Metaheuristics for the Portfolio Selection Problem

Giacomo di Tollo
Dipartimento di Scienze
Università "G.D'Annunzio" Chieti–Pescara
Viale Pindaro 42 - Pescara (Italia)
ditollo@sci.unich.it

Andrea Roli
DEIS
Università di Bologna
Campus of Cesena
Via Venezia 52 - Cesena (Italia)
andrea.roli@unibo.it

Abstract

The Portfolio selection problem is a relevant problem arising in finance and economics. Some practical formulations of the problem include various kinds of nonlinear constraints and objectives and can be efficiently solved by approximate algorithms. Among the most effective approximate algorithms, are metaheuristic methods that have been proved to be very successful in many applications. This paper presents an overview of the literature on the application of metaheuristics to the portfolio selection problem, trying to provide a general descriptive scheme.

1 Introduction

Portfolio selection is one of the most relevant and studied topics in finance. The problem, in its basic formulation, is concerned with selecting the portfolio of assets that minimizes the risk subject to the constraint of guaranteeing a given level of returns. Individuals and institutions prefer to invest in portfolios rather than single assets (or securities) because it enables them to dampen the risk, by diversification of the investments, without negatively affecting expected returns. In this paper, we deal with the so-called Mean-Variance portfolio selection (hereinafter referred to as PSP), formulated in the seminal work by Markowitz[62]. In that work, the author rejects the hypothesis that investors wish to maximize expected returns, because this criterion does not imply that a diversified portfolio is preferable to a non-diversified one. Thus, he states that the goal is to select a portfolio with minimum risk at given minimal returns. Alternatively, the problem can be formulated as a multi-criteria optimization problem in which risk has to be minimized while return has to be maximized. Notwithstanding its potential in capturing the basic properties of the problem suffers from several drawbacks. First, it might be difficult to gather enough data and information for estimating risk and returns. Second, the estimation of return and covariance (used for defining the risk) from historical data is very sensitive to measurement errors[15]. Finally, it is nowadays considered too simplistic for practical purposes, because it does not incorporate non-negligible aspects of real-world trading, such as maximum size of portfolio, minimum lots, transaction costs, preferences over assets, management costs, etc. These aspects can be modeled by adding constraints to the original formulation, leading to the *constrained PSP*: This problem has been shown to be NP-Complete[56]. In some cases, problem characteristics, such as its size, or real-world requirements, such as very limited computation time allowed or limited precision in estimating instance parameters, make exact methods not particularly suitable for tackling large instances of the constrained PSP, therefore researchers and practitioners have to resort to approximate algorithms and, in particular, to metaheuristics and hybrid techniques[11]. Among such approaches can now be found state-of-the-art solvers for some formulations of the PSP.

In this work, we give an overview of the use of metaheuristic techniques to solve the PSP. We first present and discuss the different models from the literature on metaheuristics for the PSP and we also introduce a classification of them, that can provide a general scheme for analyzing and comparing such models. Then, we survey the most relevant metaheuristic approaches for the PSP. The distinction between model and solving technique is becoming particularly effective in the recent years, due to the development of constraint programming-oriented approaches, as demonstrated by recent successes of software tools such as Comet [36], ILOG Solver [39] and EasyLocal++ [20].

The paper begins with illustrating the main motivations of this work in Sec. 2. In Sec. 3 we provide an incremental description of the various models of PSP used in metaheuristic applications. The problem model is considered as an object with three attributes: decision variables and their domains, objectives and constraints. On the basis of such attributes, we also provide classification such that each actual problem formulation can be seen as an instance of a general abstract model, the basic (or *default*) instance of which is the Markowitz model. Sec. 4 presents the various metaheuristic approaches to the PSP by analyzing them through a general framework for metaheuristics called MAGMA [64]. First, the basic building block of algorithms based on metaheuristics are presented, such as the search space, the neighborhood structures and the cost function. Then, we overview the most important techniques from the literature, starting from solution construction procedures till advanced search strategies. Sec. 5 summarizes the most important works form the literature that explicitly address the issue of comparing different metaheuristic approaches for the PSP. Finally, in Sec. 6 we briefly summarize related works and we conclude with Sec. 7 outlining future research and application directions in the field.

2 Motivations

Research on design and implementation of effective and efficient PSP solvers is very active. Many different formulations and approaches have been proposed in the literature. In the last decade, a large number of publications concerning the application of heuristic techniques to the PSP has appeared. On the other side, metaheuristic-based solvers are nowadays the state of the art in many real-world constrained combinatorial optimization problems [11][37]. Moreover, recent advances in methodologies and tools for developing metaheuristic-based solvers, such as development frameworks [20] and languages [39][36], are making them more appealing also as engineering techniques for general problem solving in the industry. For this reason, we believe that, by providing a uniform and general description of the literature on metaheuristics for PSP, we may help the development of more advanced solvers also for other formulations of portfolio selection problems. The literature on this specific subject is not homogeneous —in terminology, definitions and goals— and very fragmented. Furthermore, it is common to encounter publications with no clear distinction between problem model and solving algorithm, thus making it very difficult to re-use this piece of knowledge. With this contribution we aim at survey the literature on metaheuristics applied to PSP trying to provide a general scheme in which locate the different approaches. This would enable scholars and practitioners to identify strengths and weaknesses of current modelling and solving approaches and provide them with some hints for the developments of new hybrid solvers for this class of financial problems.

3 PSP Modeling

Constrained optimization problems can be defined by specifying variables, along with their domains, objectives and constraints among variables. These entities can also play the role of model *attributes* and serve as the basis for a classification of the different models. Attributes may have several qualifications, that, in turn, may be subdivided in more detailed categories, till reaching the specification of the actual attribute instantiation. For instance, objectives (an attribute) can either be single or multi-criteria (qualifications); each qualification can be specified by instantiating the actual objective function, for example the minimization of a given risk measure.

In this section, we provide an overview of the models that can be found in the literature on metaheuristics applied to the PSP, trying to capture the diverse formulations by means of a unique classification, with the aim of giving a general view of PSP modeling along with the possibility of making comparison among the models. We first present the Markowitz model, that constitutes the basis upon which the other models are obtained as variations and extensions. This description has not the goal of providing an overview of all the formulations of the PSP, but rather of illustrating, in a unifying view, the diverse models of the problem as described in the specific literature on metaheuristics.

3.1 The basic model: Markowitz model

In the PSP in canonical form we want to find a portfolio that minimizes the risk at given levels of return rate[1]. In the Markowitz formulation the risk measure is given by the variance of the portfolio. This measure is the objective function most commonly used in related works.

The Markowitz model [62] is as follows:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j \tag{1}$$

subject to

$$\sum_{i=1}^{n} r_i x_i \geq r_p \tag{2}$$

$$\sum_{i=1}^{n} x_i = 1 \tag{3}$$

$$x_i \geq 0 \qquad i = 1 \dots n \tag{4}$$

where n is the number of assets and x_i is the proportion of money invested in asset i. For each asset the rate of return is represented by a random variable R_i, whose mean is given by r_i and represents the expected return (per period) of asset i; σ_{ij} is the real-valued covariance of expected returns on assets i and j. The objective function is the variance (herein called risk-measure) σ_p^2, given by $\sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$. The portfolio return is represented by a random variable and the expected return is given by $\sum_{i=1}^{n} r_i x_i$, whilst r_p represents the minimum required portfolio return. Constraint (3) ensures that asset weights sum up to one, as they are considered as fractions of the whole amount of money to be invested.

The Markowitz model can be considered as the most simple formulation of the PSP. Its conceptual representation is depicted in Fig. 1. Note that the three attributes, variables, objectives and constraints, can be directly instantiated, as in the case of constraints, or further detailed through qualifications. This basic model can be varied and extended in many ways. Every modification can be viewed as the result of the combination of simple variations, each of which affecting only one attribute. For instance, different risk measures can be chosen, or constraints that make the model more realistic can be added. The problem we consider in this paper is a 'single-period' (i.e., single-stage) problem; in particular, we do not take into account possible adjustments between estimated and actual returns. Moreover, the PSP formulations we discuss are deterministic.

The works we are going to survey take into account only portfolios composed of stocks. This is a simplification of real-world market, but the proposed approaches can be easily extended to other kind of securities like options, future contracts and commodities, as well as to features as adjusted returns, dividends, spreads. The case of debentures[2] (conceived as risk-free assets, as their return is fixed and their variance is null) deserve instead more attention: They were investigated by Tobin[82, 83], removing the condition that each asset must be risky. Any combination of a risk-free asset s and a portfolio P composed of risky assets will show a correlation between the expected return and the variance:

$$r = r_s + \frac{r_P - r_s}{\sigma_P} \cdot \sigma \tag{5}$$

where r and σ are, respectively, the expected return and the variance of the combination of risk free asset and risky portfolio, r_s is the return of risk-free asset and r_P and σ_P are, respectively, the expected return and the variance of the risky portfolio. The ratio $\frac{r_P - r_s}{\sigma P}$ is referred to as *reward-to-variability ratio* and used is some works, see [60, 58] and section (4.2.8). In this framework, the investment decision is split in two phases: First determine the risky

[1] In agreement with the main literature on the subject, here we consider the problem objective as the minimization of the risk measure. The problem can also be modelled as a maximization of returns or in other ways. See [27] for a brief discussion on this topic.

[2] Examples are given by corporate bonds, mortgage bonds, gold bonds, common bonds.

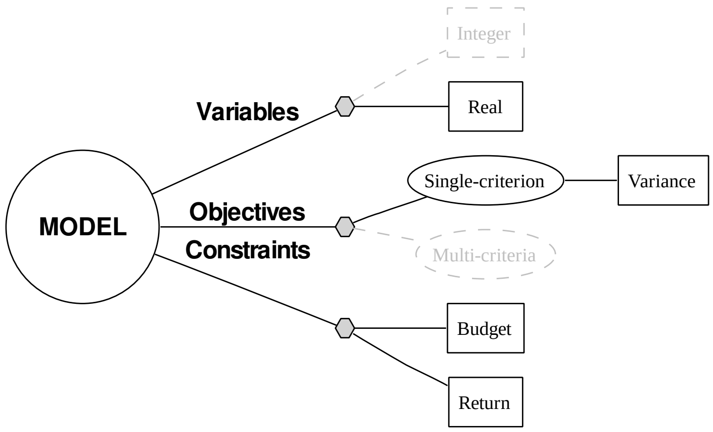

Figure 1: Conceptual representation of the Markowitz model. Rectangles represent the instantiation of a qualification (ellipse). In gray, qualifications and instantiations not present in the model.

portfolio P optimizing the *reward-to-variability ratio* and then combine it with the risk-free asset (*separation theorem*). In this way, the risk-aversion of investors does not affect the choice of the risky portfolio, but only the proportions to be assigned to the risky portfolio P and to the risk free-asset s: A very risk-averse investor will allocate a huge proportion of his endowment to the risk-free asset, while a very risky one will prefer to invest more in the risky portfolio.

In the following, we will detail the most important extensions of the basic model, by keeping in the background the conceptual model scheme.

3.2 Variables and domains

We first briefly discuss the possible choices for variable domains in a PSP model. In the Mean-Variance model, variables are real and they range between zero and one, as they represent the fraction of available money to invest in an asset[3]. This choice is quite 'natural' and has the advantage of being independent of the actual budget. Conversely, another possibility is to choose integer values for variables and make them range between zero and the maximum available budget.[4] When variables are integer, it is possible to add to the model constraints that involve actual budget values, such as minimum trading lots and also introduce more realistic objective functions. The integer formulation better explains real-world situations: For instance, it turns out that small investors are more sensitive to integer constraint on variables, as the resulting portfolios are less diversified[60]. Other advantages and disadvantages of the two approaches will be discussed in the following sections, in which variations in the basic model are presented.

3.3 Objective functions

In the most common PSP formulations, the objective can be either to minimize the risk (satisfying a given return), or maximize the return (not exceeding a given maximum risk), or both. In the former cases the problem is single-criterion, while in the latter case it is multi-objective. Metaheuristics have been mostly applied to single-criterion models, but there are some notable works which deal with multi-criteria models, such as [81][25][68]. The applications on the single-objective formulation (in which the risk has to be minimized) very often solve a PSP instance as a function of the desired expected return R, that is then seen as an instance parameter. Solving the instance for R ranging over values from a finite set, gives an estimation of the efficient frontier that could be drawn by directly solving the bi-objective formulation. In these cases, it is com-

[3]For the sake of simplicity, we are not taking into account short sales yet. They will be introduced in section 3.8.

[4]In the presence of the so-called rounds, the domain values correspond to the number of rounds. 3.9

mon to use the expression 'efficient frontier' also for that set of solution points, even if it is just an approximation of such a frontier.

In our classification, we consider these two qualifications for the attribute *objectives*, as shown in Fig. 2.

3.3.1 Single-criterion objectives

Although metaheuristics have been successfully applied to tackle both single and multi-criteria optimization problems, the PSP has been mostly modeled as a single-criterion optimization problem.

A way of tackling the problem in a single-criterion fashion consists in including constraint (2) in the objective function in a Lagrangean relaxation fashion [14] [89] [46]:

$$\max\ (1-\lambda)\sum_{i=1}^{n} r_i x_i - \lambda \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} x_i x_j \qquad (6)$$

subject to constraints (3) and (4), where λ is a trade-off coefficient ranging in [0, 1]. If $\lambda = 0$ the investor completely disregards risk and aims to maximize returns; conversely, when $\lambda = 1$, the investor is risk-adverse and only wants to minimize risk. By resolving the problem for a set of values of λ it is possible to estimate the *efficient frontier* for the Markowitz unconstrained problem (referred to as UEF). The investor can then choose the portfolio depending on specific risk/return requirements. The UEF is composed of *Pareto* optimal solution, i.e., solutions such that no criterion can be improved without deteriorating any other criterion. In our example, a solution s is said to be efficient (Pareto-optimal) if there is no other solution s_1 such that $return(s_1) > return(s)$ and $risk(s_1) \leq risk(s)$ or $return(s_1) \geq return(s)$ and $risk(s_1) < risk(s)$.

A different way or drawing out the (approximation of the) Pareto front is to solve the single-criterion objective (equations (1) (4)) for a set of equally distributed values for the minimum required return r_p, so solving several instances of the problem introducing different values of r_p in constraint (2) in order to obtain a frontier composed of equally distanced points [73][14]. In this way the efficient frontier is bounded by the minimum risk portfolio (MRP), defined as the solution of the problem without lower bound on return ($r_p = 0$). MRP has its own return r_{mrp}: It follows that solving instances whose $r_p \leq r_{mrp}$ the solution will be the MRP itself, whilst for larger values of r_p the solution will consist of portfolios so that $\sum_{i=1}^{n} x_i r_i = r_p$.

As metaheuristics provides an approximation of the actual Pareto frontier, in the following we will distinguish between the actual efficient frontier (UEF) and the approximated one (AUEF). Moreover, since we are going to introduce other classes of constraints in our discussion, we will refer to the constrained efficient frontier as CEF, whilst its approximation will be referred to as ACEF. We notice here that the unconstrained frontier dominates the constrained one and the goal of most works introducing metaheuristics tackling the PSP is to draw out the CEF for the problem at hand, so we assume that the algorithms we will discuss are aimed at drawing out the CEF instead of single portfolios, if not differently explicitly stated. The class of constraints introduced varies amongst works and will be explained for each implementation.

It worth noticing here that the Mean-Variance formulation presents its main drawbacks in being incompatible with the axiomatic models of preference for choice under risk [86] and lacking in coherence[6][5]. This consideration, together with the ones previously mentioned (see section 1) motivated researchers to define other risk measures: So far we have only considered variance as the risk measure, but other different measures can be taken, thus defining different objective functions. Markowitz himself suggested the use of semi-variance instead of variance in order to assess portfolio risk. Semi-variance can be defined as

$$semivar = \sum_{j: r_j \leq E[R]} p_j (r_j - E[R])^2 \qquad (7)$$

where R is a distribution of returns, often statistically computed by enumerating the most probable scenarios, r_j is the return of the j-th element of the distribution, p_j its probability and $E[R]$ the mean of the distribution. This measure is equivalent to variance if return distribution is symmetric around the mean and captures the essence of risk as perceived by investors, characterized by the likelihood

[5]A risk measure is said to be coherent when it fulfills properties of *translation invariance, subadditivity, positive homogeneity and monotonicity*, see Artzner et al.[6] for details.

of incurring into a loss. Its drawback is that an investor can perceive the loss not necessarily when returns are below the mean, but below some other subjective threshold τ. This idea refers to the part of distribution below a certain target of return, and for this reason the corresponding measures are referred to as *target downside risk* measures:

$$DSR(\tau) = \sum_{r_i \leq \tau} p_i(\tau - r_i)^q \tag{8}$$

When $q = 2$ the formula is referred to as *target semi-variance* expression; in this case if $\tau = E[r]$ the formula is equivalent to semi-variance.

The threshold τ is referred to as Value-at-Risk (*VaR*) and can be conceived as a measure of the portfolio catastrophic risk, since investors are concerned with the chance of loosing their wealth because of a low-probability-high-impact-event[81]. τ has been used as the threshold below which the investor perceives a loss[29, 30, 61, 60] and in their context VaR is bounded in the constraints and the objective to maximize is the expected return of the portfolio . The probability that portfolio returns fall below the VaR level is called *Shortfall Probability*:

$$SP = p(r < VaR) \tag{9}$$

where r stands for $\sum_{i=1}^{n} r_i x_i$.[6]. Furthermore, the *Expected Shortfall Probability* is defined as the expected return of portfolio given that its value has fallen below VaR:

$$ES = E(r \mid \; r < VaR) \tag{10}$$

Similarly to Variance, *VaR* lack in coherence and it does not stress the importance of portfolio–diversification in order to reduce the risk[6]. Anyway, it is nowadays one of the most used risk measures, as it is imposed by the Basel agreement, which allows banks to use their own VaR models in order to assess the credit risk[7].
Amongst other approaches it worths mentioning the Mean-Absolute-Deviation model (MAD)[50], in which the risk is defined as the mean absolute deviation of the portfolio rate of return. This model does not rely on probabilistic assumptions on returns (it is equivalent to the Markowitz model if returns are considered as normally distributed) and it is easier to handle because it does not require the covariance matrix:

$$\min E\left[\left|\sum_{i=1}^{n} r_i x_i - E\left[\sum_{i=1}^{n} r_i x_i\right]\right|\right] \tag{11}$$

Assuming $r_i = \frac{\sum_{t=1}^{T} r_{it}}{T}$, this equation can be reformulated as follows:

$$\min \frac{\sum_{t=1}^{T} |\sum_{i=1}^{n}(r_{it} - r_i)x_i|}{T} \tag{12}$$

.

Following the same ideas, in [76] risk is measured as the mean semi-absolute deviation of the rate of return below the average:

$$\frac{\sum_{t=1}^{T} |\min(0, \sum_{i=1}^{n}(r_{it} - r_i)x_i)|}{T} \tag{13}$$

This function is shown to be equivalent to MAD, as semi-deviation is equal to half of absolute deviation.

Furthermore, since the Mean-Variance formulation is non linear, efforts have been made to model the problem as a linear programming model. Amongst them, besides the above cited ones, the approach proposed by Young[90] defines the risk as the minimum return achieved by portfolio over a set of scenarios (*worst case realization*): The optimization problem, in this framework, consists in finding the portfolio whose worst case realization is maximum. A generalization of this approach is the Conditional Value at Risk (*CVaR*), obtained measuring the mean of a specified quantile of worst realization distribution[71].

Up to now we described only risk measures to be minimized. Indeed the PSP can be formalized as a maximization problem in which a safety measure must be optimized. It has been shown[55] that for each risk measure there exists a corresponding safety measure (obtained combining return and risk measures) and vice-versa, but minimizing a risk measure is not always equivalent to maximize the corresponding safety measure: Equality holds

[6]The original definitions of measures such Var and CVaR were originally based on prices rather than returns. Indeed they are also modeled with returns in some works[2, 55]. For homogeneity we decided to present them as based on returns: This could lead to different optimization results when using continuous rates of returns. It is out of the scope of this work giving details of this issue.

only if measures are independent from distribution-specific parameters (e.g. minimizing semi-variance (formula 7) is not equivalent to maximize its corresponding safety-measure, as its formulation is function of the mean of return distribution).

3.3.2 Multi-criteria objectives

In the multi-criteria variant of the PSP model, the objectives are usually the following [79][3]:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j \tag{14}$$

$$\max \sum_{i=1}^{n} r_i x_i \tag{15}$$

subject to constraints (3)(4).

Moreover, it is possible to have several functions to optimize: Subbu et al.[81], for instance, propose the following:

$$\begin{cases} max & \text{Portfolio expected return} \\ min & \text{Variance} \\ min & \text{Portfolio Value at Risk} \end{cases} \tag{16}$$

This model can also handle preferences, by introducing other three metrics: *Market-yield*, *Dollar duration weighted Market-yield* and *Transaction costs*. These metrics are used to describe and structure ordinal preferences.

The approach consisting in weighting the criteria of a multi-criteria objective function is common when the model is aimed to support decision processes. For example, in Ehrgott et al.[25], the objective is to maximize a weighted sum of five measures (annual price-performance, annual dividend, three year price-performance, S & P rating and volatility) and weights are to be defined by users in order to specify their preferences.

A different multi-objective formulation is given in Ong et al.[68]. According to existing models, they assume portfolio risk being divided in the uncertainty risk and the relation risk. The uncertainty risk measures the uncertainty on future return rates, whilst relation risk measures the trending degree of the sequence. In this framework the objective is given by

$$\begin{cases} max & \text{Portfolio expected return} \\ min & \text{Uncertainty Risk} \\ min & \text{Relation risk} \end{cases} \tag{17}$$

.

Many other objective functions and utility measures have been proposed, an overview of which can be found in [43].

A visual conceptual overview of the different kinds of objectives is depicted in Fig. 2, along with the possible choices for variable domains.

Before discussing the third attribute of the model, i.e., *constraints*, we have to note that the estimation of returns from real-world data raises statistical and practical issues that have to be taken into account when the PSP is tackled. A discussion on this topic is out of the scope of this paper and we forward the interested reader to the specific literature on the subject [67][9][8][29][30][24][85].

3.4 Constraints

Constraints can be first distinguished into two classes: *theoretical* and *practical*. The first class includes budget and return constraints, while practical constraints are motivated by actual problem requirements, such as minimum lots imposed by law.

3.5 Budget and Return Constraints

Budget and Return constraints are the most important ones, because they characterize the essential part of the problem. These constraints are included in the unconstrained Markowitz model and are used to theoretically define the feasibility of a solution:

$$\sum_{i=1}^{n} x_i = 1 \tag{18}$$

$$\sum_{i=1}^{n} r_i x_i \geq r_p \tag{19}$$

Constraint (18) means that all the capital must be invested. If an integer formulation is used, in which assets are represented by their actual value

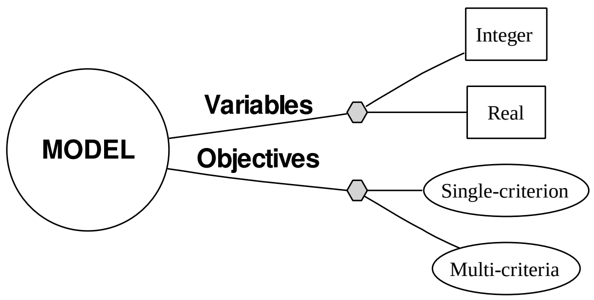

Figure 2: Conceptual representation of the PSP model attributes *variables* and *objectives*.

rather than their ratio to the whole portfolio, it can be expressed in the following way [51, 56, 76]:

$$C_0 \leq \sum_{i=1}^{n} x_i \leq C_1 \tag{20}$$

where C_0 and C_1 are respectively the lower and upper bound on the budget. Constraint(20) is less tight than constraint (18) and it is imposed when minimum lots are added to the formulation, as introducing them (in an integer formulation) makes it more difficult to find a solution w.r.t the former budget constraint (see section 3.9)[7].

Return constraint (19) is very important as returns represent one of the two main aspects of the problem.

As stated above, a shortcoming of the original Markowitz formulation is that it does not incorporate many aspects of real-world trading, such as maximum size of portfolio, minimum lots, transaction costs, preferences of which assets to include in the portfolio, management costs, etc. These aspects can be modeled by introducing constraints of the type that we have called 'practical', that are introduced in the following.

3.6 Cardinality Constraints

The number of assets in the portfolio is often either set to a given value or it is bounded. Introducing a binary variable z_i equal to 1 if asset i is in the portfolio and 0 otherwise, the constraint can be expressed as follows:

$$\sum_{i=1}^{n} z_i \leq k \tag{21}$$

This constraint is imposed to facilitate the portfolio management and to reduce its management costs. When the model contains this constraint, it can be named "The asset paring problem"[52]. Accordingly to financial and OR literature, it has been experimentally shown that, when the cardinality constraint is imposed, the ACEF tends to tightly approximate the UEF for high values of k [28][14]. The inequality form is quite common (see, for instance, [73][16][46]), however the constraint can also be expressed in the equality form [3], i.e., $\sum_{i=1}^{n} z_i = k$.

3.7 Floor and Ceiling Constraints

With these constraints we impose a minimum and maximum proportion (ε_i and δ_i respectively) allowed to be held for each asset in portfolio, so that $x_i = 0 \quad \vee \quad \varepsilon_i \leq x_i \leq \delta_i$ $(i = 1 \ldots n)$; in other words, the portion of the portfolio for a specific asset must range in a given interval:

$$\varepsilon_i z_i \leq x_i \leq \delta_i z_i \tag{22}$$

Ceiling constraints (i.e., upper bound constraints) are introduced to avoid excessive exposure to a specific asset and in some case are imposed by law. Floor constraint (i.e., lower bound) is used to avoid the cost of administrating very small portions of assets and may be implied by transaction costs (see section 3.12).

[7] In an integer formulation the budget constraint (18) must be reformulated as $\sum_{i=1}^{n} x_i = C$.

It is also possible to impose different upper and lower bounds for each asset, but this opportunity has not yet been explored in the literature.

3.8 Short sales

In the model we discussed in section (3.1) asset weights are non-negative (constraint (4)): This constraint means that no short sales are allowed and it is imposed in almost all the works we are analyzing (a notable exception is given by [72], whilst in [16] short sales are allowed in the initial formulation). Indeed, it is a common practice to sell assets that are not yet owned by the investor at the time, in expectation of a price falling: This can be formulated replacing constraint (4) with

$$x_i \in I\!R \quad \forall i \tag{23}$$

This relaxation was introduced in [10], and in this formulation it is possible to find an exact analytical solution for the PSP.

Note that, if short sales are to be forbidden, constraint (4) becomes redundant when imposing floor constraints.

3.9 Rounds

The unconstrained Markowitz model considers investments as perfectly divisible, so as to be represented by a real variable, whilst in several markets (such as the Japanese and most of European ones) securities are negotiated as multiples of minimum lots. For each asset there exists a minimum tradable lot, referred to as *round*. Rounds are usually measured in unities of money, so this constraint is encountered in the PSP integer formulation [51][45][56]. If p_j is the price of asset j and ρ_j its minimum tradable quantity, the minimum lot c_j of asset j, measured in unities of money, is given by $c_j = \rho_j p_j$. If rounds are introduced in the formulation, they becomes the integer decision variables [45, 56].

When using the continuous formulation its application consists in imposing that each weight must be multiple of a given fraction[79], and, obviously, its meaning is different from imposing *rounds* in integer formulation.

Although rounds are imposed by the Exchange Market independently of the investor size, their effects (measured in terms of deviation from the unconstrained results) seem to be relevant for small investors but negligible for big ones and their introduction has the effect of reducing the number of different assets in the optimal portfolio.

3.10 Class Constraints

In the real world of finance it may happen that investors ideally partition the assets in mutually exclusive sets (classes). Each set consists of assets with common characteristics (insurance assets, naval assets, etc.), and investors want to limit the proportion of each class. Let M be the set of classes $\Gamma_1, \ldots, \Gamma_M$, L_m and U_m the lower and upper proportion limit (respectively) for class m, the class constraint can be defined as

$$L_m \leq \sum_{i \epsilon \Gamma_m} x_i \leq U_m \qquad m = 1 \ldots M \tag{24}$$

Similar is the *Asset Class Management*, in which the universe of assets is split into subsets of assets (classes) with similar features. The best representative of each class is selected and optimization is performed on this pre-selection [42, 33].

3.11 Preassignment

An investor may wish that some specific assets be included in the portfolio, in proportion fixed or to be determined. This constraint can be imposed by setting $z_i = 1$ for the corresponding assets and imposing more or less restrictive upper and lower bounds. It has been discussed informally by Chang et al. in [14], but is not addressed in the experimental setting.

3.12 Transaction costs

Transaction costs consist of the amount of money to be paid in order to buy assets. They cannot be considered properly as constraints, as they rather represent extensions of the model allowing to take into account additional real-world features. As stated in Konno and Wijayanayake[49] the total costs follow a non-convex function on the size of the transaction: At the beginning it is concave up to a certain point (unit-transaction cost gradually decrease as size increase), then it increases linearly up to another point (unit-transaction costs are here constant) and then becomes convex due to the illiquidity premium (unit prices increases due to the shortage of supply). Thus, transaction costs can be plotted as a V-Shaped function[89].

It has been proved that ignoring transaction costs leads to inefficient portfolios[5]. Nevertheless metaheuristics lacks in including transaction costs and just a few authors considered it in their formulation [51][84][89][59][60]. This is because most works use the Mean-Variance formulation (see section 3.1) that fails in including fixed transaction costs (but it can handle proportional costs). Indeed, even if Modern Portfolio Theory states that diversified portfolio are preferable to non-diversified ones [63], there is evidence that investors choose non-diversified portfolios[12, 34, 40]. This is due to the action of transaction costs, since they are not included in the original model.

Transaction costs are instead taken into account by works exploiting other approaches, such as LP-based heuristics approaches (see section 6). In these works, exploiting an integer formulation, both fixed and proportional transaction costs are taken into account.[8]

Fixed transaction costs can also be applied if the sum of money invested in the individual asset exceeds a given threshold[45]: This is made in order to facilitate small investors exclude taxes and other fixed costs when the amount invested in an individual security is small.

Maringer [60] investigates fixed only, proportional only, proportional with lower bound and proportional plus fixed costs, using an integer formulation in which the invested amount varies from 500 up to 10000000 euros. It is shown that the higher the fixed costs, the smaller the cardinality of the portfolios and the portfolio performances: This effect is more evident for small investors. Proportional costs instead, depending exclusively on the transaction volume, cannot be avoided by substituting securities by adding shares to already included ones. Nevertheless, even in this case the higher the cost, the smaller the cardinality of the portfolios and the portfolio performances. The case of compound costs is even more interesting and has been tackled by [2] too, where the proportional cost to be paid for a given asset cannot be lower than a threshold $Pmin_i$. This means that for each asset i, assuming that pc_i represents its associated proportional cost, the investor will always buy a number of rounds (as they represents decision variables, see section 3.9) x_i such that $pc_i q_i x_i \geq Pmin_i$, where q_i represents asset price. This implicitly defines a lower bound ε for each asset:

$$\varepsilon_i = \left\lceil \frac{Pmin_i}{pc_i q_i} \right\rceil \tag{25}$$

The higher $Pmin_i$, the higher the lower bound, while the higher the proportional costs, the smaller the lower bound. This lead to the phenomenon that for a given $Pmin_i$, the portfolio will be more diversified the higher the transaction cost is. Imposing proportional plus fixed combines the effect of the two costs, as increasing them will reduce portfolio diversification.

So, considering all typologies, global transaction costs tend to reduce portfolio diversification, but this assertion must be taken *cum grano salis* as investor behavior depends on subjective factors too: In Glover et al.[31] it is explicitly stated that if the investor is risk-averse the portfolio held is more diversified with taxes and transaction costs, whilst diversification is not requested by investors with low risk-aversion if taxes an transaction costs are included in the model. It is clear however that only proportional costs are suitable to be included in the continuous model, as the remainder is sensitive to the invested amount.

3.13 Turnover and trading constraints

For the sake of completeness, we also mention a class of constraints that arise in the multi-period formulation of the problem. These constraints define upper and lower bounds, respectively in case of buying and selling, for the variation of asset values from one period to the next one. Moreover, they are usually combined with transaction costs and taxes. These constraints have been introduced by Crama and Schyns in [16] in a variant of the single-period formulation.

The complete classification of the PSP model variants that can be found in the main literature

[8] Taxes can be considered as additional proportional costs whose amount is determined w.r.t the type of trader and the kind of operation performed, see [56].

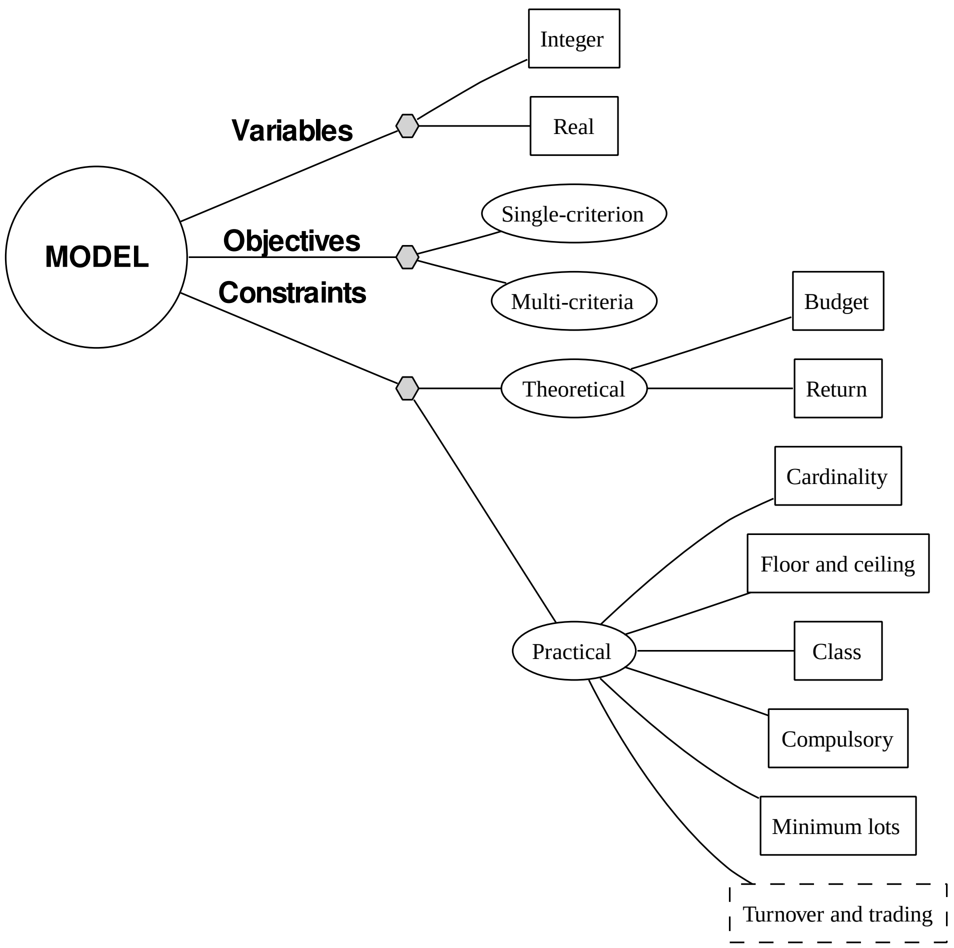

Figure 3: Conceptual representation of all the PSP model attributes (*variables*, *objectives* and *constraints*).

on the subject is depicted in Fig. 3. In the next section we present an overview of the main metaheuristic approaches for tackling the PSP.

4 Metaheuristic techniques for portfolio selection

Metaheuristics are solving strategies based on which approximate algorithms for combinatorial optimization problems can be designed and implemented. In general, metaheuristic-based algorithms can not prove the optimality of the returned solution, but they are usually very efficient in finding (near-)optimal solutions. Some techniques, such as tabu search, iterated local search, variable neighborhood search, ant colony optimization and evolutionary algorithms have proved to be very successful in tackling real-world problems. For further details on metaheuristics we forward the reader to [11] and [37]. In this section, we provide a review of the most relevant metaheuristic approaches to the PSP. To this purpose, we will adopt the standpoint provided by MAGMA, a general framework for metaheuristics [64]. MAGMA (MultiAGent Metaheuristics Architecture) provides a framework for classifying and designing metaheuristic as a multi-agent system. Metaheuristics can be seen as the result of the interaction among different kinds of agents: the basic architecture contains three levels, each hosting one or more agents. At each level there are one or more specialized agents, each implementing an algorithm. LEVEL–0 provides a feasible solution (or a set of feasible solutions) for the upper level, therefore it can be considered as the (initial) solution construction level. LEVEL–1 deals with solution improvement and agents perform a trajectory in the search space until a termination condition is verified (basic local search level). LEVEL–2 agents have a global view of the space, or, at least, their task is to guide the search toward promising regions and provide mechanisms for escaping from local optima (long term strategy level). Classical metaheuristic techniques, such as tabu search, can be easily described via these three levels. This basic three level architecture can be enhanced with the introduction of a fourth level of agents, LEVEL–3 agents, coordinating lower level agents. With this fourth level, the framework can also describe hybrid techniques such as large neighborhood search, in which complete solvers are integrated into metaheuristics [17][70]. We first survey the basic concepts metaheuristics for PSP are based upon, i.e., the various choices for defining the set of feasible solutions, the neighborhood structure(s) and the cost function. Then, we give an overview of the techniques level by level, starting from the solution construction till the most general search strategies.

4.1 Metaheuristic attributes

We can conceive a metaheuristic as an abstract class whose attributes are the *search space*, the *cost function* and the *neighborhood structure(s)* that represents the basic components of the search strategy. Once these attributes are instantiated, the search strategy can be designed by instantiating the algorithm for each of the search levels, i.e., *solution construction*, *solution improvement*, *search strategy* and *coordination strategy*.

4.1.1 The search space

Usually, a solution to the PSP is represented by an array of n variable $x_1, \ldots, x_n$, where x_i represents the fraction of the amount invested in asset i (or the actual amount of money in the integer variable model). Besides those variables, auxiliary variables and data structures can be added for improving algorithm efficiency. An important distinction has to be made in the way the different approaches deal with constraint violations. Indeed, some works define the search space explored by the algorithm as consisting of only feasible portfolios (i.e., satisfying all the constraints in the model), while in other works the search process is allowed to explore also infeasible solutions.

We therefore can classify the search processes depending on how they handle infeasibility:

- *All feasible* approach: Each candidate solution s must satisfy the constraints at any step of the search process (e.g Chang et al.[14]);
- *Repair* approach, in which if an infeasible solution is found, this is immediately forced to satisfy the constraints by means of an embedded repair mechanism (e.g Streichert et al.[79]);
- *Penalty* approach: It is allowed to visit infeasible solutions, but those will be assigned a penalty in the cost function, depending on the amount of violation (e.g Schaerf [73]).

Repair mechanisms provide a tradeoff between exploration and intensification. A basic repair approach is presented in [21, 72]: These works tackle an unconstrained formulation, so constraints likely to be violated are the budget and return constraint. In this case the only action to be performed is to normalized weights so as to sum to one, but in [21] weights are normalized at each step, whilst [72] repairs solutions only after finding five consecutive infeasible moves. This mechanism repairs asset weights in the following way:

$$x'_i = \frac{x_i}{\sum_j x_j}$$

where x_i represents the actual weight of asset i and x'_i the repaired weight.

Extended versions of this mechanism are given by [73](*idR* and idID), where the repair mechanism must satisfy the floor and ceiling constraints too. This is done by normalizing, for each asset i, values $x_i - \varepsilon_i$ rather than x_i, as previously introduced, in order to ensure that no asset can fall below the minimum allowed ε_i: The repaired asset weights (accordingly with the previously introduced notation) will be:

$$x'_i = \varepsilon_i + \frac{x_i - \varepsilon_i}{\sum_j x_j - \varepsilon_j}$$

A more complex typical repair mechanism is explained in Streichert et al.[80], referring to a formulation with cardinality and minimum lots constraints. This mechanism takes as input a non-normalized solution vector and repairs the solution through the following deterministic procedure:

1. A vector x' is generated: If cardinality constraint is imposed, it will contain only the k asset with highest weights whilst the surplus variables are set to 0; otherwise x' will be composed of all assets;

2. All weights are normalized so as to sum to one. This is done by setting weights $x''_i = x'_i / \sum_j x'_j$;

3. a further modification is required to meet minimum lots constraints: asset weights are forced to the largest roundlot level less or equal than the current asset weight, i.e., $x'''_i = x''_i - (x''_i \bmod c_i)$. The residual amount of budget is redistributed so as to meet minimum lots constraints by buying quantities of c_i of assets with the largest $(x''_i \bmod c_i)$ until all the budget is spent.

This repair mechanism can anyway fail in finding feasible solutions w.r.t return constraint. In this case, as authors exploit a genetic algorithm, the fitness of the portfolio will be assigned the worst possible value.

Indeed, these three ways of handling constraints are mostly used together, deciding, for each constraint, which is the most suitable way for handling them. For instance, budget constraint is used to norm the solution, so it is preferred having solutions strictly satisfying them, exploiting either a feasible approach[16] or a repair mechanism[46]. Also return constraint is used to norm the solution, but it can be used in both feasible or penalty approaches. The choice between these two strategies depends on the cost function used: Penalty approach is used if the cost function consists of a combination of the objective of the problem and the violation of the return constraint, as it allows moving toward an infeasible state, assigning a penalty for the violation of constraints (the main examples of this approach will be discussed in section 4.1.2); if other cost function are used, the feasible approach is preferred[16].

There is, instead, no reason for preferring one of these strategies when dealing with other constraints: For instance, at point (1) we explicitly referred to the case of cardinality constraint as being repaired by a useful mechanism, but ensuring that solutions are always feasible w.r.t. to this constraint has been exploited in [16, 46, 58, 3].

We mention that it turns out to be difficult determining which class a search method belongs to, as it can be difficult to determine if a search trajectory moves only in feasible areas because of its formulation or because an implicit repair mechanism is embedded. For this reason, the pure all-feasible approach can be hardly found in literature, and the few examples are to be reformulated as unconstrained PSP[13, 72].

4.1.2 Cost function

When the PSP is attacked by metaheuristic algorithms, it is important to distinguish between objective function and cost function. The former represents the function to be optimized to solve the problem, while the latter represents the function guiding the search process over the search space. In many metaheuristic algorithms the objective of the problem is used as evaluation function, but sometimes different cost functions can better guide the search toward promising solutions.

An example of cost function for the PSP is provided by Schaerf[73] who defines a cost function in which the cost associated to the violation of return constraint $\big(f_1(X)\big)$ is combined with the original objective function $\big(f_2(X)\big)$. The overall cost function to be minimized is a weighted sum of the two components $w_1 f_1(X) + w_2 f_2(X)$, where w_1 and w_2 vary during search according to a shifting penalties mechanism.

$$\min w_1 f_1(X) + w_2 f_2(X) \tag{26}$$

$$f_1(X) = \max\Big(0, \quad \sum_{i=1}^{n} r_i x_i \quad - r_p\Big)$$

$$f_2(X) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$$

A similar approach is followed by Gilli and Këllezi in [29]. They choose the following objective function:

$$\min\left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sigma_{ij}x_ix_j+\overline{p}\Big(r_p-\sum_{i=1}^{n}r_ix_i\Big)\right) \quad (27)$$

$$\overline{p}=\begin{cases} c & \text{if returns are higher than } r_p \\ 0 & \text{otherwise}\end{cases}$$

where $\overline{p}$ is the penalty term.

4.1.3 Neighborhood relations

The neighborhood relation defines the states of the search space that are reachable from the current state of the search. The definition of neighborhood structures to be used during search is one of the key components of metaheuristic algorithm design.

In general, neighborhood relations can be divided in two classes:

1. Neighbors are generated by modifying the weights of a subset of the assets of the current portfolio.
2. Neighbors are generated by modifying all the assets in the current portfolio.

We can ideally define a neighbor of a solution by selecting one asset to be modified, specifying the amount of variation and performing the change. This asset is referred to as *pivot*[3]. Then, this modification is counterbalanced by changing the weights of some other assets. If only a predetermined subset of assets is selected to be modified the neighborhood is said to belong to class 1, otherwise the neighborhood is said to belong to class 2.

The neighborhood structures of class 1 can either consist only of feasible solutions (e.g., [73], structure *TID*) or allowing infeasible moves too (e.g., [72]). The simplest neighborhoods in this group are generated by modifying the pivot weight and counterbalancing this change by modifying the weight of only one other asset ([72] *TID*; [29]). This structure can be generalized by introducing an integer c representing the number of assets to be modified in order to counterbalance the pivot weight variation. Crama and Schyns[16] use $c = 2$, but it is possible to set c at any number, even 0, thus allowing infeasible moves[9]. In the previously described neighborhoods, the *step* size is set before choosing the assets involved in the modification; however, neighbors can also be generated by varying this value. For example, in [3] neighbors are generated by varying *step* from a minimum of $\frac{w_{pivot}}{n}$ to a maximum of w_{pivot}, being forced to assume all multiples of $\frac{w_{pivot}}{n}$.

Neighborhoods of class 2 are generally used in population-based algorithms[21, 53, 79, 51], especially in genetic algorithms, in which crossover and mutation operators could return infeasible solutions. In this case, it is often impossible to determine which asset plays the role of *pivot*. Anyway, there are some representative cases in which the pivot is used, such as in [14][13][73]. The first examples of neighborhood relations in local search for the PSP were introduced by Rolland[72]. These neighborhoods are defined for the unconstrained model, i.e., the one with only theoretical constraints as explained in Sec. 3, and can be considered the basic structures upon which further developments have been designed. In the first structure (referred to as *RollandI*) the neighbor of a solution is defined as a solution in which the weight of only one asset is increased or decreased of a given quantity, called *step*. The second neighborhood (referred to as *RollandII*) is defined so that the weight of an asset is increased or decreased of a given *step* and the value of one other asset is respectively decreased or increased of the same value.

With these two neighborhood structures, the assets contained in the final solution are a subset of the starting portfolio, since the assets to be modified are chosen amongst the ones present in the portfolio. Anyway, this does not prevent the search from being able to explore all the possible asset combinations, because the model is unconstrained and the portfolio is initialized with $x_i = \frac{1}{n}$, for each asset $i = 1, 2, \ldots, n$. We also observe that *RollandI* might move the search to infeasible solutions. These neighborhoods are well suited for the unconstrained model, but have to be modified for the constrained models because assets can not be present in the portfolio in any quantity. Hence, these neighborhood structures are modified by embedding asset insertion and deletion operations.

[9] In this case a repair mechanism should be included.

RollandII can be modified by transferring a quantity from one assets i to another asset j even if the latter does not belong to the portfolio. In this case, asset j will be inserted in the portfolio (see the neighborhood called *TID* in [73] and [29]). This approach should also include some mechanism to handle upper and lower bounds, in case they are present in the formulation. *RollandI* can be modified by enforcing the satisfaction of the budget constraint and by allowing insertions and deletions of assets. Feasibility w.r.t. the budget constraint can be enforced by increasing the weight of one asset and decreasing the other asset weights [13]. More precisely, if a solution is given by a weight vector $(x_1 \dots x_n)$, the neighboring one is $(\frac{x_1}{1+step}, \dots \frac{x_i+step}{1+step} \dots \frac{x_n}{1+step})$, for only one i, $1 \leq i \leq n$. This neighborhood is proven to be complete, i.e., for a long enough sequence of moves, each solution can in principle be reached. Completeness does not depend on the initial solution and holds iff $step \leq \frac{1}{n-1}$. The possibility of having asset insertions and deletions lead to neighborhoods defined in [73], called *idR*, and [14]. This neighborhood takes into account the case that an asset i is decreased so that its value falls below its lower bound ϵ; hence, asset i is deleted and another asset j is inserted in the portfolio. Conversely, if asset i is increased so that its value exceeds its upper bound δ, then its weight is set to δ and all other asset weights are normalized. Observe that all these variants do not change the number of assets in the portfolio. A further improvement is thus possible by allowing neighbor solutions to have different number of assets (see [73],*idID*), defined by allowing three kinds of operations on the selected asset i:

- If asset i is already in the portfolio, increase its weight of a given quantity. If the resulting value exceeds the upper bound δ, then set the value to δ.

- If asset i is already in the portfolio, decrease its value of a given quantity. If the value falls below the lower bound ϵ, asset i is deleted and *not* replaced by any asset.

- If asset i is not in the portfolio, it is inserted in the portfolio with weight equal to its lower bound.

After these operations, asset weights are normalized.

4.2 Metaheuristic search components

In this section, we describe the search methods composing the metaheuristics for the PSP. We first present trajectory based strategies, such as simulated annealing and tabu search, and then we introduce population-based metaheuristics, such as evolutionary algorithms and ant colony optimization.

4.2.1 Initial solution

It has been empirically observed that metaheuristics for the PSP are usually quite robust with respect to the choice of the initial solution. This assertion has been formally proven by Catanas in [13], subject to the specific neighborhood structure defined therein. For this reason, most of the works assume as starting solution a randomly generated one or a solution constructed by means of a simple heuristic procedure [25], possibly embedding also a mechanism to ensure feasibility of the initial portfolio [16].

4.2.2 Iterative Improvement

Iterative improvement can be considered as the simplest local search, as it performs a path in the search space by moving from a solution to a neighboring one with a lower cost. This search can be named *best improvement*, if the neighbor chosen is the best among the feasible neighbors, or *first improvement*, if the chosen neighbor is the first state found during the neighborhood enumeration that is better than the current one. Iterative improvement is usually incorporated into a more complex strategy, rather than being used as a stand alone local search. For instance, in Glover et al. [31] iterative improvement is the local search component of a variable neighborhood search technique. As another example, we mention Armañanzas and Lozano [3] who use a greedy search to refine solutions found by an ant algorithm.

4.2.3 Simulated Annealing

The possibility of moving to solutions with a higher cost (i.e., performing degrading moves) characterizes Simulated annealing (SA). The probability of moving toward solutions worst than the current one depends on the cost difference between the two solutions and it also decreases during the search. This probabilistic acceptance criterion enables the search to escape from local optima. Crama and Schyns [16] apply SA to various PSP models by first considering in the model only one constraint class at a time (floor, ceiling and turnover first, then trading and cardinality), then they include all constraints in the model. The authors experiment with three strategies:

- Independent runs, starting from the same initial solution;
- Subsequent runs, using as initial solution for the current run the best one found in the previous one;
- Run the algorithm a number of times such that a list P of promising solutions is created, then perform $|P|$ independent runs, using as initial solutions the ones stored in P.

The fact that there is no clear dominance among these strategies gives support to the statement that such a search processes are insensitive to the initial solution. SA by Crama and Schyns is able to plot the UEF exactly and achieves good performances in the model with floor, ceiling and turnover constraints. Nevertheless, the ACEF returned in the model with trading constraints appears to be quite rugged. Anyway, in all the cases this technique is able to approximate the CEF in reasonable runtimes for medium-sized instances.

The concepts of SA can also be effectively utilized inside population-based algorithms, as done in [46] [32]. In the approach proposed in [46], an initial population of random portfolios is generated. Then, for each portfolio p_o in the initial population, a new portfolio p_n is created by selecting some assets i and modifying them according to the following rule:

$$w_{x_{in}} = \max(w_{x_{io}} + s, 0) \tag{28}$$

where s is randomly chosen in the range $[-U_t, U_t]$ and this range decreases over time. Weights are then normalized and p_n is evaluated and accepted or not depending on the Metropolis criterion. Once the new population is created, it is further refined by replacing worst portfolios either by a clone of a probabilistically selected portfolio with higher fitness with probability r or, with probability $1-r$, by a portfolio composed of assets with average weights over best portfolios.

An interesting application of SA for a multi-objective formulation is presented in [3], in which moves are always accepted if at least one criterion is improved, while deteriorating moves are subject to the SA usual probabilistic acceptance criterion. This approach, applied in a formulation with floor, ceiling and cardinality constraints, seems to find good solutions in the lower part of the frontier, where risk and profits are small.

4.2.4 Threshold Accepting

Threshold accepting (TA) shares some analogies with SA, as a degrading move can be accepted if the cost difference between the current and the new solution is within a given threshold, that is progressively decreased to zero. The threshold decreasing schedule is defined by estimating the distribution of distances between objective values of neighboring positions (an analogous parameter tuning procedure is undertaken also for SA). TA has been applied to the PSP by Dueck and Winker [24] and Gilli et al.[29] [30]. These works are primarily aimed at comparing risk measures, so the algorithm represents the technical mean to investigate financial aspects. For example, in [24] different risk measures are compared. Experimental results show that the solutions corresponding to a risk measure are generally not efficient w.r.t. another risk measure. In this way it is possible to directly compare risk measures. In [24] it is stated that the ACEF is not smooth, since it turns out to be composed of linear fragments, and the curve switches from a segment to another one when the fraction held in a particular asset changes sharply.

Besides solving a floor and ceiling constrained mean-variance PSP, Gilli and Këllezi [29] tackle a more realistic problem in a downside-risk framework in which decision variables are integers. The problem is formulated as a maximization of future returns, while value-at-risk and expected-shortfall are compared as risk measures constraining the shortfall probability for a given level of ES and Var. In a further work [30] TA is used to compare three different risk measures: *Value at Risk (VaR)*,*Expected Shortfall (ES)* and Omega measure (defined as the ratio of the weighted conditional expectation of losses over the weighted conditional expectation of gains) in a formulation with cardinality and upper/lower bounds constraints. Results show that Mean-VaR portfolios are more diversified than those obtained with ES criteria, while ES frontier dominates the other two.

These works stress the fact that much attention has to be paid to the choice of an appropriate risk measure. Indeed, efficient portfolios with respect to a risk measure are usually not efficient with respect to other measures and efficient portfolios are very different from each other with respect to different utility functions.

4.2.5 Tabu search

The Tabu search metaheuristic (TS) moves away from local optima by forbidding the search to execute the inverse of the last l recently performed moves. This simple mechanism, enhanced with the exploitation of the search history for intensifying and diversifying the search, makes TS one of the best performing local search strategies. The application of TS to the PSP has its milestones in the works by Rolland [72] and Glover et al.[31]. These works refer to different formulations of the problem and moreover Glover tackles a multi-period formulation. Nevertheless, both deserve to be analyzed for the richness of concepts presented.

Rolland uses a TS for the unconstrained problem. The author tackles two problems of minimizing variance and minimizing variance given an expected level of returns. That work is more oriented in finding a single point (describing the trajectory followed by the algorithm over time to reach it) rather than drawing out the whole UEF.

The approaches designed for tackling these two problem formulations differ in the repair mechanism. In the *minimum variance* formulation, after having executed five steps in the infeasible search space area, the algorithm repairs the incumbent solution as follows:

- If the investment exceeds the budget (i.e., if $\sum_i x_i > 1$), find the asset i with maximum sum of covariance referring to other assets (i such that $\sum_j \sigma_{ij} x_i x_j$ is maximal) and decrease x_i in order to ensure feasibility;
- If the investment is less than the budget (i.e., if $\sum_i x_i < 1$), find the asset i with minimum sum of covariance referring to other assets (i such that $\sum_j \sigma_{ij} x_i x_j$ is minimal) and increase x_i in order to ensure feasibility.

The algorithm for the *minimum-variance-given-return* formulation initially tries to reach the desired level of returns, repairing the solution as follows (after having visited consecutively five infeasible solutions):

- find i such that

$$\left| \left(\left| 1 - \sum_j x_j \right| \cdot r_i \right) - \left| \left(\sum_j x_j r_j - r_p \right) \right| \right| \quad (29)$$

  is minimized;

- If the investment exceeds the budget (i.e., if $\sum_i x_i > 1$), decrease x_i in order to make the solution feasible.
- If the total investment is less than budget (i.e., if $\sum x_i < 1$), increase x_i so as to make the solution feasible.

When the return level of the best solution found is within the $0,005\%$ of the desired level, the repair mechanism invoked is the one described for the minimum variance problem, so that the solution is feasible w.r.t. the requested minimum-variance point after the requested return level has been reached.

Even if the proposed TS is said to attain good performances, it is useful only to find single point instead of the whole UEF, therefore this implementation does not represent the most powerful solution for real-world problems; however, it can be useful when only one desired level of return is given.

Glover et al. tackle the asset-allocation with fixed-mix, a problem similar to the PSP. This is a multi-period problem in which we want, for each period, to respect the proportions of asset classes (in this case assets, bonds and treasury bills) out of the whole portfolio, in order to attain the same risk profile for each period, taking into account cash-flows generated by the portfolio management. At the beginning of each period, the portfolio must be re-balanced in order to ensure feasibility, as assets generate dividends to be re-invested, transaction costs must be taken into account and constraints on proportions held can be considered. The simplest strategy is given by selling a portion of asset classes with returns higher than the average return and buying a portion of asset classes with returns below average.

Both cases with and without transaction costs are investigated and the search strategy is implemented by interleaving TS with *variable scaling*. With this term we indicate a strategy in which the neighborhood changes over iterations due to a change of the *step* length of moves (the biggest *step* length is 5% and the smallest is 1%). *Step* lengths are defined and ranked in decreasing order, and an Iterative improvement search is performed with the first *step* length. When no improvements are obtained, the *step* length changes to the next value and the Iterative improvement procedure is repeated starting from the last solution found. This process is iteratively repeated until the last step value of the list is reached. At this point, if improvements were reached over the list, the process restarts from the first value, otherwise the procedure stops. At the end of this phase, a TS run is performed; in case of improvements, the search switches back to variable scaling and the process continues until no improvements are achieved. The *step* size is crucial for the effectiveness of the algorithm and in TS it is set at a higher value than in Variable Scaling so as to diversify the search. The ACEF is compared with the frontier obtained with exact global optimization, and it is shown that they are almost identical, in both the cases with or without transaction costs.

Tabu Search has been widely applied to solve the PSP. It is easy to find it in works aimed at comparing the performance of different algorithms on the same instance (see Sec. 5). A very successfull application of TS can be found in Schaerf[73], in which TS is improved by dynamically changing the neighborhood structures.

4.2.6 Variable Neighborhood Search

Variable Neighborhood Search[35] (VNS) is a metaheuristic that dynamically changes neighborhood structures during search, so that a neighborhood is substituted by another one when the current solution cannot be further improved. There is no explicit application of VNS to the PSP, however, as this strategy is very general, its principles can be found in some important works in the literature. This is the case of the work by Glover et al.[31], in which the implementation of *variable scaling* can be considered a VNS, as a new neighborhood is introduced by changing the *step* when no further improvement is possible. A similar technique can be found in [25], in which the search switches between two neighborhoods. Moreover, similar ideas can be found in [73], in which TS is implemented in a token ring sequence, in which runs using a different neighborhood structures are interleaved.

It is worth mentioning also the work by Speranza[76], in which a heuristic algorithm is defined and applied to Milan Stock Market using an integer formulation enriched by introducing proportional transaction costs (floor, ceiling and cardinality constraints are discussed but not addressed in the computational analysis). Here, in order to satisfy the constraints on capital, assets are ordered and re–numbered in nondecreasing order of x_i in the portfolio; then x_1 is increased (and, if this move is unsuccessful, decreased) by one unit. If the new solution is feasible, the algorithm stops, otherwise the procedure is repeated over $x_2 \dots x_n$. At the end of this phase, if no feasible solution is found, the cycle is repeated increasing assets by two units, then three and so on. This mechanism can be considered as a kind of VNS, even if the neighborhood cardinality is constant over the whole process and the neighbor selection process is deterministic.

4.2.7 Evolutionary Algorithms

Evolutionary algorithms (EA) are population-based metaheuristics whose inspiring principle is the Darwin theory of natural evolution and selection. These search strategies maintain and manipulate a set of solutions at each iteration, combining the best solutions of the current set to generate the solutions of the new set. Often EA-based metaheuristics are enhanced by hybridizing EAs with advanced constructive procedures and local search strategies. The strategies presented in these works can be better labelled as *memetic algorithms*, as local search runs are executed to improve the quality of the solutions constructed by the EA.

The first applications of EAs to the unconstrained PSP are presented by Tettamanzi et al. in [4][53][54]. In [4] a genetic algorithm (GA) is implemented for the PSP with down-side measure of risk. In the algorithm, one population is handled and individuals are generated according to investor preferences: a specie is defined for each λ (where λ is the trade-off coefficient between return and risk, as discussed in Sec. 3.3). Individuals are generated such as their probability of belonging to a specie is proportional to the investor's interest in that specie. At each generation, a new individual replaces the worst one in the previous population.

In a further work [54], a distributed genetic algorithm is applied in which each λ value is associated to a subpopulation. As the AUEF (the unconstrained PSP is tackled) is composed by plotting a point for each λ, the greater the number of populations, the finer the resolution of the frontier. Migrations of individuals between populations corresponding to neighboring values of λ are permitted, in order to avoid premature convergence of the algorithm. Individuals are allowed to mate only with individuals of the same population or of adjacent ones. This implementation outperforms the previous sequential version, and in [53] a detailed description of the implementation and risk measures is provided. In parallel implementation of GA, if the cardinality constraint is imposed it is possible to search in parallel several ACEF corresponding to each value of k, using information from each of these to improve the search process of others. With this approach, the ACEF approximates the UEF with increasing precision, as k increases and constrained optimal portfolios are shown to be not significantly different from unconstrained ones, except for very small number of assets and very low risk levels.

Liu and Stefek [52] tackle the PSP with cardinality and ceilings constraints, comparing GA with a heuristic proprietary method and they investigate crossover rates, population size and elitist strategy showing that GA can achieve good performances, even if worse than the heuristic, especially concerning execution time.

Memetic algorithms for the PSP are presented in [61], in which the use of SA and TA inside the EA framework is compared to tackle the unconstrained PSP. The results discussed indicates that TA is more suitable when *VaR* is used as risk measure, while SA makes the algorithm perform better when *ES* is chosen. An explanation of this result is given by observing that *VaR* induces a rugged search space, while *ES* induces a smoother landscape. In that work also the use of a kind *elitist* strategy is investigated, that implement a sort of intensification of the search around the best found solutions. This strategy improves the performance of the algorithm when the search space is smooth, while doesn't payoff when the search space is rugged, as it reinforce the local optimum we want escape from. Moreover, the introduction of this kind of intensification makes the algorithms more robust against parameter values.

The previously discussed works (together with [85, 25, 89, 14]) tackle the inherently multi-criterion PSP using single-objective formulations and techniques (see section (3.3.1)). Indeed GA, by being inherently effective in diversifying the search, show good performances especially in multi-objective formulations of PSP, as shown by the applications of MOEAs (MultiObjective Evolutionary Algorithms) [78][79][80][68]: For instance, *NSGA II* [18] represents one of most powerful multi–objective metaheuristics and has been applied to PSP in [51, 21]. Furthermore it has been argued that handling PSP in single-objective fashion make strategies less flexible to decision makers preferences[81].

In this multi-objective framework, PSP is tackled by Streichert et al. in [78][79][80], using a two-objectives optimization model, enriching their implementation by adding an archive in order to store the frontier obtained so far. In their work, they introduce the knapsack representation of portfolios, comparing it with the *standard* one. The authors also investigate the use of *Lamarckism*. In fact, these algorithms embed a repair mechanism that prevents the search from rejecting infeasible solutions. In the GA version without Lamarckism, only the phenotype of an individual (i.e., the normalized vector of assets) is altered by the repair mechanism, while the genotype (i.e., the non-normalized vector of assets) remains unaltered. Conversely, in the version with Lamarckism, the repair mechanism modifies the genotype too, according to the phenotype. In each case, this solution representation leads to a better performance than the standard one. Moreover, Lamarckism helps improve performances too. Furthermore, different variable representations (binary and real-valued) are also compared and different coding [79] and crossover operators [80] are examined.

We should observe that the model with floor, ceiling and cardinality constraints is the most commonly used in literature when GA are applied [14][25][26]. GA have also been used in conjunction with formulations differing from the canonical Mean-Variance one, in order to define more realistic customer-oriented frameworks. An interesting example is represented by [89], in which the objective function to maximize is given by the usual weighted objective function (eq. 6), but they solve this model for different isolated values of λ rather than trying to plot the whole frontier. They show that in the obtained portfolios return is higher than the best one provided by optimization software for Mean-Variance (LINGO[44]) even if they are more risky.

One of the main contribution of that work is that the expected return is considered as a variable, rather than an instance data. The return ranges in an interval in which arithmetical mean represents lower bound a if its recent history trend has been increasing, the upper bound b if its trend has been decreasing. No additional constraints are added to the formulation. V-Shaped transaction costs are also investigated for portfolio revision, but they are only considered as proportional[10]. Transaction costs (embedded in a MAD objective function) and single λ values analysis are considered in Wang et al.[85] in which a sample procedure for stochastic returns is introduced instead of the classical scenario analysis.

More complex approaches are proposed aimed at helping decision making by introducing other measures either to define an ordinal-preference framework in which other measures are added to the formulation (see [81][25]), or to predict the future return rate and to estimate the uncertainty risk of the future return rate when the sample is small [68].

[10] In a further work[88] risk-free asset are introduced and the formulation is based on a linear programming model.

4.2.8 Particle Swarm

The nature-inspired paradigm referred to as *Particle swarm* is a promising search paradigm, especially when continuous optimization problems are tackled. Nevertheless, its application to the PSP is still limited, and the works on this topic do not tackle the standard formulation, being aimed at finding one portfolio optimal with respect to a measure such as the reward-to-variability ratio out of a given set of assets, rather than drawing out the whole efficient frontier[47][66].

4.2.9 Ant Colony Optimization

Ant colony optimization (ACO) is a population-based metaheuristic that is inspired by the foraging behavior of ants. Solutions are built component by component, according to a probabilistic procedure that bias the choice of the next solution component on the basis of the quality of the previous constructed solutions. Usually, ACO also incorporates some local search algorithm to improve the quality of the solutions built. Initially conceived for discrete spaces, ACO has been adapted also for continuous spaces, too (see, for instance, [75]). Nevertheless, the potential of ACO for tackling the PSP appears still not completely exploited.

A successful application of ACO can be found in a PSP modeled with the cardinality constraint [3][58]. The approach consists in defining a population of n ants that explore a completely connected graph composed of n nodes. Assets and nodes are in one-to-one mapping and the path traversed by an ant corresponds to the assets to be chosen for the portfolio. Path lengths are of exactly k steps, where k is the portfolio cardinality. In the case of multi-criteria optimization, ants are divided in populations such that each population solves a problem corresponding to one objective function [3]. When ants terminate the exploration phase, a greedy search refines the solutions. This method finds better solutions than SA and iterative improvement and results are particularly striking in the upper part of the frontier, where risk and profits are high.

ACO has found application in problems similar to the PSP such as the so-called multi-objective project portfolio selection [22][23], a generalization of the bin-packing problem in which we want to choose a portfolio of project proposals (e.g. research and development projects) constraining the problem so as to ensure that the portfolio will contain not more than a given maximum number of projects out of a certain subset (e.g. projects pursuing the same goal) and imposing resource limitations and minimum benefit requirements.

5 Comparative studies

The comparison of the techniques for tackling the PSP described in the literature is an awkward task, primarily because data-sets are rarely the same, different algorithm implementations can lead to unfair comparisons, utility and performance measures are often different. Furthermore, comparisons can be driven by different criteria, such as efficiency, robustness, performance with respect to a given model, etc. For these reasons, the comparison amongst different works is not possible and we have to resort to papers describing and comparing different algorithms on the same instance set and model. Before overviewing the most relevant works on this subject, we briefly comment on the performance measures used for the comparison of the algorithms.

Performance measures are usually obtained by comparing constrained results (ACEF) with the ones obtained in the unconstrained case for each level of return (each point of the UEF) and computing statistical measures (mean and median percentage error, standard deviation etc.) for the overall frontier. There are however many ways to define an error measure. For instance, Chang et al.[14] consider the distance of the point from the UEF, defined as the minimum between the distance on the x-axis direction and the distance on the y-axis direction[11]. Another measure can be found in Streichert et al.[78] [79] [80], where the algorithm performance is computed as the percent difference between the area below the UEF and the obtained ACEF. The issue of comparing two frontiers is just an instance of the more general problem of comparing algorithms for multi-objective optimization[69]. Often, also statistical tests are used [28] [21], especially to determine if the difference between UEF and CEF is significant, and some works introduce measures to determine the best portfolio in a frontier[24].

One of the first comparative works is due to Catanas[13]. That paper is focused on investigating properties of the proposed neighborhoods (see Sec. 4.1.3). The author uses TS and SA, implemented in both *robust* and *dynamic* way to tackle the unconstrained PSP. In the *robust* implementation, the *step* is kept fixed during all iterations, while in the *dynamic* one it is decreased to zero during the execution. Furthermore, a schema for the variation of the *step* is defined such that its value is increased if solution quality worsens and decreased if solution quality improves. Moreover, a threshold on the minimum value of *step* is introduced, since too small values can make the search stagnate.

Chang et al.[14] introduce cardinality and floor and ceiling constraints and observe that the CEF becomes discontinuous. This is due to the fact that feasible proportions of assets are dominated (because of the existence of portfolios with lower variance and higher return); furthermore portions of frontier could not be reachable for a classical λ-weighting drawing approach (due to minimum proportion constraints). In [14], the authors implement GA, TS and SA to solve the problem. Results show that GA is able to approximate the UEF with the lowest average mean percentage error. Regarding the constrained problem, GA seems to perform better than SA and TS, but differences are not as clear as in the unconstrained case, so they use portfolios from the three metaheuristics to draw out the ACEF. Their approach is to store, for each heuristic, all the improving solutions found in the search process and, finally, deleting the dominated ones. The sets obtained by the three heuristics are then pooled to draw the ACEF. This approach shows that for the constrained problem the ACEF approximate the UEF when the asset cardinality is high (as already stated in [28], see Sec. 4.2.7).

Jobst et al. [41] compare the results presented in [14] against two heuristic methods. The first is an *integer-restart* procedure that plots the CEF starting from the highest return and its corresponding risk to lower return and reduced risk. The result obtained at each stage is supplied as starting point to the next (lower return) stage, considering it as first feasible value (this heuristic is referred to as *warm restart* heuristic). The second, inspired to an idea similar to [76], first solves a con-

[11] A similar approach is proposed by Fernandez and Gomez [26]

tinuous relaxation without any constraints, then uses the k assets with highest weights as input for a problem in which constraints are imposed (this heuristic is referred to as *re-optimization heuristic*). Both heuristics are embedded in a branch-and-bound and are said to outperform metaheuristics used in [14]. Anyway, we should note that re-optimization heuristic could not able to draw the whole frontier when the continuous relaxation produces a portfolio with less than k assets.

Another important work that compares different techniques is the one by Schaerf[73], in which the model includes floor, ceiling and cardinality constraints. The author defines three neighborhood relations, that specify moves that satisfy the budget constraint, and defines a cost function that account for the violation of the other constraints. The initial state is selected as the best amongst 100 randomly generated portfolios with k assets. A first phase of experiments with *Best* and *First* Iterative improvement, SA and TS run as single solvers is performed. Then TS, the most promising solver in the preliminary experimental analysis, is chosen for an extensive experimental analysis, combining neighborhood relations in various token-ring strategies. In this case, the *step* length is set to a higher value in the first used solvers to favor diversification, while it is set to a smaller value in the last used solvers for intensifying the search. Experimental results show that the best performances are achieved by token ring solvers with random steps, even if fixed steps seem to behave well too. Single solvers do not attain comparably good results.

Armañanzas and Lozano [3] compare iterative improvement, SA and ACO in a multi-objective formulation with cardinality, floor and ceiling constraints. The algorithms used are tailored to the multi-objective problem, and ACO outperforms the other techniques. The simple greedy search (iterative improvement) shows poor performances if used alone, but turns out to be effective when used to refine solutions provided by ACO. Interestingly, ACO and SA best performances are found in different areas of the frontier: the first in the upper part of the frontier, the latter in the lower part.

Also Ehrgott et al.[25] proposes a multi-objective framework with cardinality, floor and ceiling constrains in which utility functions are interpolated over utility values for a set of points. They use SA, TS, GA and a local search similar to a VNS embedding a random escaping mechanism to avoid stagnation at local minima. They test the algorithms over both random and real-world instances. Results on both instance classes show that GA appears to be the best performing solver. The local search and SA achieve good results, while TS performances appear to be the worst ones.

A further interesting comparison is made by Fernandez and Gomez[26], in which metaheuristics by Chang et al. are compared with a neural net approach. An Hopfield network[12] is used to plot the ACEF when cardinality constraint and bounds (lower and upper bounds) are imposed. Their results show that there is no significative difference between their neural network and metaheuristics such as GA, TS and SA. In order to improve the performance, portfolios from the four approaches are pooled and dominated solutions are deleted, so as to obtain an improved ACEF (the same approach pursued by Chang et al.). The quality of solutions returned is high, making this neural nets approach successful[13]. Nevertheless, the number of different portfolios returned by the neural net is lower than the number returned by other heuristics, therefore, even if the quality is high, stand-alone neural nets approaches are not suitable for solving the problem in the whole frontier.

6 Related work

For the sake of completeness, in this section we briefly review heuristic approaches based on linear programming, that can be very useful as components of more robust and complex metaheuristic strategies. These works are also important because they provide experimental results for the *mean semi-absolute deviation* (see section (3.3.1)) and they deal with integer formulations of the PSP, in which assets are assigned integer values corresponding to the actual amount of money to be invested in each asset and variables are formalized

[12]Hopfield networks[38] are neural network composed of a single layer of neurons fully connected and are widely applied in combinatorial optimization[74].

[13]Indeed, neural nets can capture non linear relations among variables and do not need model assumptions, therefore they are suited for forecasting future returns without relying on the stock returns normal distribution assumption. This idea has been also exploited in [77] and [91] in order to optimize portfolio management.

as the number of rounds to be purchased for each asset.
Speranza[76] models the problem by including transaction costs, minimum lots, cardinality, floor and ceiling constraints and by introducing two auxiliary binary variables to indicate whether a security has fixed transaction costs and whether it belongs to the portfolio. The idea presented is to relax the integer constraint on quantities, transforming the problem into a linear programming one (to be solved efficiently even when the number of securities is high) and finding a solution to it. Fractional asset weights are then rounded to the closest integer and heuristics are applied to force the solution to satisfy capital and rate of return requirements. If the algorithm terminates without solutions, less restrictive bounds on capital are iteratively set. This algorithm is tested on small instances and does not guarantee to find a feasible solution (it has been further tested in [56] and did not compare very favourably against competitor solvers); nevertheless it provide good performance when the total number of assets is low and reaches a solution close to the optimal one when the invested capital is large.

In Mansini and Speranza[56], the formulation of the problem includes minimum lots and proportional taxes. The authors provide three heuristic algorithms based upon the idea of solving subproblems of the original formulation, involving subsets of initial universe of assets: These subsets are composed of assets chosen exploiting the information obtained by solving the continuous relaxation, i.e., reduced costs. In the first heuristic (referred to as *Basic-MILP-based–heuristic*) they solve the continuous relaxation of the problem. Then, they use this solution to feed the mixed integer-linear programming solver. The second heuristic (referred to as *Reduced–cost–MILP–heuristic*) considers a vector x_R with a number of assets greater than the vector of assets i s.t. $x_i \neq 0$ as input of MILP-procedure, thus including also assets whose quantity in the solution of the relaxed problem is zero. The third method consists in an iterated routine: after solving the relaxed problem, the vector x_R is used as input for a MILP procedure. After each step, half of the assets i s.t. $x_i = 0$ is deleted and half is replaced in the solution. The process ends when a given number of securities has been considered. This third heuristic is the most effective, but requires more computational time. These heuristics performs reasonably better than simple problem specific heuristics proposed in [76] and they have the advantage of being more general and are also used in Kellerer et al.[45] in a formulation enriched by introducing fixed transaction costs and minimum lots. These heuristics are applied to four different models that include rounds and fixed costs applied if the amount of money invested in a security exceeds a minimum threshold.

Even if adding constraints make it intractable to solve real-world instances of the PSP with proof of optimality [see, e.g., 41], also exact method have been proposed for the constrained PSP: Konno and Wijayanayake[48, 49] use a piecewise linear convex underestimation strategy inside a branch and bound to model a continuous PSP with concave transaction costs, imposing ceiling constraints, no short sales, no cardinality and no floor constraint. The same approach is then modified to handle minimum lots, rounding off the solution found[14]. Mansini and Speranza [57] consider the semi-MAD related safety measure (downside underachievement) to be maximized, using an integer formulation with rounds, ceiling constraint, fixed and proportional transaction costs and forbidden short sales. They divide the problem into two subproblems, solving the first and considering the solution as lower bound for the second subproblem. Experiments show that the first procedure alone can be effectively used as heuristic, indeed the authors show that in the instances considered this procedure can find an optimal solution, therefore it is very likely that it can achieve a very good performance in general. Nevertheless, like [48, 49] this work is aimed at finding single points over the frontier instead of the whole frontier, so a comparison with metaheuristic techniques is not possible.

Exact methods have been also employed as components of hybrid metaheuristics for the constrained PSP. For example, quadratic programming (QP) has been used in [19, 65] in a problem decomposition in which a metaheuristic searches in the space of assets only (i.e., on the binary variables z_i) and at each step the QP solver determines the optimal allocation over them (cardinality, floor and ceiling constraints are introduced). The difference

[14] Also portfolio rebalancing is considered.

between these two works is the metaheuristic strategy they use: [19] use Iterative Improvement (first and best) and Tabu Search, whilst [65] use a genetic algorithms. The promising results obtained by these works indicate that hybridization is able to improve performances in both terms of time and solution quality.

7 Conclusions

In this work, we have defined a framework for classifying metaheuristic approaches for the PSP, introducing the main aspects of the models of the problem and the general components of the metaheuristics developed to tackle it.

This overview enlightens the potential of metaheuristics for portfolio selection problems and it may also enable us to delineate some guidelines for the design of metaheuristic solvers for the PSP. First of all, from the analysis of the literature it clearly emerges that a best method whatsoever does not exists.[15] A second important observation is that the development of a successful metaheuristic solver is the result of a skilled combination of factors, among which a careful choice of algorithm as a function of the model, the choice of a suitable programming language and a development phase in which the different design choices are systematically evaluated. To address this issue, we envision a massive use of programming languages such as Comet [36] and frameworks such as EasyLocal++ [20]. Indeed, these tools enable the designers to test different search strategies and variants on the same problem model. Moreover, the concentration on a sole heuristic is often not sufficient to achieve results meeting practical requirements. Hence, the need for the design of hybrid metaheuristics and hybrid solvers in general, nowadays state-of-the-art solvers for many real-world applications.[16]

The PSP is only a representant of a class of problems consisting in the management of portfolios of different nature. There is plenty of scope for applying metaheuristic techniques to this classes of problems, as to date they appear to be not investigated enough. Indeed, metaheuristics provide flexible and powerful solving strategies that can effectively and efficiently tackle the various instantiations of the PSP, from the basic Markowitz formulation, to more elaborated models including also side constraints. Moreover, we believe that metaheuristic and hybrid approaches could be very successful also to tackle dynamic and multi-period formulations of portfolio selection, in which issues of *rebalancing*, *index-tracking* and *re-optimization* arise.

The works we discussed in this paper show, on the one hand, the potential of such a solving strategies and, on the other hand, the modelling and algorithm design issues that have to be addressed for implementing effective tools. Future research is now focusing on the development of methodologies for designing and implementing constraint-based metaheuristics and hybrid techniques. Furthermore, the practical importance of stochastic optimization has contributed to increase the efforts in providing effective solvers for such a kind of problems, both off-line and on-line. Finally, it is important to recognize that research on the PSP is inherently interdisciplinary and, for these problems to be effectively attacked, it requires a cross-fertilization between algorithmics and finance.

[15] This empirical observation is also supported by the so-called *No free lunch theorems* [87].

[16] Advances on hybrid metaheuristics can be found in the proceedings of HM – The international workshop on hybrid metaheuristics [1].

Acknowledgments

We thank Andrea Schaerf and the anonymous referees for very useful comments and suggestions.

References

[1] F. Almeida, M.J. Blesa Aguilera, C. Blum, J.M. Moreno Vega, M. Pérez Pérez, A. Roli, and M. Sampels, editors. *Hybrid metaheuristics*, number 4030 in LNCS. Springer, 2006.

[2] E. Angelelli, R. Mansini, and M.G. Speranza. A comparison of MAD and CVaR models with real features. Technical Report 238, Department of Quantitative Methods, University of Brescia, 2004.

[3] R. Armañanzas and J.A. Lozano. A multiobjective approach to the portfolio optimization

problem. In *The 2005 IEEE Congress on Evolutionary Computation, CEC 2005*. Springer Verlag, 2005.

[4] S. Arnone, A. Loraschi, and A. Tettamanzi. A genetic approach to portfolio selection. *Neural Network World, International Journal on Neural and Mass-Parallel Computing and Information Systems*, 3(6):597–604, 1993.

[5] R.D. Arnott and W.H. Wagner. The measurement and control of trading costs. *Financial Analysts Journal*, 46:73–80, 1990.

[6] P. Artzner, F. Delbaen, J.M Eber, and D. Heath. Coherent measures of risk. *Mathematical Finance*, 9(3):203–208, 1999.

[7] *Basel Committee on Banking Supervision. International Convergence of Capital Measurement and Capital Standards, a revised framework*, 2004.

[8] J. Beasley. Benchmarks for the portfolio selection problem. http://people.brunel.ac.uk/ mastjjb/jeb/info.html, Visited on July 2006.

[9] J.E. Beasley. OR-library: distributing test problems by electronic mail. *Journal of the Operational Research Society*, 41(11):1069–1072, 1990.

[10] F. Black. Capital market equilibrium with restricted borrowing. *Journal of business*, 45(3):444–455, 1972.

[11] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308, 2003.

[12] M.E. Blume and I. Friend. The asset structure of individual portfolios and some implications for utility functions. *The journal of finance*, 30(2):585–603, 1975.

[13] F. Catanas. On a neighbourhood structure for portfolio selection problems. Technical report, Departamento de Metodos Quantitativos do ISCTE, Lisboa, Portugal, 1998.

[14] T.J. Chang, N. Meade, J.E. Beasley, and Y.M. Sharaiha. Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research*, 27(13):1271–1302, 2000.

[15] V.K. Chopra and W.T. Ziemba. The effect of errors in means, variances and covariances on optimal portfolio choice. *Journal of Portfolio Management*, 19:6–11, 1993.

[16] Y. Crama and M. Schyns. Simulated annealing for complex portfolio selection problems. *European Journal of Operational Research*, 150:546–571, 2003.

[17] B. De Backer, V. Furnon, and P. Shaw. Solving Vehicle Routing Problems Using Constraint Programming and Metaheuristics. *Journal of Heuristics*, 6:501–523, 2000.

[18] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. Technical Report KanGAL 200001, Indian institute of technology, Kanpur, India, 2000.

[19] L. Di Gaspero, G. di Tollo, A. Roli, and A. Schaerf. Hybrid local search for constrained financial portfolio selection problems. In *Proceedings of CPAIOR*, 2007.

[20] L. Di Gaspero and A. Schaerf. EasyLocal++: an object-oriented framework for the flexible design of local-search algorithms. *SoftwarePractice & Experience*, 3(8):733–765, 2003.

[21] L. Diosan. A multi-objective evolutionary approach to the portfolio optimization problem. In *International Conference on Computational Intelligence for Modelling, Control and Automation, 2005*. IEEE Conference Proceedings, 2005.

[22] K. Doerner, W. Gutjahr, R. Hartl, C. Strauss, and C. Stummer. Ant colony optimization in multiobjective portfolio selection. In *Proceedings of the 4th Metaheuristics International Conference (MIC 2001)*, pages 243–248, 2001.

[23] K. Doerner, W.J. Gutjahr, R.F. Hartl, C. Strauss, and C. Stummer. Pareto ant colony

optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals of Operations Research*, 1:79–99, 2004.

[24] G. Dueck and P. Winker. New concepts and algorithms for portfolio choice. *Applied stochastic models and data analysis*, 8:159–178, 1992.

[25] M. Ehrgott, K. Klamroth, and C. Schwehm. An MCDM approach to portfolio optimization. *European Journal of Operational Research*, 155(3), 2004.

[26] A. Fernandez and S. Gomez. Portfolio selection using neural networks. *Computers & Operations Research*, 2005.

[27] K.V. Fernando. Practical portfolio optimization. Technical Report TR2/00 (NP3484), NAG Ltd, Oxford, UK, 2000.

[28] J.E. Fieldsend, J. Matatko, and M. Peng. Cardinality constrained portfolio optimisation. In *Proceedings of the Fifth International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'04), Lecture Notes in Computer Science (LNCS 3177)*, pages 788–793. Springer, 2004.

[29] M. Gilli and E. Këllezi. A global optimization heuristic for portfolio choice with VaR and expected shortfall. In *Computational Methods in Decision-making, Economics and Finance*, Applied Optimization Series. Kluwer Academic Publishers, 2001.

[30] M. Gilli, E. Këllezi, and H. Hysi. A data-driven optimization heuristic for downside risk minimization. *The Journal of Risk*, 3:1–19, 2006.

[31] F. Glover, J.M. Mulvey, and K. Hoyland. Solving dynamic stochastic control problems in finance using tabu search with variable scaling. In *Proceedings of the Metaheuristics International Conference (MIC 95)*, pages 429–448. Kluwer Academic Publishers, 1995.

[32] M.A. Gomez, C.X. Flores, and M.A. Osorio. Hybrid search for cardinality constrained portfolio optimization. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 1865–1866. Elsevier Science, 2006.

[33] E.M. Gratcheva and J.E. Falk. Optimal deviations from an asset allocation. *Computer & operations research*, 30(11):1643–1659, 2003.

[34] L. Guiso, T. Jappelli, and D. Terlizzese. Income risk, borrowing constraints and portfolio choice. *American economic review*, 86(1):158–172, 1996.

[35] P. Hansen and N. Mladenović. An introduction to variable neighborhood search. In *Metaheuristics: advances and trends in local search paradigms for optimization*, chapter 30, pages 433–458. Kluwer Academic Publishers, 1999.

[36] P. Van Hentenryck and L. Michel. *Constraint-Based Local Search*. The MIT Press, 2005.

[37] H.H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications*. Morgan Kaufmann Publishers, 2004.

[38] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. In *Feynman and computation: exploring the limits of computers*, pages 7–19. Perseus Books, 1999.

[39] ILOG. *ILOG Solver 5.1 User's manual*, 2001.

[40] R. Jansen and R. van Dijik. Optimal benchmark tracking with small portfolios. *The journal of portfolio management*, 28(2):9–22, 2002.

[41] N.J. Jobst, M.D. Horniman, C.A. Lucas, and G. Mitra. Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints. *Quantitative Finance*, 1:489–501, 2001.

[42] J. Farrell Jr. *Portfolio Management: Theory and application*. McGraw–Hill, 1997.

[43] J.G. Kallberg and W.T. Ziemba. Comparison of alternative utility functions in portfolio selection problems. *Management Science*, 29:1257–76, 1983.

[44] J. Kallrath. *Modeling Languages in Mathematical Optimization*. Springer, 2004.

[45] H. Kellerer, R. Mansini, and M.G. Speranza. On selecting a portfolio with fixed costs and minimum transaction lots. *Annals of Operations Research*, 99:287–304, 2000.

[46] H. Kellerer and D. Maringer. Optimization of cardinality constrained portfolios with a hybrid local search algorithm. *OR Spectrum*, 25(4):481–495, 2003.

[47] G. Kendall and Y. Su. A particle swarm optimisation approach in the construction of optimal risky portfolios. In *Proceedings of the 23rd IASTED International Multi-Conference on artificial intelligence and applications*, pages 140–145, 2005.

[48] H. Konno and A. Wijayanayake. Mean-absolute deviation portfolio optimization model under transaction costs. *Journal of the Operational Research Society of Japan*, 42(4), 1999.

[49] H. Konno and A. Wijayanayake. Portfolio optimization problem under concave transaction costs and minimal transaction unit constraints. *Mathematical Programming*, 89:233–250, 2001.

[50] H. Konno and H. Yamazaki. Mean absolute deviation portfolio optimization model and its applications to Tokyo stock market. *Management Science*, 37:519–531, 1991.

[51] D. Lin, S. Wang, and H. Yan. A multiobjective genetic algorithm for portfolio selection. Technical report, Institute of Systems Science, Academy of Mathematics and Systems Science Chinese Academy of Sciences, Beijing, China, 2001.

[52] S. Liu and D. Stefek. A genetic algorithm for the asset paring problem in portfolio optimization. In *Operations Research and Its Application. Proc. First International Symposium, ISORA*, pages 441–449, 1995.

[53] A. Loraschi and A. Tettamanzi. An evolutionary algorithm for portfolio selection within a downside risk framework. In *Forecasting Financial Markets*, Series in Financial Economics and Quantitative Analysis, pages 275–285. John Wiley & Sons, 1997.

[54] A. Loraschi, A. Tettamanzi, M. Tomassini, and P. Verda. Distributed genetic algorithms with an application to portfolio selection problems. In *Artificial Neural Networks and Genetic Algorithms*, pages 384–387, 1995.

[55] R. Mansini, W. Ogryczak, and M.G. Speranza. LP solvable models for portfolio optimization: a classification and computational comparison. *IMA Journal of Management Mathematics*, 14(3), 2003.

[56] R. Mansini and M.G. Speranza. Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research*, 114(2):219–233, 1999.

[57] R. Mansini and M.G. Speranza. An exact approach for portfolio selection with transaction costs and rounds. *IIE transactions*, 37:919–929, 2005.

[58] D.G. Maringer. Optimizing portfolios with ant systems. In *International ICSC congress on computational intelligence: methods and applications*, pages 288–294, 2001.

[59] D.G. Maringer. Portfolioselektion bei transaktionskosten ung ganzzahligkeitsbeschränkungen. *Zeitschrift für Betriebswirtschaft*, 72(11):1155–1176, 2002.

[60] D.G. Maringer. *Portfolio Management with heuristic optimization.* Springer, 2005.

[61] D.G. Maringer and P. Winker. Portfolio optimization under different risk constraints with modified memetic algorithms. Technical Report 2003-005E, University of Erfurt, Faculty of Economics, Law and Social Sciences, 2003.

[62] H. Markowitz. Portfolio selection. *Journal of Finance*, 7(1):77–91, 1952.

[63] H. Markowitz. *Portfolio selection, efficient diversification of investments.* John Wiley & Sons, 1959.

[64] M. Milano and A. Roli. MAGMA: A multiagent architecture for metaheuristics. *IEEE Trans. on Systems, Man and Cybernetics – Part B*, 34(2):925–941, 2004.

[65] R. Moral-Escudero, R. Ruiz-Torrubiano, and A. Suárez. Selection of optimal investment with cardinality constraints. In *Proceedings of the IEEE World Congress on Evolutionary Computation (CEC 2006)*, pages 2382–2388, 2006.

[66] L. Mous, V.A.F. Dallagnol, W. Cheung, and J. van den Berg. A comparison of particle swarm optimization and genetic algorithms applied to portfolio selection. In *Proceedings of Workshop on Nature Inspired Cooperative Strategies for Optimization NICSO 2006*, pages 109–121, 2006.

[67] D.N. Nawrocki. Portfolio optimization, heuristics and the "Butterfly Effect". *Journal of Financial Planning*, pages 68–78, 2000.

[68] C.S. Ong, J.J. Huang, and G.H. Tzeng. A novel hybrid model for portfolio selection. *Applied Mathematics and Computation*, 169:1195–1210, 2005.

[69] L. Paquete and T. Stützle. A study of stochastic local search algorithms for the biobjective QAP with correlated flow matrices. *European Journal of Operational Research*, 169:943–959, 2006.

[70] G. Pesant and M. Gendreau. A Constraint Programming Framework for Local Search Methods. *Journal of Heuristics*, 5:255–279, 1999.

[71] R.T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–41, 2002.

[72] E. Rolland. A tabu search method for constrained real number search:applications to portfolio selection. Technical report, Dept. of accounting and management information systems, Ohio State University, Columbus. U.S.A., 1997.

[73] A. Schaerf. Local search techniques for constrained portfolio selection problems. *Comput. Econ.*, 20(3):177–190, 2002.

[74] K.A. Smith. Neural networks for combinatorial optimization: a review of more than a decade of research. *INFORMS J. on Computing*, 11(1):15–34, 1999.

[75] K. Socha. ACO for Continuous and Mixed-Variable Optimization. In *Proceedings of ANTS 2004 – Fourth International Workshop on Ant Colony Optimization and Swarm Intelligence*, volume 3172 of *LNCS*, pages 25–36. Springer-Verlag, Berlin, Germany, 2004.

[76] M.G. Speranza. A heuristic algorithm for a portfolio optimization model applied to the Milan stock market. *Computers & Operations Research*, 23(5):433–441, 1996.

[77] M. Steiner and H. G. Wittkemper. Portfolio optimization with a neural network implementation of the coherent market hypothesis. *European Journal of Operational Research*, 1(3):27–40, 1997.

[78] F. Streichert, H. Ulmer, and A. Zell. Evolutionary algorithms and the cardinality constrained portfolio optimization problem. In *Selected Papers of the International Conference on Operations Research (OR 2003)*, pages 253–260. Springer Verlag, 2003.

[79] F. Streichert, H. Ulmer, and A. Zell. Comparing discrete and continuous genotypes on the constrained portfolio selection problem. In *Genetic and Evolutionary Computation Conference - GECCO 2004*, volume 3103 of *LNCS*, pages 1239–1250. Springer Verlag, 2004.

[80] F. Streichert, H. Ulmer, and A. Zell. Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem. In *Congress on evolutionary computation (CEC 2004)*, pages 932–939, 2004.

[81] R. Subbu, P. Bonissone, N. Eklund, S. Bollapragada, and K. Chalermkraivuth. Multi-objective financial portfolio design: A hybrid evolutionary approach. In *IEEE Congress on Evolutionary Computation (CEC 2005)*, 2005.

[82] J. Tobin. Liquidity preference as behavior towards risk. *Review of economic studies*, 26(1):65–86, 1958.

[83] J. Tobin. The theory of portfolio selection. In *The theory of interest rates*. Macmillan & Co.Ltd, 1965.

[84] G. Vedarajan, L.C. Chan, and D.E. Goldberg. Investment portfolio optimization using genetic algorithms. In *Late Breaking Papers at the Genetic Programming 1997 Conference*, pages 255–263. Stanford Bookstore, Stanford University, California, 1997.

[85] S.M. Wang, J.C. Chen, H.M. Wee, and K.J. Wang. Non-linear stochastic optimization using genetic algorithm for portfolio selection. *International Journal of Operations Research*, 3(1):16–22, 2006.

[86] G.A. Whitmore and M.C. Findlay (eds.). *Stochastic dominance: An approach to Decision-making under risk*. D.C.Heath, 1978.

[87] D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[88] Y. Xia, S. Wang, and X. Deng. Compromise solution to mutual funds portfolio selection with transaction costs. *European Journal of Operational Research*, 134(3):564–581, 2001.

[89] Y.S. Xia, B.D. Liu, S. Wang, and K.K. Lai. A model for portfolio selection with order of expected returns. *Computers & Operations Research*, 27(5), 2000.

[90] M.R. Young. A minimax portfolio selection rule with linear programming solution. *Management Science*, 44(5):673–683, 1998.

[91] H.G. Zimmermann and R. Neuneier. Active portfolio-management with neural networks. In *Proceedings of Computational Finance, CF1999*. Springer Verlag, 1999.