## On the Computation of the Efficient Frontier in Advanced Portfolio Optimization

Arturo Annunziata<sup>1</sup>, Matteo Lapucci<sup>1</sup>, Pierluigi Mansueto<sup>1</sup>, Davide Pucci<sup>1\*</sup>

<sup>1</sup>Global Optimization Laboratory – Department of Information Engineering (DINFO), University of Florence, Via di Santa Marta, 3, Florence, 50139, Italy.

\*Corresponding author(s). E-mail(s): davide.pucci@unifi.it; Contributing authors: arturo.annunziata@unifi.it; matteo.lapucci@unifi.it; pierluigi.mansueto@unifi.it;

#### Abstract

In this work, we deal with the problem of computing a comprehensive front of efficient solutions in multi-objective portfolio optimization problems in presence of sparsity constraints. We start the discussion pointing out some weaknesses of the classical scalarization approach when applied to the considered class of problems. We are then motivated to propose a suitable algorithmic framework that is designed to overcome these limitations: the novel algorithm combines a gradient-based exploration-refinement strategy with a tailored initialization scheme based on memetic or multi-start descent procedures. Thorough computational experiments highlight how the proposed method is far superior to both scalarization and popular genetic algorithms.

**Keywords:** Portfolio optimization; multi-objective problems; sparsity constraints; scalarization; Pareto front reconstruction.

 $\mathbf{MSC\ Classification:}\ 90C29\ ,\ 90C26\ ,\ 90B50\ ,\ 91G10$ 

#### 1 Introduction

Portfolio selection has undoubtedly constituted one of the most attractive and studied problems in decision science, given its relevance to the finance community. Starting

with the pioneering work of Markowitz [1], the task of suitably choosing how to allocate resources among a set of assets in the market has been formalized as an optimization problem, with a growing complexity as details and real-world specifications got taken into consideration.

From the one hand, the convex quadratic optimization problem with standard simplex constraints associated with the baseline Markowitz model has been further addressed taking into account more complex objectives, such as the Sharpe ratio [2], the ESG criterion [3], or portfolio skewness [4]; while some of these objectives are fairly simple to manage – for instance, ESG is linear – other objectives (skewness, Sharpe) even introduce nonconvexity elements within the problem, making it much harder to deal with.

Then, we might note that the mean-variance Markowitz model [5] is already an intrinsically bi-objective optimization model balancing two conflicting objectives: maximizing the expected return and minimizing the variance of the portfolio. The trade-off is commonly handled using a scalarized formulation, where a positive coefficient reflects the investor's risk aversion. Different objectives in most cases then come as additions to portfolio mean and variance, rather than as replacements. The growing importance of sustainable investments motivates the inclusion of ESG criteria in portfolio optimization, to quantify the environmental, social, and governance aspects of investments [3, 6]. Skewness allows to distinguish between upside potential and downside risk; furthermore, the inclusion of a higher-order moment partly overcomes the unlikeliness of the utility functions depending only on the first two moments of portfolio return's distribution [4, 7]. While a weighted sum of the objectives is a practical way to deal with all requirements at once (see, e.g., [8, 9]), many researchers have investigated how to explicitly address the multi-objective formulation [10–18]. Indeed, retrieving a wide set of efficient solutions, i.e., optimal according to Pareto's theory, delegates the final choice of balance between the objectives to the final investor, who is provided with a clear picture of all possible trade-off scenarios to choose from.

On the other hand, new constraints allowed to model the task with increased realism: the addition of general linear constraints, for example, allows to take into account no-short-selling constraints, maximum or minimum investment bounds, exposure limits to specific sectors or industries, turnover constraints, and constraints modeling benchmark considerations; for further information, we refer the reader to [19]. A much harder challenge is then posed by the introduction of a sparsity constraint [20–22] to limit the number of active assets in the portfolio. This constraint is motivated mainly because managing and monitoring a large number of assets is costly, and sparse portfolios are often perceived by investors as reflecting active management strategies [20]. Furthermore, there is a fixed cost associated with each transaction, which further incentivizes sparse portfolios. Sparsity constraints involve thresholds on the zero pseudo-norm, making the feasible set heavily nonconvex and introducing a strong combinatorial element within the problem [22].

A complete, modern formulation of the portfolio selection problem thus consists of a sparse multi-objective optimization (MOO) problem of the form:

$$\min_{x \in \mathbb{R}^n} F(x) = [f_1(x), \dots, f_m(x)]^\top$$
s.t.  $Ax \le b$ ,  $\mathbf{1}_n^\top x = 1$  (1)
$$\|x\|_0 \le s, \quad x \ge \mathbf{0}_n,$$

where n is the number of assets,  $f_1, \ldots, f_m$  are continuously differentiable objective functions,  $A \in \mathbb{R}^{p \times n}$ ,  $b \in \mathbb{R}^p$ ,  $\|\cdot\|_0$  denotes the  $\ell_0$  pseudo-norm, i.e., the number of non-zero components of a vector,  $s \in \mathbb{N}$  such that  $1 \leq s < n$  and  $\mathbf{1}_n$  and  $\mathbf{0}_n$  indicate the n-dimensional vectors of all ones and all zeros, respectively. The above formulation has been the subject of some studies [20, 23], but no thorough framework has been proposed on the methodological side to properly tackle it.

In this paper, we deal with this challenge. In the first place, we discuss the weakness of the simple scalarization strategy. Specifically, we point out that the presence of the cardinality constraint breaks, even in the mean-variance case, the property of the problem only having supported solutions: some efficient trade-offs cannot be obtained as the optimal solution of the scalarized problem for any choice of the weights. In fact, entire portions of the Pareto front, corresponding to "good" choices of active assets, are somewhat shadowed in this setting. Moreover, the obtainment of a well spanned and uniformly spread front of solutions is further precluded by the complexity of choosing suitable wights. Inspired by the work in [24], we then propose a detailed algorithmic framework specifically designed to effectively retrieve an accurate approximation of the Pareto front in instances of problem (1).

The proposed method exploits a heuristic evolutionary strategy [11, 25], equipped with a tailored derivative-based "hard-thresholding" procedure [23, 24], to feed with good starting points a second-stage multi-objective optimization method, based on the Front descent algorithm [13], for filling the Pareto front. All the cogs of the proposed methodology have been polished to be compatible with the presence of general linear constraints. By a thorough computational experimentation, we show that the proposed approach consistently solves the problem with high effectiveness, with superior performance w.r.t. state-of-the-art approaches.

The rest of the paper is organized as follows. In Section 2, we review objective functions and constraints typical of portfolio problems; we also recall basic notions in multi-objective optimization, with some focus on scalarization approaches. In Section 3, we show through a simple example the limitations of the latter class approaches in finding specific efficient solutions. Then, in Section 4 we propose an algorithmic framework designed to effectively reconstruct the Pareto front of MOO sparse portfolio problems. In Section 5, we show the results of thorough computational experiments where we compared our framework with state-of-the-art approaches. Finally, in Section 6 we provide some concluding remarks.

#### 2 Preliminaries

In this preliminary section, we provide a brief description of common objective functions and constraints in portfolio problems. Then, we continue reviewing some basic notions for (sparse) MOO problems.

# 2.1 Objective Functions and Constraints in Portfolio Optimization

In the core discussion of this paper, we are going to consider multi-objective formulations of the (sparse) portfolio selection problem, considering several combinations of objectives. Specifically, we will be interested in:

Expected Return The expected value of the return of investment is defined as

$$f^{\text{ER}}(x) = c^{\top} x,\tag{2}$$

with  $c \in \mathbb{R}^n$  being the expected returns vector [5]; maximizing this quantity is arguably the easiest and most straightforward goal in portfolio optimization.

Portfolio Risk The well-known variance function, defined as

$$f^{\mathbf{V}}(x) = \frac{1}{2}x^{\top}Qx,\tag{3}$$

where  $Q \in \mathbb{R}^{n \times n}$  denotes the positive semi-definite variance-covariance matrix, is classically used to measure the risk associated with a portfolio. Roughly speaking, the higher is the variance of the return of investment, the higher will be the chances of the realized return to be far lower than the expected value.

**ESG Score** The *Environmental*, *Social*, and *Governance* (ESG) score [3, 6] has become an essential criterion in portfolio optimization, reflecting an investor's preference for sustainable and responsible investments.

For each asset  $i \in \{1, ..., n\}$  in the investment universe, let us denote as  $v_i$  its ESG score, which quantifies its performance on environmental, social, and governance dimensions. Let us denote the vector of ESG scores as  $v = [v_1, ..., v_n]^{\top} \in \mathbb{R}^n$ . The portfolio's ESG score is then calculated as the weighted sum of the ESG scores of its constituent assets:

$$f^{\mathrm{ESG}}(x) = x^{\top}v.$$

Maximizing the ESG score, portfolio managers aim to balance traditional financial objectives (e.g., risk and return) with non-financial considerations, such as sustainability and ethical alignment so as to ensure that the portfolio aligns with the investor's sustainability goals.

Sharpe Ratio The Sharpe ratio [2] is a widely used metric that incorporates the expected return and variance functions to evaluate the risk-adjusted performance

of a portfolio. Formally, it is defined as

$$f^{\rm SR}(x) = \frac{f^{\rm ER}(x)}{\sqrt{f^{\rm V}(x)}},$$

i.e., it is the ratio between the expected value and the standard deviation of the return of investment.

This quantity aims to measure how much return a portfolio generates per unit of risk, with higher values indicating better risk-adjusted performance. In brief, the Sharpe ratio helps investors compare portfolios with different risk profiles, penalizing the ones with high volatility unless accompanied by proportionally higher returns.

Portfolio Skewness Skewness represents a measure for the asymmetry of the return distribution [4, 7]: a positive value indicates a higher probability of large positive returns, while negative skewness reflects a higher likelihood of extreme losses. Investors often prefer portfolios with positive skewness, as they provide enhanced upside potential while mitigating downside risks. Thus, by considering skewness as objective function, portfolio managers can address tail risks and align portfolios with investor preferences for asymmetrical return profiles.

Mathematically, following the idea in [4, 26], the skewness of a portfolio can be formalized as the third moment of portfolio return, i.e.,

$$f^{\text{SW}}(x) = \mathbb{E}[(\mathbf{R}_p - f^{\text{ER}}(x))^3], \tag{4}$$

where  $R_p$  is the random variable of portfolio return and  $f^{ER}(x)$  is the expected return of the portfolio.

In order to calculate equation (4), we need to define the coskewness tensor  $\mathbf{C}$ , which captures third-order interactions among asset returns. In particular,  $\mathbf{C}$  is a tensor of size  $n \times n \times n$  whose  $\mathbf{C}_{ijk}$  is given by

$$\mathbf{C}_{ijk} = \mathbb{E}[(r_i - c_i)(r_j - c_j)(r_k - c_k)],$$

where  $r_i$  denotes the return of asset i and  $c_i$  is its mean. The expectation is computed across the time series. The value of  $\mathbf{C}_{ijk}$  quantifies the asymmetric interaction between the deviations of assets i, j, and k from their respective means. The portfolio skewness is then computed by combining the coskewness tensor  $\mathbf{C}$  with the vector of portfolio weights x:

$$f^{\text{SW}}(x) = x^{\top} \mathbf{C}(x \otimes x),$$

where  $\otimes$  indicates the Kronecker product.

As for the constraints, the general portfolio model we address in this work possibly encompasses the following specifications [19]:

**Standard Simplex Constraints** The standard simplex constraint is a basic requirement in portfolio optimization. Mathematically, this constraint is defined as:

$$x \ge \mathbf{0}_n, \qquad \mathbf{1}_n^\top x = 1.$$

The first condition prevents short-selling, meaning that the budget  $x_i$  allocated to each asset  $i \in \{1, ..., n\}$  shall be non-negative.

The second condition ensures that an investor allocates all the available capital to the given set of assets, without leaving any budget portion uninvested.

Minimum and Maximum Investment Constraints Minimum investment constraints ensure that a certain minimum portion of the total capital is allocated to specific assets, while maximum investment constraints prevent concentration in any single asset. These constraints can be expressed as:

$$\ell_i \le x_i \le u_i, \quad \forall i \in \{1, \dots, n\},$$

where  $\ell_i \in \mathbb{R}$  and  $u_i \in \mathbb{R}$  such that  $l_i \leq u_i$  are the minimum and the maximum investment bounds for asset i. In a compact representation, this set of constraints becomes

$$\ell \leq x \leq u$$
,

where  $\ell, u \in \mathbb{R}^n$  are the vectors of minimum and maximum investment bounds respectively.

**Portfolio Beta Constraints** The beta coefficient measures the sensitivity of an asset return to movements in a market index. Formally, for an individual asset  $i \in \{1, ..., n\}$ , the beta is defined as

$$\beta_i = \frac{\operatorname{Cov}(r_i, r_m)}{\operatorname{Var}(r_m)},$$

where  $r_i$  and  $r_m$  are random variables, respectively the return of asset i and the return of the market index,  $Cov(r_i, r_m)$  is the covariance between  $r_i$  and  $r_m$ ,  $Var(r_m)$  is the variance of the market index return. Let us define the vector  $\beta = [\beta_1, \ldots, \beta_n]^{\top} \in \mathbb{R}^n$ .

In practical applications, investors may seek to limit the portfolio beta within a predefined range to manage exposure to market risk; thus, we may want to impose lower and upper bound constraints on the weighted sum of the betas:

$$x^{\top} \beta \ge \beta_{\min}$$
  
 $x^{\top} \beta \le \beta_{\max}$ 

with  $\beta_{\min}$ ,  $\beta_{\max} \in \mathbb{R}_+$ ,  $\beta_{\min} \leq \beta_{\max}$ . In order to embed these linear constraints in formulation (1), we just need to set

$$A = \begin{bmatrix} -\beta \\ \beta \end{bmatrix} \qquad b = \begin{bmatrix} -\beta_{\min} \\ \beta_{\max} \end{bmatrix}.$$

Sector or Industry Exposure Constraints Another common requirement in portfolio management is to limit the exposure to specific sectors or industries. Given a set  $\mathcal{I}$  representing the set of assets belonging to a given industry or sector, the following linear constraint can be imposed

$$\eta_{\mathcal{I}} \le \sum_{j \in \mathcal{I}} x_j \le \mu_{\mathcal{I}}$$

where  $\eta_{\mathcal{I}}, \mu_{\mathcal{I}} \in \mathbb{R}$ , with  $\eta_{\mathcal{I}} \leq \mu_{\mathcal{I}}$ , are the minimum and maximum exposure respectively to the considered industry or sector.

**Turnover Constraint** Given an initial allocation vector  $x^0 \in \mathbb{R}^n$ , the turnover constraint limits the changes in portfolio composition in the following period, reducing transaction costs. This kind of constraint is defined as

$$\sum_{j=1}^{n} |x_j^0 - x_j| \le \tau,$$

where  $\tau$  is the maximum total turnover allowed. The above nonlinear formulation can of course be rewritten equivalently as a set of linear constraints. In particular, introducing a new vector of auxiliary variables  $y \in \mathbb{R}^n$ , we can impose

$$-y_i \le x_i - x_i^0 \le y_i \quad \forall i = 1, \dots, n,$$
 and  $\sum_{i=1}^n y_i \le \tau.$ 

Cardinality Constraint The cardinality constraint is a requirement introduced to limit the number of assets actively included in the portfolio. This constraint is motivated by practical considerations, such as the cost and complexity of managing a large number of assets, as well as the investor's preference for a more concentrated and actively managed portfolio. Sparse portfolios are also more efficient due to reduced transaction costs. Formally, the constraint is imposed setting an upper bound s < n to the  $\ell_0$ -pseudo-norm of the portfolio allocation vector x, i.e.,

$$||x||_0 \le s$$
.

#### 2.2 Fundamentals of Multi-objective Optimization

Let us consider a multi-objective optimization problem of the form

$$\min_{x \in \mathbb{R}^n} F(x) = [f_1(x), \dots, f_m(x)]^\top$$
s.t.  $x \in \Omega$ . (5)

where  $\Omega$  is some nonempty closed feasible set. To compare solutions of problem (5) to each other, we make use of the standard partial ordering on  $\mathbb{R}^m$ : given  $u, v \in \mathbb{R}^m$ ,  $u \leq v$  means  $u_i \leq v_i$  for all  $i \in \{1, \ldots, m\}$ ; similarly, u < v if  $u_i < v_i$  for all i; finally,

we denote by  $u \leq v$  when  $u \leq v$  and  $u \neq v$ . The latter relation allows to introduce the notion of dominance between solutions of problem (5): given  $x, y \in \Omega$ , we say that y is dominated by x if  $F(x) \not\leq F(y)$ . We are now able to introduce Pareto optimality concepts for problem (5).

**Definition 1.** A solution  $\bar{x} \in \Omega$  is:

- 1. Pareto optimal (or efficient) for problem (5) if there is no  $y \in \Omega$  such that  $F(y) \leq F(\bar{x})$ ;
- 2. weakly Pareto optimal (or weakly efficient) for problem (5) if there is no  $y \in \Omega$  such that  $F(y) < F(\bar{x})$ .

If one of the above properties holds in  $\Omega \cap \mathcal{N}(\bar{x})$  for some neighborhood  $\mathcal{N}(\bar{x})$  of  $\bar{x}$ , then  $\bar{x}$  is locally (weakly) Pareto optimal.

Clearly, Pareto optimality is a stronger property and implies weak Pareto optimality. The set of Pareto optimal solutions is called  $Pareto\ set\ (or\ efficient\ set)$ , while the image of the latter through  $F(\cdot)$  is named  $Pareto\ front\ (or\ efficient\ frontier)$ . Under differentiability assumptions on F, first-order necessary conditions for Pareto optimality can also be given.

**Lemma 1.** Let point  $\bar{x} \in \Omega$  be a (weakly) efficient solution of (5); then  $\bar{x}$  is Pareto stationary for problem (5), i.e.,

$$\min_{d \in \mathcal{D}_{\Omega}(\bar{x})} \max_{j \in \{1, \dots, m\}} \nabla f_j(\bar{x})^\top d + \frac{1}{2} ||d||^2 = 0,$$
(6)

where  $\mathcal{D}_{\Omega}(\bar{x}) = \{v \in \mathbb{R}^n \mid \exists \bar{t} > 0 : \bar{x} + tv \in \Omega \ \forall t \in [0, \bar{t}] \}$  is the set of feasible directions at  $\bar{x}$ .

We will denote by  $\theta(\bar{x})$  and  $v(\bar{x})$  the minimum and optimal solution of problem (6), respectively. Note that, when  $\Omega$  is non-convex,  $v(\bar{x})$  may not be unique.

The easiest and arguably most common practice to tackle problem (5) consists in resorting to the weighted sum scalarization method [9, 27]. The focus then shifts to solving the single-objective problem

$$P(\lambda) = \min_{x \in \Omega} \sum_{j=1}^{m} \lambda_j f_j(x), \tag{7}$$

where the weights  $\lambda = [\lambda_1, \dots, \lambda_m]^{\top} \in \mathbb{R}^m$  are such that  $\lambda_j \geq 0$  for all  $j \in \{1, \dots, m\}$  and  $\sum_{j=1}^m \lambda_j = 1$ . If we consider, for instance, the variance  $f^{V}$  and the expected return  $f^{ER}$  as objectives of the portfolio problem, we immediately retrieve the original Markovitz formulation [1, 5].

The relationship between optimal solutions of a scalarized problem and efficient solutions of the multi-objective problem can be characterized by the following concept. **Definition 2** ([27, Definition 8.7]). Let  $\bar{x}$  be an efficient solution of problem (5). If there exists some  $\lambda \in \mathbb{R}^m$  such that  $\bar{x}$  is optimal for problem  $P(\lambda)$  defined in (7), then  $\bar{x}$  is called a *supported efficient solution*, while y = F(x) is called a *supported non-dominated point*.

It is easy to prove that, for any choice of  $\lambda \geq \mathbf{0}_m$ , every globally optimal solution of  $P(\lambda)$  constitutes a Pareto optimal point for problem (5) and, thus, a *supported* 

efficient solution. The converse is in general not necessarily true. In these particular cases, scalarization might be seen as a sufficient strategy to deal with the multi-objective problem, although choosing the weights for obtaining a well spread Pareto front actually remains a nontrivial task.

#### Additional Notation for Cardinality Constraints

In the following sections, we will consider the case where the cardinality constraint  $||x||_0 \leq s$  is included in the definition of the feasible set  $\Omega$ . To deal with this setting, we need to introduce some additional notation. Given an index set  $S \subseteq \{1, \ldots, n\}$ , its cardinality is indicated with |S|, while we denote by  $\bar{S} = \{1, \ldots, n\} \setminus S$  its complementary set. Given  $x \in \Omega$ , we indicate by  $x_S$  the sub-vector of x induced by S, i.e., the vector composed by the components  $x_i$ , with  $i \in S$ . Then, we define the support set of x, i.e., the set of the indices corresponding to the nonzero components of x, as  $S_1(x) = \{i \in \{1, \ldots, n\} \mid x_i \neq 0\}$ ;  $S_0(x) = \{1, \ldots, n\} \setminus S_1(x)$  is the complementary set of  $S_1(x)$ . Finally, according to [28], we say that an index set  $S_1(x)$  is a super support set for  $S_1(x)$  if and  $S_1(x)$  is the set of all super support sets at  $S_1(x)$  is denoted by  $S_1(x)$  and  $S_1(x)$  if and only if  $S_1(x)$  if and only if  $S_1(x)$  if and only if  $S_1(x)$  is the complementary set of  $S_1(x)$  and  $S_1(x)$  if and only if  $S_1(x)$  is the set of all super support sets at  $S_1(x)$  is denoted by  $S_1(x)$  and  $S_1(x)$  if and only if  $S_1(x)$  if and only if  $S_1(x)$  is the complementary set of  $S_1(x)$ .

## 3 Limitations of Scalarization in Cardinality-constrained Problems

Although scalarization methods have arguably been the most employed among the approaches for portfolio problems (see, e.g., [1, 5, 29-31]), they become provably faulty as the problems get less regular and, in particular, when there are non-supported efficient solutions; this is the case of cardinality-constrained portfolio problems (7). Indeed, even with convex objective functions, such as variance  $f^{V}$  (3) and expected return  $f^{ER}$  (2), the presence of the cardinality constraint possibly makes problem (1) not supported anymore.

In order to illustrate this downside of the scalarization approach, let us consider a simple three-dimensional example with variance and expected return as objective functions:

$$\min_{x \in \mathbb{R}^3} F(x) = [2x_1^2 + 0.5x_2^2 + 3x_3^2, 4x_1 + 5x_2 + x_3]^\top 
\text{s.t. } \mathbf{1}_3^\top x = 1, \quad ||x||_0 \le 1, \quad x \ge \mathbf{0}_3.$$
(8)

The above problem can be seen as a mean-variance portfolio problem with cardinality constraint in multi-objective form, setting

$$Q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \qquad c = [4, 5, 1]^{\top}.$$

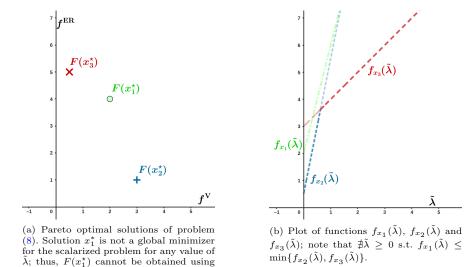


Fig. 1: Example showing the limitations of weighted-sum scalarization approach.

The corresponding weighted-sum scalarization problem is

the weighted-sum scalarization method.

$$\min_{x \in \mathbb{R}^3} 2x_1^2 + 0.5x_2^2 + 3x_3^2 + \tilde{\lambda}(4x_1 + 5x_2 + x_3) 
\text{s.t. } \mathbf{1}_3^\top x = 1, \quad ||x||_0 \le 1, \quad x \ge \mathbf{0}_3,$$
(9)

where for simplicity and with no loss of generality we consider a single weight  $\tilde{\lambda}$  associated with the second objective function.

It is trivial to see that the combination of the cardinality (s = 1) and simplex constraints induces a feasible set containing only three solutions:  $x_1^* = [1, 0, 0]^\top$ ,  $x_2^* = [0, 1, 0]^\top$  and  $x_3^* = [0, 0, 1]^\top$ ; the images of these solutions through F are shown in Figure 1a. From problem (9) we can then define the following three  $\tilde{\lambda}$ -dependent functions, each one related to one of the three solutions:

$$f_{x_1}(\tilde{\lambda}) = 2 + 4\tilde{\lambda}, \qquad f_{x_2}(\tilde{\lambda}) = 0.5 + 5\tilde{\lambda}, \qquad f_{x_3}(\tilde{\lambda}) = 3 + \tilde{\lambda};$$

the latter ones are plotted for  $\tilde{\lambda} \geq 0$  in Figure 1b.

Functions  $f_{x_1}(\tilde{\lambda})$ ,  $f_{x_2}(\tilde{\lambda})$ ,  $f_{x_3}(\tilde{\lambda})$  give us an immediate intuition of the solutions found given a value for  $\tilde{\lambda}$ . For  $0 \leq \tilde{\lambda} < \frac{5}{8}$ , where  $\tilde{\lambda} = \frac{5}{8}$  is the intersection point of lines  $f_{x_2}(\tilde{\lambda})$  and  $f_{x_3}(\tilde{\lambda})$ , the minimum value is attained by  $f_{x_2}(\tilde{\lambda})$ , i.e.,  $f_{x_2}(\tilde{\lambda}) < \min\{f_{x_1}(\tilde{\lambda}), f_{x_3}(\tilde{\lambda})\}$ ; thus, the best solution is represented by  $x_2^{\star}$ . For  $\tilde{\lambda} = \frac{5}{8}$ , we have  $f_{x_2}(\tilde{\lambda}) = f_{x_3}(\tilde{\lambda}) < f_{x_1}(\tilde{\lambda})$ :  $x_2^{\star}$  and  $x_3^{\star}$  are the two global minimizers. For  $\tilde{\lambda} \geq \frac{5}{8}$ , we get that  $x_3^{\star}$  is the optimal solution of the scalarized problem. The solution  $x_1^{\star}$  is thus never obtained through scalarization. However, we can immediately observe from Figure 1a that  $F(x_1^{\star})$  is not dominated and thus  $x_1^{\star}$  actually is an efficient solution.

The possible failure at obtaining some efficient solutions gets exacerbated with much more complex problems. A particularly troublesome situation occurs when scalarization is unable to detect any of the efficient solutions associated with a particular support set. This issue thus represents a strong motivation to devise alternative strategies to scalarization when dealing with cardinality-constrained portfolio problems. In the rest of the paper, we will describe a new methodology aimed at fully reconstructing the frontier of efficient solutions in portfolio optimization problems.

### 4 A Tailored Sparse Front Descent Method for Portfolio Problems

In this section, we propose a tailored algorithmic framework for cardinality-constrained portfolio problems, aimed at overcoming two major challenges associated with the task: i) the hardness of finding efficient solutions associated with a diverse pool of supports; ii) the lack of methods to properly span the Pareto frontier. In the next two subsections we separately address the two issues.

#### 4.1 Sparse Front Steepest Descent

In the literature, the methodologies considered to solve problems of the form (1) consist in scalarization [5, 8, 9, 32, 33] and evolutionary [11, 34–37] methods. However, both the two classes of approaches have drawbacks regarding the spanning of the Pareto front; in particular, Pareto front approximations are obtained whose spread highly depends on the (fixed) number of solutions; moreover, the selection of suitable weights to obtain a uniform front is not trivial in scalarization approaches, whereas evolutionary methods typically scale poorly as the dimensions of the problem grow [12, 14, 38]. In this work, we thus propose a framework revolving around the employment of the *Sparse Front Steepest Descent* (SFSD) algorithm [24]; for the case of MOO problems without additional constraints other than sparsity, this method was shown to be highly effective at retrieving accurate and well-spread Pareto front approximations, provided a suitable strategy is available to find starting solutions with various and good support sets.

The pseudocode of an SFSD procedure, adapted to handle the additional linear constraints commonly appearing in portfolio problems, is reported in Algorithm 1. The method starts with an initial phase (line 1) - crucial for the computational performance - where feasible solutions associated with promising supports are provided; each one is then paired with one of its super support sets, so that the resulting set is

$$\mathcal{X}^0 = \{ (x, J_x) \mid x \in \Omega, \ J_x \in \mathcal{J}(x) \}. \tag{10}$$

Then, at a generic iteration k, each point  $x_c$  in the current list of solutions is processed as follows.

• All the objectives are first decreased by an optimization step carried out in the active subspace induced by the support  $J_{x_c}$  (lines 7-12). In particular, a feasible direction of descent for all the objectives is computed (line 7) solving an instance

#### Algorithm 1 Sparse Front Steepest Descent with Linear Constraints (SFSD)

```
Require: F: \mathbb{R}^n \to \mathbb{R}^m, \, \delta \in (0,1), \, \gamma \in (0,1).
  1: \mathcal{X}^0 = \mathtt{Initialize}(F)
  2: k = 0
        while a stopping criterion is not satisfied do
  3:
                 \widehat{\mathcal{X}}^k = \mathcal{X}^k
  4:
                 for all (x_c, J_{x_c}) \in \mathcal{X}^k do
  5:
                        if (x_c, J_{x_c}) \in \widehat{\mathcal{X}}^k then
  6:
                                 if \theta_{J_{x_c}}(x_c) < 0 then
  7:
                                        \alpha_c^k = \max_{h \in \mathbb{N}} \{ \delta^h \mid F(x_c + \delta^h v_{J_{x_c}}(x_c)) \le F(x_c) + \mathbf{1}_m \gamma \delta^h \theta_{J_{x_c}}(x_c) \}
  8:
  9:
                                        \alpha_c^k = 0
 10:
                                 end if
 11:
                                z_c^k = x_c + \alpha_c^k v_{J_{x_c}}(x_c)
 12:
                                \widehat{\mathcal{X}}^k = \widehat{\mathcal{X}}^k \setminus \left\{ (y, J_y) \in \widehat{\mathcal{X}}^k \mid J_y = J_{x_c}, \ F(z_c^k) \nleq F(y) \right\} \cup \left\{ (z_c^k, J_{x_c}) \right\}
 13:
                                for all \mathcal{I} \subseteq \{1,\ldots,m\} s.t. \theta_{J_{x_c}}^{\mathcal{I}}(z_c^k) < 0 do
 14:
                                         if (z_c^k, J_{x_c}) \in \widehat{\mathcal{X}}^k then
 15:
                                               \alpha_c^{\mathcal{I}} = \max_{h \in \mathbb{N}} \left\{ \delta^h \middle| \begin{array}{l} \forall (y, J_y) \in \widehat{\mathcal{X}}^k, \ J_y = J_{x_c}, \ \exists j \text{ s.t.} \\ f_j(z_c^k + \delta^h v_{J_{x_c}}^{\mathcal{I}}(z_c^k)) < f_j(y) \end{array} \right\}
 16:
                                                \hat{z} = z_c^k + \alpha_c^{\mathcal{I}} v_{J_x}^{\mathcal{I}} \ (z_c^k)
17:
                                                \widehat{\mathcal{X}}^k = \widehat{\mathcal{X}}^k \setminus \left\{ (y, J_y) \in \widehat{\mathcal{X}}^k \mid J_y = J_{x_c}, \ F(\hat{z}) \nleq F(y) \right\} \cup \left\{ (\hat{z}, J_{x_c}) \right\}
 18:
                                         end if
 19:
                                 end for
 20:
21:
                        end if
                 end for
22:
                 \mathcal{X}^{k+1} = \widehat{\mathcal{X}}^k
23:
                 k = k + 1
25: end while
26: return \mathcal{X}^k
```

of problem

$$\theta_{J_{x_c}}(x_c) = \min_{d \in \mathbb{R}^n} \max_{j \in \{1, ..., m\}} \nabla f_j(x_c)^\top d + \frac{1}{2} ||d||^2$$
s.t.  $A(x_c + d) \le b$ ,  $\mathbf{1}_n^\top (x_c + d) = 1$ 

$$d_{\bar{J}_{x_c}} = \mathbf{0}_{|\bar{J}_{x_c}|}, \quad x_c + d \ge \mathbf{0}_n.$$
(11)

We denote by  $v_{J_{x_c}}(x_c)$  the problem solution, which we call *constrained common descent direction*. Note that, given the strong convexity of the objective function and the convexity of the feasible set of (11),  $v_{J_{x_c}}(x_c)$  is unique. An Armijo-type

line search is employed (line 8) to compute a corresponding stepsize that provides a sufficient decrease of F, leading to a new solution  $z_c^k$ .

• As long as  $z_c^k$  is not dominated by any other point y in the current list of solutions such that  $J_y = J_{x_c}$  (line 15), additional line searches in the subspace are carried out starting at  $z_c^k$  itself, this time only considering subsets of objectives  $\mathcal{I} \subseteq \{1, \ldots, m\}$  (lines 14-17). The resulting points are sought to be simply non-dominated by any other point in the current set (line 16). These "exploration" steps are carried out along constrained partial descent directions (line 14), found solving the problem

$$\theta_{J_{x_c}}^{\mathcal{I}}(z_c^k) = \min_{d \in \mathbb{R}^n} \max_{j \in \mathcal{I}} \nabla f_j(z_c^k)^{\top} d + \frac{1}{2} ||d||^2$$
s.t.  $A(z_c^k + d) \le b$ ,  $\mathbf{1}_n^{\top}(z_c^k + d) = 1$ 

$$d_{\bar{J}_{x_c}} = \mathbf{0}_{|\bar{J}_{x_c}|}, \quad z_c^k + d \ge \mathbf{0}_n.$$
(12)

We indicate with  $v_{J_{x_c}}^{\mathcal{I}}(z_c^k)$  the optimal solution of the problem; similarly as  $v_{J_{x_c}}(x_c)$ , the solution  $v_{J_{x_c}}^{\mathcal{I}}(z_c^k)$  is unique; clearly, when  $\mathcal{I} = \{1, \ldots, m\}$ , we have  $\theta_{J_{x_c}}^{\mathcal{I}}(z_c^k) = \theta_{J_{x_c}}(z_c^k)$  and  $v_{J_{x_c}}^{\mathcal{I}}(z_c^k) = v_{J_{x_c}}(z_c^k)$ .

• Every time a new point is added to the temporary list  $\widehat{\mathcal{X}}^k$ , a filtering operation is performed to delete all dominated points associated with the same super support set (lines 13-18).

It is important to remark that each point only keeps getting compared with other solutions associated with the same support set  $\mathcal{J}$ , i.e., lying in the same subspace induced by  $\mathcal{J}$ . The algorithm thus spans "in parallel" different portions of the Pareto front, one for each super support set present in the set  $\mathcal{X}$ .

**Remark 1.** Since  $\mathcal{X}^0$  contains feasible points, by definition of constrained descent direction (problems (11)-(12)) and the Armijo-type line searches (lines 8-16), we immediately get that Algorithm 1 always produces feasible solutions for problem (1).

We report here below the convergence properties of SFSD. Readers can find more details and proofs on the theoretical issues related to Algorithm 1 in Appendix A.

In order to state the convergence result, we need to introduce the set  $X_J^k = \{x \mid \exists (x, J) \in \mathcal{X}^k\}$ , with J being a super support set, and recall the concept of *linked sequence* [39].

**Definition 3** ([24, Definition 4.1]). Let  $X_J^k$  be the sequence of sets of nondominated points, associated with the super support set J, produced by Algorithm 1. We define a linked sequence as a sequence  $\{x_{j_k}^J\}$  such that, for all k, the point  $x_{j_k}^J \in X_J^k$  is generated at iteration k-1 of Algorithm 1 while processing point  $x_{j_{k-1}}^J \in X_J^{k-1}$ .

**Proposition 2.** Let us assume that  $X_J^0$  is a set of mutually nondominated points and there exists  $x_0^J \in X_J^0$  such that the set  $\widehat{\mathcal{L}}_F(x_0^J) = \bigcup_{j=1}^m \{x \in \Omega \mid f_j(x) \leq f_j(x_0^J)\}$  is compact. Moreover, let  $X_J^k$  be the sequence of sets of nondominated points, associated with the super support set J, produced by Algorithm 1, and  $\{x_{j_k}^J\}$  be any linked sequence. Then,  $\{x_{j_k}^J\}$  admits accumulation points, each one being Pareto stationary for problem (1) restricted to the subspace induced by J.

#### 4.2 Generation of Starting Points

Algorithm 1 effectively works if solutions with "good" supports are provided as input (line 1). In this section, we list some of the possible methods that can be used in this preliminary, yet crucial phase.

Weighted-sum Scalarization Method A simple choice is to solve multiple times the scalarized problem with different configurations for the weights  $\lambda$ . However, as already outlined in section 3, the scalarization method may miss "good" supports and, as a consequence, may not allow SFSD to span large portions of the Pareto front. The following strategies aim to overcome this issue.

**Descent Methods** Following the study in [24], gradient based methods can be exploited, running in a multi-start fashion, for this task:

• The first available option is the *Multi-objective Sparse Penalty Decomposition* (MOSPD) approach [23], which generates a sequence  $\{(x_{k+1}, y_{k+1})\}$  such that, for all k,  $x_{k+1}$  is approximately Pareto-stationary for the problem

$$\min_{x} Q^{\tau_k}(x, y_{k+1}) = F(x) + \mathbf{1}_m \frac{\tau_k}{2} ||x - y_{k+1}||^2$$
  
s.t.  $Ax \le b$ ,  $\mathbf{1}_n^\top x = 1$ ,  $x \ge \mathbf{0}_n$ ,

with  $\tau_k > 0$ , whereas  $y^{k+1}$  is the projection of  $x^{k+1}$  onto the sparse feasible set, i.e.,

$$y^{k+1} \in \mathop{\arg\min}_{y \in \mathbb{R}^n, y \geq \mathbf{0}_n, \|y\|_0 \leq s} \frac{1}{2} \|x^{k+1} - y\|^2.$$

Algorithmically, such a pair  $(x_{k+1}, y_{k+1})$  is obtained at each iteration by means of an alternate minimization scheme. As  $\tau_k \to \infty$ ,  $\{x^k\}$  is proved to converge to limit points  $\bar{x}$ , each one Pareto-stationary for the problem restricted to the subspace induced by one of the super support sets in  $\mathcal{J}(\bar{x})$ ; computationally, the Penalty Decomposition approach was shown to be effective at finding good quality solutions.

• A second approach, called *Multi-Objective Iterative Hard Thresholding* (MOIHT) [24], can also be adapted to deal with additional linear constraint and finally generate a solution  $x^*$  such that

$$\theta_L(x^*) = \min_{d \in \mathbb{R}^n} \max_{j \in \{1, \dots, m\}} \nabla f_j(x^*)^\top d + \frac{L}{2} ||d||^2 = 0$$
s.t.  $A(x^* + d) \le b$ ,  $\mathbf{1}_n^\top (x^* + d) = 1$   
 $||x^* + d||_0 \le s$ ,  $x^* + d \ge \mathbf{0}_n$ ,

with L>0. The above condition, called L-stationarity, is necessary for optimality and is stronger than the one achieved by MOSPD and SFSD.

A hybrid approach is also possible where MOIHT and MOSPD are run in cascade;
 the solutions resulting from the two executions are then merged to form the

set to give as input to SFSD; doing so, we can exploit both the exploration capabilities of MOSPD and the stronger L-stationarity condition to escape from bad Pareto stationary points. More information on the theoretical properties of the two approaches can be found in Appendix A.

Genetic Methods Genetic approaches, although merely heuristic, have proved to be effective with highly irregular problems. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) [11] is a famous instance of this class.

Like any other evolutionary approach, NSGA-II works with a fixed size population of N solutions; the latter is evolved at each iteration through the application of specific random-based operators: crossover, mutation and selection. In particular, the first two operations aim to generate new and hopefully better solutions from selected ones of the current population; after the addition of the new solutions, the resulting population is then reduced leaving exactly N members. The selection is driven by two metrics, i.e., ranking (sorting solutions based on the dominance relation) and  $crowding\ distance$  (measuring solution density).

Actually, NSGA-II has been already proposed to solve sparse MOO portfolio problems [34]: the only fix needed by the base algorithm is to project each point onto the feasible set before being inserting it into the population.

On the other hand, a memetic extension of NSGA-II, called NSMA, has been recently proposed for global continuous multi-objective optimization [25]: at the end of each NSGA-II iteration, all points in the population are optimized via a descent method. Here we can thus think of adapting NSMA to tackle problem (1). Instead of simply projecting the offspring solutions onto the feasible set, we can indeed carry out a sequence of steps of MOIHT: not only we are guaranteed to satisfy the sparsity constraint, but we also improve the quality of the solution, speeding up the whole process.

The preliminary phase is completed associating each of the solutions found with one of the above methods to one of its super support sets. Finally,  $\mathcal{X}^0$  is created filtering out dominated points w.r.t. F.

## 5 Computational Experiments

In this section, we present vast computational experiments aimed at identifying the most effective strategies to tackle problems of the form (1). In the following, we are firstly going to analyze the available options for generating initial solutions within SFSD, measuring the variety of "optimal" supports retrieved; then, we will show the empirical benefits of using SFSD compared to other baseline approaches; we will finally compare different SFSD-based algorithms with state-of-the-art methodologies, reporting the overall performance in a large collection of problems.

To carry out this detailed analysis, we considered a benchmark of real-world datasets consisting in both daily and monthly prices of a subset of stocks belonging to major financial indices. Similarly as in [23], we employed datasets DTS1, DTS2 and DTS3 - obtained from daily prices of the FTSE 100 index from 01/2003 to 12/2007 and containing 12, 24 and 48 securities respectively - and FF10, FF17 and FF48 - extracted from Fama/French benchmark collection, using monthly returns from

**Table 1**: Details of the datasets employed in the experimental analysis.

Dataset	n	s
DTS1	12	2,5,10
DTS2	24	2,5,10,15
DTS3	48	2,5,10,15,20,30
FF10	10	2,5,10
FF17	17	2,5,10,15
FF48	48	2,5,10,15,20,30
EuroStoxx	50	2,5,10,15,20,30
NASDAQ	54	2,5,10,15,20,30
FTSE	80	2,5,10,15,20,30,50
S&P	101	2,5,10,15,20,30,50

07/1971 to 06/2011 and containing 10, 17, 48 securities. In both cases datasets were generated following [40, 41], resulting in a set of 6 Mean-Variance portfolio problems.

We furthermore considered four datasets extracted from major financial indexes: EuroStoxx 50, NASDAQ 100, FTSE 100 and S&P 500. These datasets¹ consist of daily prices, adjusted for dividends and stock splits, and of daily ESG scores obtained from Refinitiv for the time window from 01/01/2013 to 12/31/2020. We assume each index as the benchmark for beta calculations. We selected 50, 54, 80 and 101 stocks from each index respectively. These datasets contain all the parameters required to compute the objective functions mean, variance, ESG score, Sharpe ratio and skewness and also the beta constraints. The expected returns, the covariance matrix and the coskewness tensor were computed from historical data following the sample estimation approach as in [42]. The betas for individual stocks were also computed using sample-based methods, assuming the market portfolio returns to be those of the corresponding benchmark index. For the ESG scores, we assumed the last observed value in the respective time series as the score for each stock.

For each dataset, different instances of the problem were considered, changing the value s of the upper bound on the  $\ell_0$  norm. Table 1 summarizes the details for each dataset employed in the experimentation. For the datasets EuroStoxx, NASDAQ, FTSE and S&P we considered the combinations of objectives mean-variance, SR-ESG, SR-Skew, ESG-Skew, SR-ESG-Skew, variance-mean-ESG, variance-mean-Skew and variance-mean-ESG-Skew, and for each combination we consider the problem with and without beta constraints (in the former case  $\beta_{\min}=0.8$  and  $\beta_{\max}=1.2$ ). In total, the benchmark consists of 442 problems (210 with  $s \leq 10$  and 232 with s > 10). To obtain values at similar scales for all the objective functions, we multiplied the objectives by a  $10^2$  factor for both variance and mean,  $10^{-2}$  for ESG score,  $10^{-1}$  for skewness, while no scaling was applied to Sharpe ratio.

In all the experiments we set  $10^{-7}$  as tolerance for nonzero components in the evaluation of the zero norm sparsity. In MOIHT, L is selected as 1.1 times the Lipschitz constant of  $f^V$ , i.e.,  $L = 1.1\lambda_{\max}(Q)$ , with  $\lambda_{\max}(Q)$  being the largest eigenvalue of

<sup>&</sup>lt;sup>1</sup>Daily prices and ESG scores are available at francescocesarone.com/data-sets.

matrix Q; we set  $\theta_{tol} = -10^{-7}$  as a threshold to discriminate stationary points. In MOSPD, we employ  $||x_{k+1} - y_{k+1}|| \le 10^{-3}$  as stopping condition,  $\tau_0 = 10^{-2}$ ,  $\tau_{k+1} = 2\tau_k$ ,  $\varepsilon_0 = 10^{-3}$  and  $\varepsilon_{k+1} = 0.9\varepsilon_k$ . For NSGA-II and NSMA we set the maximal population size to 100, while standard settings for crossover and mutation probabilities are used. Single-objective problems resulting from scalarization are solved (to global optimality) using Gurobi v11 [43] for problems with linear and quadratic objectives, whereas the penalty decomposition scheme [44] is used in the other cases. Gurobi v11 is also used to solve all subproblems associated with the computation of descent directions.

For SFSD we set  $\theta_{tol} = -10^{-7}$ ; in order to improve the efficiency of SFSD, points are processed by first considering those that are not dominated. Moreover, a condition based on the crowding distance [11] determines the points that are processed within a fixed support, so as to prevent the production of points too close to each other. For Armijo-type line-searches we set  $\delta = 0.5$  and  $\gamma = 10^{-4}$ .

The initial solutions for MOHyb, NSGA-II and NSMA consist of 2n feasible portfolios, where n is the dimensionality of the problem considered. The first n are selected as the components of the standard basis in  $\mathbb{R}^n$ , the remaining ones are random sparse vectors in  $\mathbb{R}^n$ . Projection of these points onto the feasible set is then performed. In MOHyb both MOIHT and MOSPD are fed with the same starting solutions as above. As for the scalarization method, 2n choices of  $\lambda$  are considered in order to span as evenly as possible the combinations of the m objectives, while a random feasible portfolio is selected as starting point when needed. Given the randomness in the initialization, we ran each experiment 5 times with different seeds.

The code<sup>2</sup> of the computational experiments reported in this section was implemented in Python3 and executed on a computer with an Intel Xeon Processor E5-2430 v2 6 cores 2.50 GHz and 32 GB RAM running Ubuntu 22.04.

#### 5.1 Analysis of Supports at Initialization

As a starting point of our experimental analysis, we need to identify the most effective way to provide SFSD with as many supports associated with Pareto solutions as possible. To accomplish this task, we approximated the true Pareto front of each problem by merging all the front reconstructions obtained by running, from 5 different seeds (4 minutes for each seed and algorithm), MOHyb, NSGA-II, NSMA and the scalarization method, all followed by SFSD (for 2 additional minutes). To improve the accuracy of the reference front, we did 5 additional runs of pure NSGA-II (6 minutes per run).

Figure 2 shows the reference fronts obtained for two bi-objective problems: the mean-variance problem on DTS2 and the SR-Skew problem on EuroStoxx. We can observe that, as the value of the  $\ell_0$  threshold s increases, the number of optimal supports found grows. This behavior is somewhat natural, as larger values of s lead to an increasing number of possible supports.

In order to evaluate each initialization strategy, we measure, for each problem, the ratio between the number of different optimal supports found over the total number of optimal supports in the reference front. Given  $S_m^{(p)}$  the set of supports of the solutions found in problem p by the method m, and  $S_R^{(p)}$  the set of supports of the solutions in

<sup>&</sup>lt;sup>2</sup>The code is publicly available at github.com/dadoPuccio/MO-Portfolio.

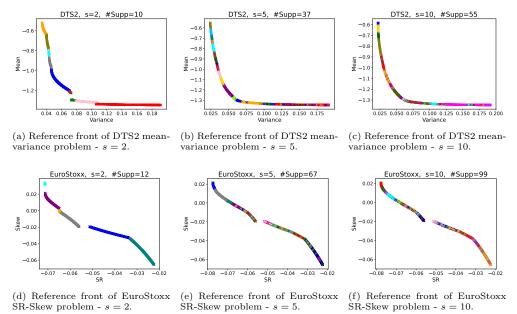


Fig. 2: Reference fronts of selected problems. Different colors denote different supports of the corresponding solutions. The title of each plot reports also the total number of optimal supports in the reference front.

the reference front of problem p, we compute the  $\mathit{recall}$  of method m in problem p as  $R_m^{(p)} = \frac{|S_m^{(p)} \cap S_R^{(p)}|}{|S_R^{(p)}|}$ . We evaluated MOHyb, NSGA-II, NSMA and the scalarization method according to this metric, running them on the entire benchmark of 442 problems. All methods share a 4 minutes time limit as a stopping condition.

Given the large number of problems considered, we summarize the outcome of this analysis by plotting the cumulative distribution of the recall values across all problems. The plots in Figure 3 present in the x-axis the recall threshold  $\tau$ , while the y-axis indicates the percentage of problems for which the recall is at least  $\tau$ . We split the analysis between small values of maximal cardinality ( $s \leq 10$ ) and large maximal cardinality (s > 10), and also between problems with and without beta constraints. We can observe that for smaller values of s the most effective methods are NSMA and NSGA-II, as their cumulative curve is consistently above other methods, meaning that a larger portion of optimal supports is generally retrieved. For greater values of s, MOHyb instead appears to be the most effective one, as the cumulative curve for this method is slightly above the others, especially for smaller values of  $\tau$ . No significant change is observed when considering problems with the beta constraints.

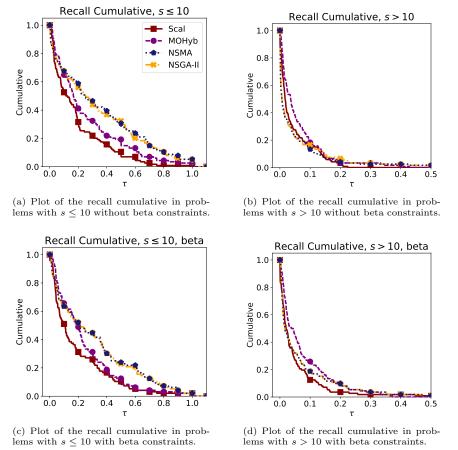


Fig. 3: Plots of the cumulative, across problems, of the ratio of optimal supports found by MOHyb, NSGA-II, NSMA and the scalarization method, relative to the optimal supports of the reference front.

#### 5.2 Impact of SFSD-based exploration

To demonstrate the benefits provided by the employment of SFSD, we compare the Pareto front reconstructions of MOHyb, NSGA-II, NSMA and the scalarization method, when SFSD is employed on top of those, against the case no further processing of points is performed after the first phase. Each "phase 1" method is run, as usual, for 4 minutes, then SFSD is used with a time limit of 2 minutes. Each algorithm is executed 5 times from different initialization, then the solutions of the run that obtain maximal hypervolume [45] are selected as the best front reconstruction for that method. Figures 4-5 report the best fronts for each algorithm when SFSD is used and not, on the DTS2 mean-variance problem (s=2) and the EuroStoxx SR-Skew problem (s=5). We observe that SFSD allows to fill the gaps in the Pareto front, offering a wider set of solutions for the problems considered. We can also note that using SFSD on top of

NSGA-II has also the additional effect of driving solutions towards stationarity, since derivatives are used only in this second phase.

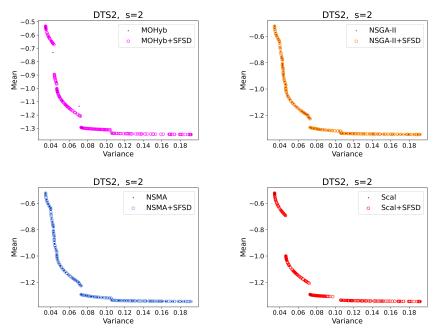


Fig. 4: Comparison of the Pareto front reconstructions of MOHyb, NSGA-II, NSMA and the scalarization method, when SFSD is used, on the DTS2 mean-variance problem in opposition to the scenario where no further processing of solutions is performed after "phase 1".

We shall observe that the solutions found by SFSD highly depend on the initialization points provided by "phase 1" methods. To better investigate on this aspect, we report in Figure 6 a comparison of the Pareto front reconstructions when using SFSD on top of MOHyb, NSGA-II, NSMA and the scalarization method, in a subset of convex and non convex problems. For the sake of completeness, we also report in the plots the "best" Pareto front reconstruction of NSGA-II executed with a time limit of 6 minutes, without SFSD.

The Pareto front reconstructions of the DTS1 mean-variance problem with s=2 are of particular interest: the scalarization method is unable to obtain any of the solutions belonging to the central portion of the Pareto front. This behavior is in line with the discussion in section 3, as some efficient solutions were not found not only because few values of the scalarization parameter  $\lambda$  were tested, but also because some portion of the Pareto front might have been impossible to reach for any value of  $\lambda$ . In the S&P SR-Skew problem with beta constraints and s=10, we observe that, without SFSD, the genetic algorithm NSGA-II struggles to obtain a good approximation

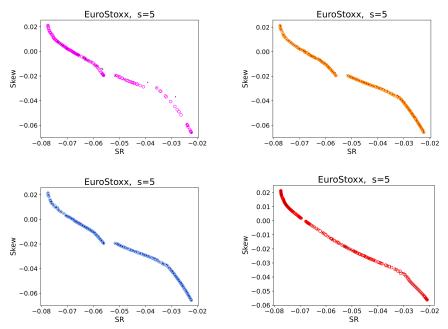


Fig. 5: Comparison of the Pareto front reconstructions of MOHyb, NSGA-II, NSMA and the scalarization method, when SFSD is used, on the EuroStoxx SR-Skew problem in opposition to the scenario where no further processing of solutions is performed after "phase 1".

of the Pareto front, even when running for 6 minutes, as in this problem not using the information of the gradient massively deteriorates the performances. This result confirms once more how SFSD is crucial, as NSGA-II with SFSD on the other hand is capable of obtaining a good front approximation.

In problems where the maximal cardinality s is large we observe that SFSD with MOHyb is the most effective approach, as it is capable of reaching a more accurate approximation of the Pareto front. In particular, in the FTSE SR-ESG problem with beta constraints and s=30, it is able of obtaining the best Pareto front reconstruction among the methods considered. On the other hand, SFSD equipped with NSGA-II or NSMA obtains better results when s is low. These numbers are in line with the observations reported in section 5.1.

#### 5.3 Overall evaluation

As a final step, we show the performance of SFSD (initialized either with MOHyb or NSMA) compared to state-of-the-art approaches, namely the scalarization and NSGA-II, in the full benchmark of 442 problems. Given the large number of problems, we make use of performance profiles [46] to summarize the outcome of the experiments. We consider standard metrics in our comparison: purity,  $\Gamma$ -spread [12] and hypervolume

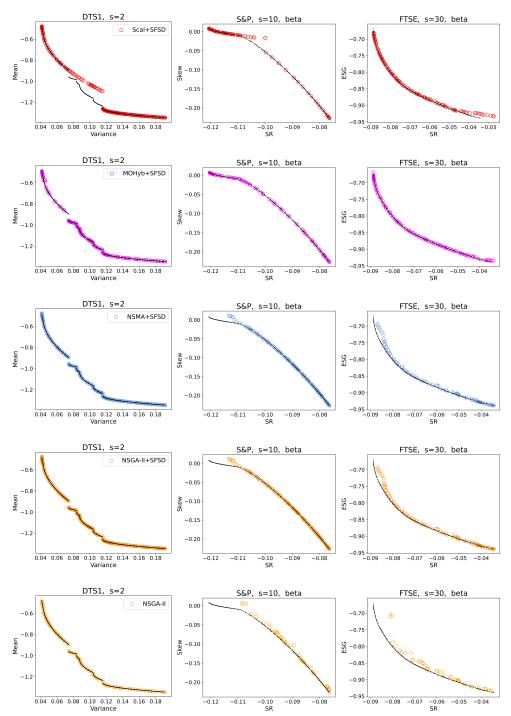


Fig. 6: Pareto front reconstruction obtained, on selected problems, with SFSD initialized using MOHyb, NSGA-II, NSMA and the scalarization method; the result of vanilla NSGA-II is also reported. The small black dots represent the reference fronts. 22

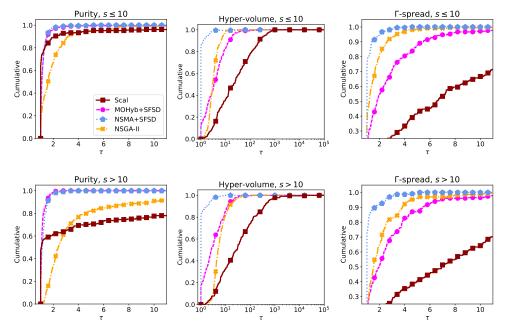


Fig. 7: Performance profiles of purity, Hyper-volume and  $\Gamma$ -spread for MOHyb and NSMA, with SFSD, and for the scalarization approach and vanilla NSGA-II. We analyze, in the first row, problems with  $s \leq 10$ ; in the second row, problems with s > 10.

[45]. The setup of the experiment is the same of Section 5.2. We split our analysis again by separately considering the 210 problems with  $s \le 10$ , and the remaining 232 with s > 10.

We show in Figure 7 the outcome of this analysis. We can observe that SFSD clearly outperforms vanilla NSGA-II across all metrics. While both initialization strategies appear to provide solid purity values, the employment of NSMA seems to be a more efficient choice overall. Somewhat unsurprisingly, the scalarization method falls far behind the other approaches, exhibiting once again its limitations in portfolio optimization.

#### 6 Conclusions

In this paper, we studied solution approaches for the multi-objective portfolio selection problem with sparsity constraints. After pointing out some serious limitations of the scalarization strategy, we proposed a composite framework, combining and suitably adapting to constrained problems some state-of-the-art methods, that is specifically designed to retrieve a well spanned front of efficient solutions.

From the experimental analysis reported in this manuscript, it is clear that the proposed Sparse Front Steepest Descent method is a very effective algorithm for completing the Pareto front of possible investment portfolios, when both convex and non-convex objective functions are considered, compared to both pure evolutionary

and scalarization methods. While we found that feeding the method with starting solutions found by the memetic NSMA procedure apparently leads to the best results, we expect that using starting points produced by the combination of different strategies could even bring better Pareto front approximations, even though at the cost of a larger computational burden.

#### **Declarations**

#### Code Availability

The implementation code of the methodologies proposed in this paper can be found at github.com/dadoPuccio/MO-Portfolio.

#### Conflict of interest

The authors declare that they have no conflict of interest.

## Appendix A Further Details on Descent Methods

In this appendix, we report some description and theoretical properties of descent algorithms mentioned throughout the paper. Although the key concepts nearly are the same as those exposed in the referenced papers, in the setting here considered we dealt with new constraints and, thus, some adjustments are needed in the theory of the presented approaches.

#### A.1 Multi-objective Sparse Penalty Decomposition

The scheme of the *Multi-objective Sparse Penalty Decomposition* (MOSPD) [23] can be found in Algorithm 2. Here below, we list the key steps of the approach.

- The algorithm starts at a point  $x_0$  which is feasible for problem (1). Here, we introduce the notation  $\Omega_c = \{x \in \mathbb{R}^n \mid Ax \leq b, \ \mathbf{1}_n^\top x = 1, \ x \geq \mathbf{0}_n\}$  to indicate the feasible set induced by the convex constraints in problem (1).
- As already anticipated in section 4.2, the method aims to find, at each iteration k, a pair  $(x_{k+1}, y_{k+1})$  such that  $y_{k+1}$  is the projection of  $x_{k+1}$  onto the sparse set and  $x_{k+1}$  is approximately Pareto stationary w.r.t.  $Q^{\tau_k}(\cdot, y_{k+1})$  with Pareto stationarity approximation degree  $\varepsilon_k \to 0$  and  $\tau_k \to \infty$  for  $k \to \infty$ . The goal is attained employing the Multi-objective Projected Gradient (MOPG) [47] method in an alternate minimization scheme.
- Before entering in the *alternate minimization loop*, a test (lines 4-8) is carried out to ensure that every next iterate will stay inside an appropriate level set; more information on this scheme part can be found in [23].
- In an alternate minimization fashion, in lines 10-13 we employ a run of MOPG on  $Q^{\tau_k}$  w.r.t. x (line 11) and a projection operator onto the sparse feasible set (line 12); when the resulting solution is approximately Pareto-stationary on  $Q^{\tau_k}$  w.r.t. x (line 10), we then increase  $\tau_k$  and decrease the Pareto stationarity approximation degree  $\varepsilon_k$  for the next iteration.

#### Algorithm 2 Multi-objective Sparse Penalty Decomposition (MOSPD)

```
Require: F: \mathbb{R}^n \to \mathbb{R}^m, x_0 = y_0 \in \Omega = \Omega_c \cap \{x \in \mathbb{R}^n \mid ||x||_0 \le s\}, \tau_0 > 0, \sigma > 1,
         \{\varepsilon_k\} \subseteq \mathbb{R}_+ \text{ s.t. } \varepsilon_k \to \infty.
   1: for k = 0, 1, \dots do
                Let Q^{\tau_k}(x, y_k) = F(x) + \mathbf{1}_m \frac{\tau_k}{2} ||x - y_k||^2
   2:
   3:
                x_{\text{trial}} = \text{MOPG}(Q^{\tau_k}(\cdot, y_k), x_k, \varepsilon_k)
   4:
                if Q^{\tau_k}(x_{\text{trial}}, y_k) \leq F(x_0) then
   5:
   6:
                        Set u_0, v_0 = x_k, y_k
   7:
                        Set u_0, v_0 = x_0, y_0
   8.
                end if
  9:
                       \begin{aligned} & \text{tile} & \min_{d \in \mathcal{D}_{\Omega_c}(u_\ell)} \max_{j \in \{1, \dots, m\}} \nabla_u q_j^{\tau_k} (u_\ell, v_\ell)^\top d + \frac{1}{2} \|d\|^2 < -\varepsilon_k & \text{do} \\ & u_{\ell+1} = \texttt{MOPG}(Q^{\tau_k}(\cdot, v_\ell), \, u_\ell, \, \varepsilon_k) \end{aligned}
 10:
 11:
                       v_{\ell+1} = \mathop{\arg\min}_{v \in \mathbb{R}^n, v \geq \mathbf{0}_n, ||v||_0 \leq s} ||u_{\ell+1} - v||^2
 12:
                        \ell = \ell + 1
 13:
 14:
                end while
                \tau_{k+1} = \sigma \tau_k
 15:
                x_{k+1} = u_{\ell}
 16:
                y_{k+1} = v_{\ell}
 18: end for
 19: return sequence \{(x_k, y_k)\}
```

In order to state the convergence properties of MOSPD, we need the following assumption.

**Assumption 1.** F has bounded level sets in the multi-objective sense, i.e., the set  $\mathcal{L}_F(z) = \{x \in \Omega \mid F(x) \leq z\}$  is bounded for any  $z \in \mathbb{R}^m$ .

Moreover, we have to state a technical lemma.

**Lemma 3.** Let  $D \subset \mathbb{R}^n$  be a (possibily non-convex) set such that if  $d \in D$  then  $td \in D$  for all  $t \in [0,1]$ ,  $A \in \mathbb{R}^{p \times n}$ ,  $x \geq \mathbf{0}_n$  and  $F : \mathbb{R}^n \to \mathbb{R}^m$  be a continuously differentiable function. If

$$\min_{d \in D} \max_{j \in \{1, \dots, m\}} \nabla f_j(x)^\top d + \frac{1}{2} ||d||^2 = 0$$
s.t.  $A(x+d) \le b$ ,  $\mathbf{1}_n^\top (x+d) = 1$ ,  $x+d \ge \mathbf{0}_n$ , (A1)

then

$$\min_{d \in D} \max_{j \in \{1, \dots, m\}} \nabla f_j(x)^\top d + \frac{1}{2} ||d||^2 = 0$$

$$s.t. \ a_i^\top d \le 0 \quad \forall i : a_i^\top x = b$$

$$\mathbf{1}_n^\top d = 0, \quad d_i \ge 0 \quad \forall i : x_i = 0.$$
(A2)

*Proof.* First, note that the constraints (excluding the set D) in problem (A2) constitute the feasible direction set  $\mathcal{D}_{\Omega_c}(x)$ . Let us denote as  $\tilde{\mathcal{D}}(x)$  the set formed by the related constraints in problem (A1). It is easy to prove that  $\tilde{\mathcal{D}}(x) \subseteq \mathcal{D}_{\Omega_c}(x)$ ,  $\mathbf{0}_n \in \tilde{\mathcal{D}}(x)$  and, finally, both sets are convex. Thus, also considering the hypothesis on the set D, we can follow the proof of [23, Lemma 4.6] to get the thesis.

We are finally ready to state the convergence properties of MOSPD.

**Lemma 4.** Let Assumption 1 hold and  $\{(x_k, y_k)\}$  be the sequence generated by Algorithm 2. Then,  $\{(x_k, y_k)\}$  admits cluster points. Moreover, let  $(\bar{x}, \bar{y})$  a limit point of the sequence, i.e., there exists  $K \subseteq \{0, 1, \ldots\}$  such that  $(x_k, y_k) \to (\bar{x}, \bar{y})$  for  $k \to \infty$ ,  $k \in K$ . We have the following conditions hold:

- 1.  $\bar{x}$  is feasible for problem (1);
- 2.  $\bar{x}$  is Pareto stationary in a subspace associated to one of its super support set for problem (1).

*Proof.* The theses straightforwardly follow as in [23, Propositions 4.5, 4.7], respectively, where Lemma 3 can be used in place of Lemma 4.6 of the referenced paper.  $\Box$ 

Remark 2. The stationarity condition of MOSPD has been referred to as Multi-objective Lu-Zhang first-order (MOLZ) conditions [23]. In the cited paper, MOLZ conditions are shown to be weaker than standard Pareto stationarity for problem (1) (see Lemma 1): while the latter one implies MOLZ conditions, a point must satisfy MOLZ conditions for every associated super support set to be Pareto stationary.

#### A.2 Multi-objective Iterative Hard Thresholding

The scheme of the *Multi-objective Iterative Hard Thresholding* (MOIHT) [24] is reported in Algorithm 3.

#### Algorithm 3 Multi-objective Iterative Hard Thresholding (MOIHT)

**Require:**  $F: \mathbb{R}^n \to \mathbb{R}^m, x_0 \in \Omega, L > \max_{j \in \{1, ..., m\}} L(\nabla f_j).$ 

- 1: k = 0
- 2: while  $x_k$  is not L-stationary for problem (1) do
- 3: Compute

$$v_{L}(x_{k}) \in \underset{d \in \mathbb{R}^{n}}{\operatorname{arg \, min}} \ \underset{j \in \{1, \dots, m\}}{\operatorname{max}} \nabla f_{j}(x_{k})^{\top} d + \frac{L}{2} \|d\|^{2}$$
s.t.  $A(x_{k} + d) \leq b, \quad \mathbf{1}_{n}^{\top}(x_{k} + d) = 1$ 

$$\|x_{k} + d\|_{0} \leq s, \quad x_{k} + d \geq \mathbf{0}_{n}.$$
(A3)

- 4:  $x_{k+1} = x_k + v_L(x_k)$
- 5: Let k = k + 1
- 6: end while
- 7: **return** sequence  $\{x_k\}$

For a brief description of the method mechanisms, we need first an assumption on the objective functions gradients.

**Assumption 2.** For all  $j \in \{1, ..., m\}$ ,  $\nabla f_j$  is Lipschitz-continuous over  $\mathbb{R}^n$  with constant  $L(\nabla f_j)$ , i.e.,  $\|\nabla f_j(x) - \nabla f_j(y)\| \le L(\nabla f_j)\|x - y\|$  for all  $x, y \in \mathbb{R}^n$ .

The algorithm starts from an initial feasible point  $x_0 \in \Omega$  and iteratively updates the point until it is L-stationary, with L being an input of the approach. At each iteration, the update is performed by solving problem (A3). Note that, by definition of the feasible set of problem (A3), all the iterates generated by MOIHT are feasible for problem (1). Moreover, managing the cardinality constraint in problems like (A3) is actually a practical operation to be carried out with mixed-integer quadratic solvers.

The convergence properties directly follows from the theoretical analysis of MOIHT in [24]: indeed, the presence of constraints other than the cardinality one does not spoil any reported proof.

**Lemma 5** ([24, Proposition 4.1]). Let Assumptions 1-2 hold and  $\{x_k\}$  be the sequence generated by Algorithm 3 with constant  $L > \max_{j \in \{1,...,m\}} L(\nabla f_j)$ . Then, the sequence admits cluster points, each one being L-stationary for problem (1).

#### A.3 Sparse Front Steepest Descent

The Sparse Front Steepest Descent (SFSD) is already described in section 4.1. Moreover, by definition of the constrained descent directions (problems (11)-(12)) and the fact that the first tentative step size in the Armijo-type line searches is  $\alpha_0 = 1$ , most of theoretical properties of the approach, including the finite termination of the line searches, are already stated and proved in [24, 48]. Here, we only report the proof of Proposition 2, which must take into account the new feasible set  $\Omega$ .

**Proposition 2.** Let us assume that  $X_J^0$  is a set of mutually nondominated points and there exists  $x_0^J \in X_J^0$  such that the set  $\widehat{\mathcal{L}}_F(x_0^J) = \bigcup_{j=1}^m \{x \in \Omega \mid f_j(x) \leq f_j(x_0^J)\}$  is compact. Moreover, let  $X_J^k$  be the sequence of sets of nondominated points, associated with the super support set J, produced by Algorithm 1, and  $x_{j_k}^J$  be a linked sequence. Then, the latter admits accumulation points, each one being Pareto stationary in the subspace associated with J for problem (1).

*Proof.* By instructions of the algorithms, each linked sequence  $\{x_{j_k}^J\}$  can be seen as generated by applying front steepest descent [13, 48] to the problem

$$\begin{aligned} \min_{x \in \mathbb{R}^n} F(x) \\ \text{s.t. } Ax &\leq b, \quad \mathbf{1}_n^\top x = 1 \\ x_{\bar{J}} &= \mathbf{0}_{|\bar{J}|}, \quad x \geq \mathbf{0}_n, \end{aligned}$$

employing as descent directions the constrained ones proposed in equations (11)-(12) and initial step size  $\alpha_0=1$ . Thus, following the same reasoning used for proof of [48, Propositiom 3.4], we can straightforwardly prove that each accumulation point  $\bar{x}$  of the linked sequence  $\{x_{j_k}^J\}$  is such that  $\theta_{J_{x_c}}(\bar{x})=0$ . Recalling now Lemma 3, we finally get that  $\bar{x}$  is Pareto stationary in the subspace associated with J for problem (1).  $\square$ 

## References

- [1] Markowitz, H.: Portfolio selection. The Journal of Finance **7**(1), 77–91 (1952) https://doi.org/10.1111/j.1540-6261.1952.tb01525.x
- [2] Sharpe, W.F.: The Sharpe Ratio (Fall 1994). Streetwise: The Best of The Journal of Portfolio Management, pp. 169–178. Princeton University Press, 41 William Street, Princeton, NJ 08540-5237, USA (1998). https://doi.org/10.2307/ j.ctv1mjqtwg.24
- [3] Pedersen, L.H., Fitzgibbons, S., Pomorski, L.: Responsible investing: The esg-efficient frontier. Journal of Financial Economics **142**(2), 572–597 (2021) https://doi.org/10.1016/j.jfineco.2020.11.001
- [4] De Athayde, G.M., Flôres, R.G.: Finding a maximum skewness portfolio—a general solution to three-moments portfolio choice. Journal of Economic Dynamics and Control 28(7), 1335–1352 (2004) https://doi.org/10.1016/S0165-1889(02) 00084-2
- [5] Markowitz, H.M.: The general mean-variance portfolio selection problem. Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences **347**(1684), 543–549 (1994)
- [6] Steuer, R.E., Utz, S.: Non-contour efficient fronts for identifying most preferred portfolios in sustainability investing. European Journal of Operational Research 306(2), 742–753 (2023) https://doi.org/10.1016/j.ejor.2022.08.007
- [7] Mandal, P.K., Thakur, M.: Higher-order moments in portfolio selection problems: A comprehensive literature review. Expert Systems with Applications 238, 121625 (2024) https://doi.org/10.1016/j.eswa.2023.121625
- [8] Pascoletti, A., Serafini, P.: Scalarizing vector optimization problems. Journal of Optimization Theory and Applications 42(4), 499–524 (1984) https://doi.org/10. 1007/BF00934564
- [9] Eichfelder, G.: An adaptive scalarization method in multiobjective optimization.
   SIAM Journal on Optimization 19(4), 1694–1718 (2009) https://doi.org/10.1137/060672029
- [10] Fliege, J., Vaz, A.I.F.: A method for constrained multiobjective optimization based on sqp techniques. SIAM Journal on Optimization 26(4), 2091–2119 (2016) https://doi.org/10.1137/15M1016424
- [11] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002) https://doi.org/10.1109/4235.996017

- [12] Custódio, A.L., Madeira, J.F.A., Vaz, A.I.F., Vicente, L.N.: Direct multisearch for multiobjective optimization. SIAM Journal on Optimization **21**(3), 1109–1140 (2011) https://doi.org/10.1137/10079731X
- [13] Lapucci, M., Mansueto, P., Pucci, D.: Effective Front-Descent Algorithms with Convergence Guarantees (2024). https://arxiv.org/abs/2405.08450
- [14] Cocchi, G., Lapucci, M., Mansueto, P.: Pareto front approximation through a multi-objective augmented lagrangian method. EURO Journal on Computational Optimization 9, 100008 (2021) https://doi.org/10.1016/j.ejco.2021.100008
- [15] Garcia-Bernabeu, A., Hilario-Caballero, A., Tardella, F., Pla-Santamaria, D.: Esg integration in portfolio selection: A robust preference-based multicriteria approach. Operations Research Perspectives 12, 100305 (2024) https://doi.org/ 10.1016/j.orp.2024.100305
- [16] Radziukynienė, I., Žilinskas, A.: Evolutionary methods for multi-objective portfolio optimization. In: Proceedings of the World Congress on Engineering, vol. 2, pp. 1155–1159 (2008)
- [17] Armananzas, R., Lozano, J.A.: A multiobjective approach to the portfolio optimization problem. In: 2005 IEEE Congress on Evolutionary Computation, vol. 2, pp. 1388–13952 (2005). https://doi.org/10.1109/CEC.2005.1554852
- [18] Chen, C., Wei, Y.: Robust multiobjective portfolio optimization: a set order relations approach. Journal of Combinatorial Optimization **38**(1), 21–49 (2019) https://doi.org/10.1007/s10878-018-0364-9
- [19] Cornuejols, G., Peña, J., Tütüncü, R.: Optimization Methods in Finance. Cambridge University Press, Cambridge (2018)
- [20] Bertsimas, D., Cory-Wright, R.: A scalable algorithm for sparse portfolio selection. INFORMS Journal on Computing **34**(3), 1489–1511 (2022) https://doi.org/10.1287/ijoc.2021.1127
- [21] Bertsimas, D., Shioda, R.: Algorithm for cardinality-constrained quadratic optimization. Computational Optimization and Applications 43(1), 1–22 (2009) https://doi.org/10.1007/s10589-007-9126-9
- [22] Bienstock, D.: Computational study of a family of mixed-integer quadratic programming problems. Mathematical Programming **74**(2), 121–140 (1996) https://doi.org/10.1007/BF02592208
- [23] Lapucci, M.: A penalty decomposition approach for multi-objective cardinality-constrained optimization problems. Optimization Methods and Software 37(6), 2157–2189 (2022) https://doi.org/10.1080/10556788.2022.2060972

- [24] Lapucci, M., Mansueto, P.: Cardinality-constrained multi-objective optimization: Novel optimality conditions and algorithms. Journal of Optimization Theory and Applications 201(1), 323–351 (2024) https://doi.org/10.1007/s10957-024-02397-3
- [25] Lapucci, M., Mansueto, P., Schoen, F.: A memetic procedure for global multiobjective optimization. Mathematical Programming Computation **15**(2), 227–267 (2023) https://doi.org/10.1007/s12532-022-00231-3
- [26] Pedro Brito, R., Sebastião, H., Godinho, P.: Efficient skewness/semivariance portfolios. Journal of Asset Management 17(5), 331–346 (2016) https://doi.org/10. 1057/jam.2016.9
- [27] Ehrgott, M.: Multicriteria Optimization vol. 491. Springer, Berlin/Heidelberg, Germany (2005)
- [28] Beck, A., Hallak, N.: On the minimization over sparse symmetric sets: Projections, optimality conditions, and algorithms. Mathematics of Operations Research 41(1), 196–223 (2016) https://doi.org/10.1287/moor.2015.0722
- [29] Markowitz, H.M., Todd, G.P.: Mean-variance Analysis in Portfolio Choice and Capital Markets vol. 66. John Wiley & Sons, Hoboken, New Jersey, United States (2000)
- [30] HM, M.: Portfolio selection: efficient diversification of investments. New York: John Wiley Sons (1959)
- [31] Cesarone, F., Scozzari, A., Tardella, F.: An optimization–diversification approach to portfolio selection. Journal of Global Optimization **76**(2), 245–265 (2020)
- [32] Fliege, J.: Gap-free computation of Pareto-points by quadratic scalarizations. Mathematical Methods of Operations Research **59**(1), 69–89 (2004)
- [33] Drummond, L.G., Maculan, N., Svaiter, B.F.: On the choice of parameters for the weighting method in vector optimization. Mathematical Programming 111(1-2), 201–216 (2008)
- [34] Chen, B., Lin, Y., Zeng, W., Xu, H., Zhang, D.: The mean-variance cardinality constrained portfolio optimization problem using a local search-based multiobjective evolutionary algorithm. Applied Intelligence 47(2), 505–525 (2017) https://doi.org/10.1007/s10489-017-0898-z
- [35] Laumanns, M., Thiele, L., Deb, K., Zitzler, E.: Combining convergence and diversity in evolutionary multiobjective optimization. Evolutionary Computation 10(3), 263–282 (2002)

- [36] Mostaghim, S., Branke, J., Schmeck, H.: Multi-objective particle swarm optimization on computer grids. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 869–875 (2007). ACM
- [37] Konak, A., Coit, D.W., Smith, A.E.: Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering & System Safety **91**(9), 992–1007 (2006)
- [38] Cocchi, G., Liuzzi, G., Lucidi, S., Sciandrone, M.: On the convergence of steepest descent methods for multiobjective optimization. Computational Optimization and Applications 77(1), 1–27 (2020) https://doi.org/10.1007/s10589-020-00192-0
- [39] Liuzzi, G., Lucidi, S., Rinaldi, F.: A derivative-free approach to constrained multiobjective nonsmooth optimization. SIAM Journal on Optimization **26**(4), 2744–2774 (2016) https://doi.org/10.1137/15M1037810
- [40] Brito, R.P., Vicente, L.N.: Efficient cardinality/mean-variance portfolios. In: Pötzsche, C., Heuberger, C., Kaltenbacher, B., Rendl, F. (eds.) System Modeling and Optimization, pp. 52–73. Springer, Berlin, Heidelberg (2014)
- [41] Cocchi, G., Levato, T., Liuzzi, G., Sciandrone, M.: A concave optimization-based approach for sparse multiobjective programming. Optimization Letters 14(3), 535–556 (2020) https://doi.org/10.1007/s11590-019-01506-w
- [42] Lai, T.-Y.: Portfolio selection with skewness: A multiple-objective approach. Review of Quantitative Finance and Accounting 1, 293–305 (1991) https://doi. org/10.1007/BF02408382
- [43] Gurobi Optimization, LLC: Gurobi Optimizer Reference Manual (2024). https://www.gurobi.com
- [44] Lapucci, M., Levato, T., Sciandrone, M.: Convergent inexact penalty decomposition methods for cardinality-constrained problems. Journal of Optimization Theory and Applications 188, 473–496 (2021)
- [45] Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms—a comparative case study. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) Parallel Problem Solving from Nature—PPSN V, pp. 292–301. Springer, Berlin, Heidelberg (1998)
- [46] Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Mathematical Programming 91(2), 201–213 (2002) https://doi.org/10.1007/s101070100263
- [47] Drummond, L.M.G., Iusem, A.N.: A projected gradient method for vector optimization problems. Computational Optimization and Applications **28**(1), 5–29

 $(2004)\ \mathrm{https://doi.org/10.1023/B:COAP.0000018877.86161.8b}$ 

[48] Lapucci, M., Mansueto, P.: Improved front steepest descent for multi-objective optimization. Operations Research Letters **51**(3), 242–247 (2023) https://doi.org/10.1016/j.orl.2023.03.001