*Corresponding author

The multiobjective nature of the problem has made it attractive to researchers on multiobjective evolutionary algorithms (MOEAs). This has resulted on many contributions either suggesting new solutions to the core problem, or using it as a benchmarking ground for different algorithms. Among these we could mention Skolpadungket et al. [17], who test a set of multi-objective algorithms (VEGA, SPEA2, NSGA-II...); Radziukyniene and Xilinskas [14], where authors compare FastPGA, MOCELL, AbYSS, and NSGA-II on both of the basic framework, and an extended version that considers the dividend yield as a third objective. More recently, Anagnostopoulos and Mamanis [1] compared the performance of different multiobjective algorithms. Deb et al. [4], introduce a customized hybrid version of NSGA-II to tackle the problem.

Even though Modern Portfolio Theory is consolidated, it is not unusual finding portfolio managers who question the reliability of the solutions provided by models. One of the major issues is the sensitivity of the results to key parameters. Regardless of the optimization method chosen to find the efficient frontier, the optimization process requires estimates for the asset returns and the variance-covariance matrix. Given that the forecasts for these parameters are extremely likely to be inaccurate, the selected set of portfolios tend to show risks and returns that differ from the expected ones. In these instances, the deviations can be quite substantial.

Figure 2 illustrates this phenomenon with a real example. It pictures the expected risk and returns for the portfolios in the solution evaluated using the forecasted parameters vs the the real ones (observed ex-post). On the one hand, the set of portfolios that were optimized for the traditionally considered most likely scenario (mean return for each asset over a period of time, and variance-covariance matrix computed using the same data) define the upper efficient frontier. On the other, we represent the risk and return for the same portfolios using the real observed parameters (actual asset returns instead of the mentioned averages) in the lower set. difference is quite apparent.

Since it is clear that this unreliability constitutes a major factor, there is a growing body of literature devoted to this matter. The effort to control the uncertainty associated with the mentioned parameters largely fall in two categories. The first

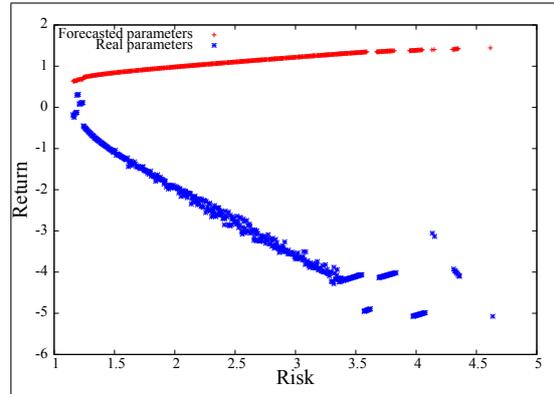


Fig. 2. Solution evaluated with forecasted (red points, on top) and real parameters (blue points, on bottom)

one puts an emphasis on having robust estimates for the parameters. This usually results in efforts to filter the estimates to reduce, for instance, the influence of extreme past events on their computation [12]. Conversely, the second group of authors focus their attention on approaches that handle uncertainty in the parameters during the optimization process [13,18]. This work is related to the latter. We introduce an approach based on MOEAs that mitigates the problem. This is interesting because, as it was mentioned before, these evolutionary algorithms are better suited to operate under real-world constraints.

The standard approaches to find the efficient frontier are based on a single scenario. That means that it involves a single prediction for the asset returns and the variance-covariance matrix. Given that is almost certain that the estimates will not be accurate, the decision maker bears the risk of trusting a solution that may be extremely sensitive to deviations. For this reason we consider that assessing the candidate solutions in different likely scenarios and identifying those that consistently offer a good performance might be a good approach. That is the reason why we suggest extending the classic mean-variance optimization problem with a third explicit robustness objective that considers performance under different potential values of the parameters.

The solution introduced in this paper is related to alternatives based on resampling [15,16]. The most comparable traditional approach is the one described by Idzorek [7]. He suggests the use of a combination of traditional QP with Monte Carlo simulation to derive a set of fronts that are subse-

quently merged into a single solution. Even though the idea is pretty interesting, the approach suffers the same shortcomings as QP, that is, the ability to deal with real-world constraints. A solution based on MOEAs would be less likely to suffer from this limitation.

There is a previous effort based of evolutionary algorithms along the lines of this work. The solution introduced in [6] uses a simple resampling that replaces every generation the parameters used in the fitness functions. The alternative presented in this work has the advantage of providing the decision maker with a wider range of alternatives grouped according to their reliability. This allows for a sensitivity analysis which contributes to better-informed choices.

Given that proposed approach is compatible with different MOEAs, the solution will be tested on three alternatives. The algorithms considered in the experimental section will be SPEA2, NSGA-II and SMPSO. The first two, SPEA2 [19] and NSGA-II [3], are some of the most referenced multiobjective genetic algorithms in the literature. Their suitability in the domain has been confirmed in previous works like [17,1]. The other, SMPSO [11], is a multiobjective evolutionary algorithm based on particle swarm optimization.

As with any other multiobjective problem, the analysis of the solutions will require the use of metrics. However, the metrics that are most used in this field, such as Hypervolume, Spread or Set-Coverage are not appropriate indicators of stability in this context. For this reason we use three specific ones, Estimation Error, Extreme Risk and Unrealized Returns that capture different aspects of stability. These metrics will be described in detail later.

The rest of the paper is organized as follows. First, we make a formal introduction to the financial portfolio optimization problem. Next, we describe the encoding and the fitness function at the core of the process. Then, we explain the mentioned metrics. This will be followed by the experimental framework and the experimental results. Finally, the last section will be devoted to the conclusions.

2. Financial Portfolio Optimization Problem

Financial portfolios can be defined as a collection of investments or assets held by an institution

or a private individual. Modern Portfolio Theory was originated in the article published by Harry M. Markowitz, in 1952 [9]. It explains how to use the diversification to optimize the Portfolio. In general, the portfolio optimization problem is the choice of an optimum set of assets to include in the portfolio and the distribution of investor's wealth among them. Markowitz [10] assumed that solving the problem requires the simultaneous satisfaction of the maximization of the expected portfolio return $E(R_p)$ and the minimization of risk (variance) σ_p^2 , that is, solving a multiobjective optimization problem with two objective functions. The portfolio optimization problem can be formally defined as:

- Minimize the **risk** (variance) of the portfolio:

$$\sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij} \quad (1)$$

- Maximize the **return** of the portfolio:

$$E(R_p) = \sum_{i=1}^n w_i \mu_i \quad (2)$$

- Subject to:

$$\sum_{i=1}^n w_i = 1 \quad (3)$$

$$0 \leq w_i \leq 1; i = 1..n \quad (4)$$

where n is the number of available assets, μ_i the expected return of the asset i , σ_{ij} the covariance between asset i and j , and w_i are the decision variables giving the composition of the portfolio. The constraints referenced in equations 3 and 4 require full investment of funds and prevent the investor from shorting any asset, respectively. In practice, risk is usually characterized in terms of the standard deviation σ_p instead of the variance σ_p^2 . We will adhere to this criterion.

The solution to the problem should also consider some real world constraints [2] such as:

- Cardinality constraint: it is possible to define the maximum C_{max} and minimum C_{min} number of assets in which it is possible to invest ($w_i \neq 0$):

$$C_{min} \leq \Sigma(w_i \neq 0) \leq C_{max} \quad (5)$$

- Values limit constraint: each weight w_i must have a value in the interval $[lim_{inf}, lim_{sup}]$, where:

$$0.0 \leq lim_{inf} \leq w_i \leq lim_{sup} \leq 1.0 \quad (6)$$

All of these equations are solved by a set of points that constitute the efficient frontier of the problem. The points of that curve represent portfolios which have the minimum amount of risk, given a certain expected return (and viceversa).

3. Proposed Robustness-Enhancing Multiobjective Formulation

In this Section, it will be explained how MOEAs has been used to tackle the problem of achieving robust or stable portfolios. In order to specify the problem, both the representation of the solutions or chromosome (portfolios) and the function to be optimized (the fitness function) must be described.

3.1. Encoding

Individuals for evolutionary algorithms will represent portfolios, so that each portfolio is a single element of the population. The solution encoding is given by a vector of real numbers. Each of these numbers represents the percentage of investment per asset (also called weight: w_i where i ranges from $1 \dots n$, and n is the number of investable assets). As it is observed from the definition problem explained before, every individual must meet the constraints specified by eqs. 3, 4. The sum of weights per individual must be 1, that is full investment is required. Also, the individuals must satisfy additional real-world constraints showed in eqs. 5 and 6. The satisfaction of these constraints is guaranteed by a process of repairing solutions. This process basically consists on adjusting the number of invested assets when they do not belong to the interval $[C_{min}, C_{max}]$ to ensure compliance with the cardinality constraint. This is done adding or dropping assets until the requirement is met. In case the sum of weights per individual is not 1.0, the repairing process fine tunes the holdings adding or subtracting random amounts up to the required adjustment. These changes are forced to meet the investment limits $[lim_{inf}, lim_{sup}]$. The individuals are repaired both after initializing the population and applying the genetic operators to guarantee the effectiveness of solutions.

3.2. Adjusted Fitness Function

As it is previously commented, predicted parameters are subject to uncertainty and usually differ from the real ones. Basic approaches usually optimize portfolios for just one scenario, the forecasted, getting solutions that may be hyper-specialized on it. Therefore, when these portfolios are computed using the observed parameters, the predicted efficient frontier lies far from the actual one. The way to avoid this situation is to find those portfolios that are stable in more than one scenario, so that they are robust to big changes in parameters becoming, in consequence, more reliable in practice. With this purpose, a third objective that evaluates the stability of a portfolio has been included in the formulation of the fitness function.

Let us remember that the problem is tackled using multi-objective evolutionary algorithms with three objectives to optimize. The first two ones are given by the formulation explained in section 2, this is: the return $E(R_p)$ (eq. 2) to be maximized and the risk σ_p (eq. 1) to be minimized. Next, the third objective to improve robustness considered in this work is explained in detail.

This objective is denoted as Z_p . Minimizing it at the same time than it maximizes $E(R_p)$ and minimizes σ_p , our approach tries to get a set of portfolios that are not only robust, but also performs well in terms of risk and return while keeping good trade-off among the three objectives. For this purpose, Z_p measures the robustness of one particular individual evaluating it under different resampled parameters. This value is got averaging the Mahalanobis distances among the pair of $E(R_p)/\sigma_p$ objectives of the individual p (denoted as \bar{x}_p) and the expected risk/return in G different and feasible scenarios $(\bar{x}_{p1}, \dots, \bar{x}_{p4})$. Eq. 7 shows how this objective is calculated:

$$Z = \frac{\sum_{i=1}^G [d_M(\bar{x}_p, \bar{x}_{pi})]^2}{G} \quad (7)$$

where G is the number of resampled scenarios and d_M is the Mahalanobis distance [8] defined as follows (with Σ as the variance-covariance matrix):

$$d_M(\bar{x}, \bar{y}) = \sqrt{(\bar{x} - \bar{y})^T \Sigma^{-1} (\bar{x} - \bar{y})} \quad (8)$$

The rationale for the choice of Mahalanobis distance as basis for comparison is the fact that it captures the structural relationship among variables.

Among other advantages, this makes the comparison resistant to scale issues derived from the difference in value range for each objective.

As \bar{x}_p is computed under the expected parameters, $d_M(\bar{x}_p, \bar{x}_{pi})$ gives the deviation of the portfolio when its variables are evaluated in a different scenario. New groups of parameters, G , are created in each generation and they are used to evaluate the stability of every existing individual (the new and the old ones). Using different scenarios reduces significantly the risk of generating one that do not differ enough from the one used by the algorithm. Even more, the distance is raised to the square in order to emphasize more the deviations. Logically, smaller values of distances correspond to more reliable portfolios.

The set of the new feasible scenarios is generated using nonparametric bootstrap, so that the data of the set of parameters used to evaluate the population is resampled creating a new one as it is described in Alg. 1.

Algorithm 1 Resampling method

S is the original sample set with a size N_s .
 S' is the new sample set with a size N'_s . At the beginning, $S' = 0$ and $N'_s = 0$.
while $N'_s \neq N_s$ **do**
 Select randomly $X_i \in S$.
 $S' = S' \cup \{X_i\}$.
end while
Return(S')

4. Evaluation Metrics

In the context studied in this work, the most used metrics, as Hypervolume (HV) and Spread [20], can not be used to measure the quality of solutions from a robustness point of view. These metrics could be useful when we look for good distributed and strong dominant fronts. However, they are not appropriate to capture the effect showed in fig. 2. To measure the robustness of the solutions, we define three metrics that evaluate different aspects related portfolio robustness. They are named: Estimation Error, Unrealized Returns and Extreme Risk.

- **Estimation Error:** It evaluates the average difference between the expected risk and return for every portfolio in the efficient frontier and the actual risk and return a posteriori once

the real values of the parameters are observed. Formally, it can be expressed as:

$$EE = \frac{\sum_{p=1}^N [d_M(\bar{x}_p, \bar{x}_p')]^2}{N} \quad (9)$$

where N is the number of portfolios in the Pareto front; \bar{x}_p represents the pair $(E(R_{np}), \sigma_{np}^2)$ for portfolio p and period t_n calculated using forecasted parameters; \bar{x}_p' is defined by the pair $(E(R_{np})', \sigma_{np}'^2)$ where both return and risk are computed using the the real parameters (the ones observed for t_n) for the same portfolio and moment in time; and, d_M is the Mahalanobis distance defined in eq. 8.

The smaller is the difference between the forecast and reality, the lower is the value of this metric and, therefore, the higher is the reliability of the original front.

- **Unrealized Returns:** This indicator provides information on the average potential return left on the table. Namely, it measures the average squared difference between the realized return and the maximum potential return for that risk level for all the portfolios in the solution. This is defined as:

$$UR = \frac{\sum_{p=1}^N [E(R_{npe}) - E(R_{np})]^2}{N} \quad (10)$$

where N is the number of portfolios in the solution, $E(R_{np})$ is the return of portfolio p computed using the ex-post parameters for time t_n and $E(R_{npe})$ is the return of the portfolio, pe , with the most similar risk level in the efficient frontier derived using the ex-post parameters. High values on this metric would indicate large unrealized potential returns.

- **Extreme Risk:** This metric evaluates the sensitivity of the solutions to worst-case scenarios. It is closely related to the third objective. For computation purposes, it uses the same basic definition. There are, however, two differences. The first one is that instead of relying a relatively small set, G , of resampled scenarios, this metric is derived from a considerably larger one, Q . The second is that, instead of considering the average for all Q scenarios, it only takes into account a small subset. Specifically, the computation would include only the W worst-case ones. For this purposes we define worse-case scenarios as the parameter combinations that result in the highest av-

erage Mahalanobis distance between the expected risk and returns, and the risk and return for the same portfolios using the resampled parameters. This metric is defined as:

$$ER = \frac{\sum_{w=1}^W \frac{\sum_{p=1}^N [d_M(\bar{x}_p, \bar{x}_{pw})]^2}{N}}{W} \quad (11)$$

where N is the number of portfolios in the solution, $d_M(\bar{x}_p, \bar{x}_{pw})$ is the Mahalanobis distance between the pair $(E(R_{np}), \sigma_{np}^2)$ and $(E(R_{npw}), \sigma_{npw}^2)$; this last one represents the return/risk for the w scenario, p portfolio and t_n period of time.

The rationale for this indicator is providing an estimate for the expected average negative outcome of the realization of the worst scenario with probability W/Q . The higher the metric, the higher the risk.

5. Experimental Framework

5.1. Data Sets

Experimentation has been performed using sample that consists of 240 monthly returns for eight broad financial indexes representing that many asset classes. The series of monthly returns cover the time period from January, 1990 to December, 2009 and the source for the data is *Datastream*. The list of indexes is provided in Table 1.

Table 1
Data Sets

Name	Code
Frank Russell 1000 Growth	FRUS1GR
Frank Russell 1000 Value	FRUS1VA
Frank Russell 2000 Growth	FRUS2GR
Frank Russell 2000 Value	FRUS2VA
S&P GSCI Commodity Total Return	GSCITOT
MSCI EAFE	MSEAFEL
BOFA ML CORP MSTR (\$)	MLCORPM
BOFA ML US TRSY /AGCY MSTRAAA(\$)	MLUSALM

For experimental purposes, in order to test our approach in many real and historical situations, a sliding window has been implemented. The aim of this is generating a large set of single-period portfolio optimization problem instances based on real data, not a single testing ground for a dynamic multi-period approach. The size of the window is

set to $n = 120$, return periods that correspond to 10 years of data. So that, the algorithm will rely on data from t_1 to t_n to identify the best possible allocations for the period t_{n+1} . Each time, the 10-year window will move one month, 120 times in total to cover the date interval from 31/01/1990 to 31/12/2009.

5.2. Evolutionary Multi-objective Algorithms and their Parametrization

Motivated by our interest in the multi-objective approach, we will test the effectiveness of the proposed approach on different MOEAs using the same chromosome structure and fitness evaluation procedure described in sections 3.1 and 3.2.

We have selected three different algorithms: NSGA-II, proposed by Deb et al. [3], is one of the most widely used multiobjective metaheuristics and the reference algorithm, SMPSO [11] that is based on particle swarm optimization, and SPEA2 algorithm [19], developed by Zitzler et al., has been also selected because performs better than other MOEAs in the field of Portfolio Optimization [17,1]. A summary of their parameter settings is included in Table 2.

Table 2

Parameters. $L = 8$ (individual length). The termination condition is to compute 300 iterations.

SPEA2	
<i>Population size</i>	200 individuals
<i>Archive size</i>	200 individuals
<i>Crossover</i>	SBX, $p_c = 0.9$
<i>Mutation</i>	Polynomial, $p_m = 1/L$
<i>Selection of Parents</i>	Binary tournament
NSGA-II	
<i>Population size</i>	200 individuals
<i>Crossover</i>	SBX, $p_c = 0.9$
<i>Mutation</i>	Polynomial, $p_m = 1/L$
<i>Selection of Parents</i>	Binary tournament
SMPSO	
<i>Archive size</i>	200 particles
<i>Swarm size</i>	200 particles
<i>Mutation</i>	Polynomial, $p_m = 1/L$

These algorithms are implemented in jMetal [5], a Java framework aimed at multiobjective optimization with metaheuristics. By reusing the base classes of jMetal, all the techniques share the same basic core components (solution encodings, oper-

ators, etc.). This ensures a fair comparison of the considered algorithms.

One study varying the value of G in fitness function (described in sec. 3.2) has been done. We consider that choosing four scenarios ($G = 4$) to evaluate the individual is enough to represent the variability in parameters. $G > 4$ would increment the computing time of the algorithm without highly increasing the heterogeneity.

Moreover, every algorithm is carried out under the same cardinality $[C_{min}, C_{max}] = [2, 6]$ and value-limits per asset $[lim_{inf}, lim_{sup}] = [0.1, 0.8]$ constraints described in section 2 (eqs. 5 and 6).

5.3. Front Processing

The set of fronts resulting from the experimentation contains non-dominated individuals according to the three objectives driving the evolution process. Figure 3 shows one solution picked at random from the set of experiments based on the SPEA2 algorithm for illustration purposes.

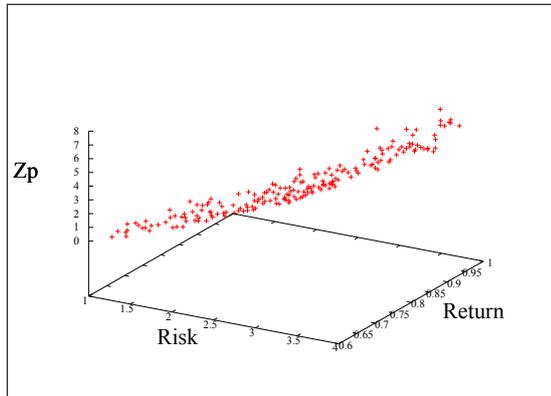


Fig. 3. SPEA2. Time window from 31/05/1996 to 28/04/2006.

The three-objective solutions are not readily comparable with the ones resulting from the standard two-objective formulation. For this reason, these sets of portfolios will be split in subsets according to the robustness objective. The identification of the non-dominated solutions in terms of risk and return for each subset results in fronts whose quality can be evaluated using the stability metrics defined in section 4.

With this purpose, at the end of the execution, the final solution set is ranked by Zp objective and split in three parts. Zp with lower values means

higher stability of the portfolio (see sec. 2) so that, the first third of individuals with smaller Zp will be classified as “high stability”, followed by the second third as “medium stability” and the last one as “low stability”. Then, Zp values are discarded from solutions keeping only the non-dominated individuals in terms of risk and return objectives. Figs. 4, 5 and 6 show the three fronts derived from the processing of the front represented in fig. 3.

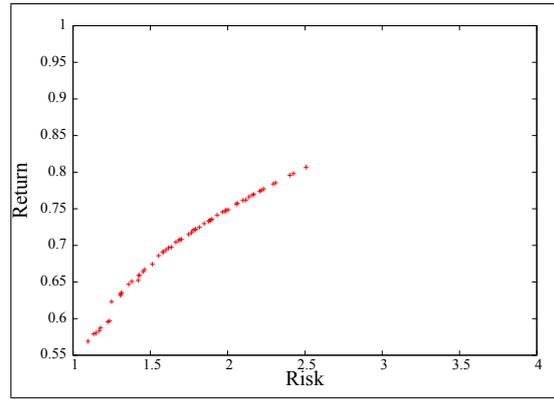


Fig. 4. High stability front. SPEA2. Time window from 31/05/1996 to 28/04/2006.

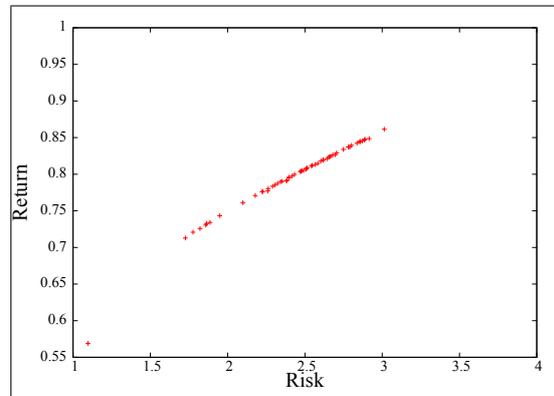


Fig. 5. Medium stability front. SPEA2. Time window from 31/05/1996 to 28/04/2006.

6. Experimental Results

In this section we present the experimental results based on the data and procedures already introduced. The approach is tested on SPEA2, NSGA-II and SMPSO and the high, medium and low stability fronts have been evaluated using the

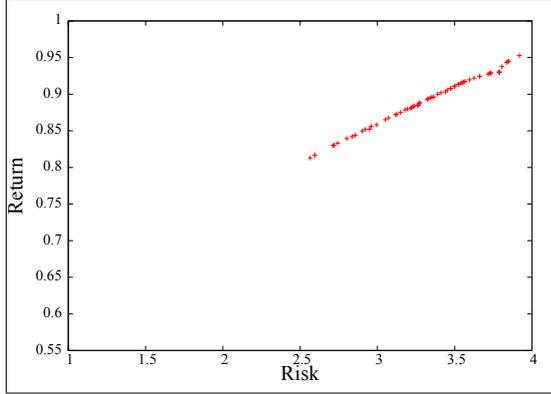


Fig. 6. Low stability front. SPEA2. Time window from 31/05/1996 to 28/04/2006.

metrics defined in section 4. These results are benchmarked against the results obtained by the standard versions of the same algorithms, using only the two objectives that define the basic portfolio optimization problem (risk minimization and return maximization).

The experimental settings are the same, regardless of the approach (robust vs. standard), and all the algorithms use the same encoding described in section 3.1. Given the stochastic nature of MOEAs, 20 runs have been carried out for each approach and each algorithm. The results are therefore presented in terms of standard descriptive statistics. The differences among the different results have been formally tested for statistical significance using the Wilcoxon test. The choice of this test is justified by the lack of normality in the distribution of the metrics.

Tables 3, 4 and, 5 show the values of Estimation Error, Unrealized Return and Extreme Risk metrics, respectively. Tables include the average, medium and variance for both standard and robust approaches for the tested MOEAs (SPEA2, NSGAII and, SMPSO). Tables also report the percentage of improvement of the robust version over the standard approach (see column labeled as **Av. imp.**), which it is computed as:

$$Av. imp. = 1 - \frac{AverageStandardAlg.}{AverageRobustAlg.}$$

The estimation error shows the expected pattern of results. The higher the stability of the solution, the lower the metric. Regardless of the algorithm chosen, the average improvement is considerable. The highest average gains come from the genetic algorithms, but this is due to the high baseline error. SMPSO seems to offer a better performance as

Table 3
Estimation Error

EE	Average	Median	Variance	Av. imp.
SPEA2	2.5198	1.7945	5.1421	
SPEA2 High S	0.9387	0.6448	0.9844	62.75 %
SPEA2 Medium S	2.0338	1.5425	3.6269	19.29 %
SPEA2 Low S	4.1945	3.2386	16.4057	-66.46 %
NSGAII	2.2823	1.7996	3.9817	
NSGAII High S	0.8557	0.5442	1.2544	62.51 %
NSGAII Medium S	1.9449	1.4377	3.4238	14.78 %
NSGAII Low S	4.4823	3.5236	18.2888	-96.40%
SMPSO	1.5204	1.2878	1.7354	
SMPSO High S	0.7660	0.5744	0.5298	49.62 %
SMPSO Medium S	1.2768	0.9673	1.3110	16.03 %
SMPSO Low S	2.4097	2.1001	4.2289	-58.48%

it provides the best average result, the second best median error (close to the lowest one, offered by NSGA II) and the most consistent results as shown in the variance. In this case, the average reduction in the mean value of the metric across all the experiments is 49.62%. This means that the discrepancy between the expected behavior of the selected portfolios and reality is significantly lower for the ones evolved to be stable than for the alternatives identified using the standard mean-variance optimization framework. The estimation error on the bottom third subset of portfolios is, as expected, substantially high. The result is particularly bad for NSGA II, almost doubling the baseline.

Table 4
Unrealized Returns

UR	Average	Median	Variance	Av. imp.
SPEA2	3.5848	2.7989	8.5604	
SPEA2 High S	2.3903	1.9530	2.8606	33.32 %
SPEA2 Medium S	3.3552	2.6677	7.6028	6.40 %
SPEA2 Low S	4.3958	3.3277	14.7493	-22.62 %
NSGAII	3.4871	2.7326	7.8823	
NSGAII High S	2.2413	1.8409	2.2544	35.72 %
NSGAII Medium S	3.3600	2.6389	7.6546	3.64 %
NSGAII Low S	4.5813	3.4372	16.4506	-31.38%
SMPSO	2.9805	2.4745	4.9961	
SMPSO High S	1.9837	1.6159	1.9944	33.44 %
SMPSO Medium S	2.6855	2.2980	4.1098	9.90 %
SMPSO Low S	3.7202	3.0062	8.1424	-24.82%

Table 4 shows a similar set of results regarding the second metric. The unrealized returns drop significantly once we consider the stable subset of solutions. The difference is much when we consider the medium stability solutions. Once again,

SMPSO provides the best performance. Even though the basic versions of SPEA2 and NSGA-II tend to offer significantly larger unrealized returns than SMPSO, the improvement gained from the third objective is similar in relative terms. The gains from selecting the most stable solutions range from 33.32% and 35.72%. The choice of the medium stability front reduces the mentioned gain from 33.40% to 9.9% in the best case, SMPSO, and from 35.70% to 3.64% for NSGA-II.

Table 5
Extreme Risk

ER	Average	Median	Variance	Av. imp.
SPEA2	3.5308	3.0614	4.9670	
SPEA2 High S	1.5177	1.2593	1.1936	57.02 %
SPEA2 Medium S	2.9764	2.5328	3.6940	15.70 %
SPEA2 Low S	5.6684	4.6919	15.8470	-60.54 %
NSGAI	3.2407	2.8451	3.9687	
NSGAI High S	1.3633	1.0087	1.4098	57.93%
NSGAI Medium S	2.8657	2.4719	3.5085	11.57%
NSGAI Low S	6.0523	5.0495	17.8200	-86.76%
SMPSO	2.2485	2.0297	1.8884	
SMPSO High S	1.2384	1.0332	0.6596	44.92 %
SMPSO Medium S	1.9356	1.7212	1.5613	13.92 %
SMPSO Low S	3.4179	3.0564	4.9731	-52.01%

When we consider extreme scenarios only, we obtain consistent results. The selection of the 5 worst case scenarios out of a set of 100 per time period leads to a generalized increase in the prediction errors. The distribution of these errors matches the ones reported for *EE* metric. Table 4 shows how the additional robustness objective reduces the average metric up to 57.93%. The PSO-based solution provides both the best average results and the highest consistency. Is worth noting that, for the stable case, the median error for NSGA-II is slightly better than the one for SMPSO. However, this comes at a cost of a variance that is more than twice the size. This phenomenon does not hold when we consider the medium stability solution. Here, the superiority of SMPSO over the multiobjective genetic algorithms is apparent regardless of the considered statistic. The results for the low stability section of the three-objective front are consistently bad. Having said that, there is an unexpected result. The average error in the worst-case scenarios across the whole period of study for SMPSO Low Stability is lower than baseline for SPEA2 in both the average and median cases.

According to the results presented in tables 3, 4 and 5, the inclusion of the third objective to guide the evolution process enhances the robustness of the solutions regardless of the algorithm considered. The highest improvement over the standard MOEAs takes place on the high stability third of the solutions. There are also some improvements on solutions for the medium stability range and, as we could anticipate, the results are extremely poor on the bottom third. All the reported differences observed in the metrics within the three core algorithms are statistically significant at the 1% conventional level. The only exception to this is the comparison of *UR* between the medium stability front computed using NSGA-II and its baseline, which is significant at 5%.

Regarding the behavior of the evolutionary algorithms, for the high stability portfolios, SPEA2 and NSGAI perform worse than SMPSO in terms of estimation error and unrealized returns. The differences tend to be milder for the extreme scenarios. However, the gap on this metric gets wider once we lower the requirements for the value of the robustness in the third objective.

7. Summary and Conclusions

Portfolio optimization has been for decades one of the core research areas in Finance. This problem is usually framed as a multiobjective problem aimed at the identification of the asset allocations that result on the simultaneous maximization of return and minimization of risk. The identification of the efficient frontier is a challenging optimization task, specially once real world constraints are taken into consideration. This is the reason why multiobjective evolutionary algorithms are increasingly getting traction on this area.

In this work, we suggested extending the mean-variance standard model with a third stability objective. The aim of this change is mitigating the exposure of the decision maker created by the uncertainty on the real value of the parameters. Given that both the forecast for the expected returns of the investable assets and the variance-covariance matrix is likely to be inaccurate, we suggest a solution that prevents overspecialization. The solution is the definition of a stability objective that measures the sensitivity of the portfolios that define the population to unexpected val-

ues for the parameters. This approach results on a three-dimensional efficient frontier that provides the decision maker relevant information regarding the reliability of the forecasted risk and return for each portfolio.

The effectiveness of this approach was tested comparing the estimates for the efficient frontiers in terms of the robustness metrics: estimation error, unrealized return and extreme risk. In order to be able to compare the three objective efficient frontier and the two objective standard version, the set of portfolios that form latter has been split into three subsets of equal size according to the stability. The selection of the dominant solutions in terms of risk and return for each of them, resulted in three efficient frontier that were readily comparable with the standard approach.

The experimental work covered twenty years worth of data for eight broad financial indexes and three multiobjective evolutionary algorithms. The results show that the suggested third stability objective is closely related to robustness. Regardless of the basic algorithm chosen, the third objective allows the identification of portfolios that are substantially more reliable than the ones provided by the standard methods. The results show consistency; high stability fronts are more reliable than medium stability fronts which, in turn, are more reliable than the low stability frontiers. This is true across metrics and algorithms. The comparison of these three categories with the baseline standard algorithms also presents a regular pattern: both the high stability and the medium stability frontiers beat the basic solution, whose results outperform the low stability frontier.

Even though the results structure is independent of the basic algorithms, the choice of the MOEA does affect the solution. Out of the three alternatives considered, NSGA-II, SPEA2 and SMPSO, the third one provides the most stable results. The other two, both genetic algorithms, offer a similar performance.

Acknowledgements

The authors acknowledge financial support granted by the Spanish Ministry of Science under contract TIN2011-28336 (MOVES)

References

- [1] K.P. Anagnostopoulos and G. Mamanis. The mean-variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms. *Expert Systems with Applications*, In Press, Corrected Proof, 2011.
- [2] H.J.C. Barbosa and A.C.C. Lemonge. A genetic algorithm encoding for a class of cardinality constraints. In *Proceedings of the 2005 conference on Genetic and evolutionary computation*, GECCO '05, New York, USA, 2005. ACM, pp. 1193-1200
- [3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, Vol. 6(2), 2002, pp. 182-197
- [4] K. Deb, R.E. Steuer, R. Tewari, and R. Tewari. Bi-objective portfolio optimization using a customized hybrid NSGA-II procedure. In *EMO*, 2011, pp. 358-373.
- [5] J.J. Durillo, A.J. Nebro, and E. Alba. The jmetal framework for multi-objective optimization: Design and architecture. In *CEC 2010*, volume 5467 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, Barcelona, Spain, July 2010, pp. 4138-4325.
- [6] S. García, D. Quintana, I.M. Galván, and P. Isasi. Portfolio Optimization using SPEA2 with Resampling. In *The 12th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, 2011, pp. 127-134.
- [7] T.M. Idzorek. Developing robust asset allocations. *Working Paper*, Ibbotson Associates, Chicago, 2006.
- [8] P. C. Mahalanobis. On the generalized distance in statistics. *Proc. Nat. Inst. Sci.*, India, 2, 1936, pp. 49-55.
- [9] H.M. Markowitz. Portfolio selection. *The Journal of Finance*, Vol. 7(1), 1952, pp.77-91.
- [10] H.M. Markowitz *Portfolio Selection: efficient diversification of investments*. John Wiley & Son, 1959.
- [11] A.J. Nebro, J.J. Durillo, J. García-Nieto, C.A. Coello Coello, F. Luna, and E. Alba. Smpso: A new pso-based metaheuristic for multi-objective optimization. In *2009 IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*, IEEE Press, 2009, pp. 66-73.
- [12] C. Perret-Gentil and M.P. Victoria-Feser. Robust mean-variance portfolio selection. Fame research paper series, International Center for Financial Asset Management and Engineering, April 2005.
- [13] G. Pflug and D. Wozabal. Ambiguity in portfolio selection. *Quantitative Finance*, Vol. 7(4), pp. 435-442, 2007.
- [14] I. Radziukyniene and A. Xilinskas. Evolutionary methods for multi-objective portfolio optimization. In S. I. Ao, Len Gelman, David WL Hukins, Andrew Hunter, and A. M. Korsunsky, editors, *Proceedings of the World Congress on Engineering 2008 Vol II, WCE '08, July 2 - 4, 2008, London, U.K.*, Lecture Notes in Engineering and Computer Science, International Association of Engineers, Newswood Limited, 2008, pp. 1155-1159.

- [15] D. Ruppert. Statistics and finance: An introduction. *Journal of the American Statistical Association*, 101:849-850, June 2006.
- [16] H. Shiraishi. Resampling procedure to construct value at risk efficient portfolios for stationary returns of assets. ., 2008.
- [17] P. Skolpadungket, K. Dahal, and N. Harnpornchai. Portfolio optimization using multi-objective genetic algorithms. In *Proceeding of 2007 IEEE Congress on Evolutionary Computation*, pp. 516-523, 2007.
- [18] R.H. Tütüncü and M. Koenig. Robust asset allocation. *Annals OR*, 132(1-4):157-187, 2004.
- [19] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 2001.
- [20] E. Zitzler and L. Thiele. An evolutionary algorithm for multiobjective optimization: The strength pareto approach. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, 1998.des