Applied Soft Computing 36 (2015): 125 – 142

DOI: http://dx.doi.org/10.1016/j.asoc.2015.06.053

A Memetic Algorithm for Cardinality-Constrained Portfolio Optimization with Transaction Costs

Rubén Ruiz-Torrubiano[a,*], Alberto Suárez[b]

[a] *Gruber&Petters, Belvederegasse 11, 2000 Stockerau, Austria*
[b] *Computer Science Department, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid, Spain*

Abstract

A memetic approach that combines a genetic algorithm (GA) and quadratic programming is used to address the problem of optimal portfolio selection with cardinality constraints and piecewise linear transaction costs. The framework used is an extension of the standard Markowitz mean-variance model that incorporates realistic constraints, such as upper and lower bounds for investment in individual assets and/or groups of assets, and minimum trading restrictions. The inclusion of constraints that limit the number of assets in the final portfolio and piecewise linear transaction costs transforms the selection of optimal portfolios into a mixed-integer quadratic problem, which cannot be solved by standard optimization techniques. We propose to use a genetic algorithm in which the candidate portfolios are encoded using a set representation to handle the combinatorial aspect of the optimization problem. Besides specifying which assets are included in the portfolio, this representation includes attributes that encode the trading operation (sell/hold/buy) performed when the portfolio is rebalanced. The results of this hybrid method are benchmarked against a range of investment strategies (passive management, the equally weighted portfolio, the minimum variance portfolio, optimal portfolios without cardinality constraints, ignoring transaction costs or obtained with L_1 regularization) using publicly available data. The transaction costs and the cardinality constraints provide regularization mechanisms that generally improve the out-of-sample performance of the selected portfolios.

1. Introduction

The classical framework for the selection of optimal portfolios was established by H. Markowitz in [25]. The problem consists in finding the allocation of a fixed budget in a universe of assets that maximizes the expected return

*Corresponding author

Email addresses: `ruben.rtorrubiano@grupet.at` (Rubén Ruiz-Torrubiano), `alberto.suarez@uam.es` (Alberto Suárez)

Preprint submitted to Elsevier *February 24, 2015*

from the investment in a given period while minimizing the corresponding risk. Since the future evolution of stock returns is uncertain, these returns are modeled as random variables. In the standard Markowitz formulation the risk is quantified in terms of the variance of the portfolio returns. Portfolio selection is therefore a multiobjective optimization task with two conflicting goals: The maximization of profit and the minimization of risk. It can be formulated as a quadratic programming (QP) problem that can be readily solved using standard quadratic optimization techniques. One of the shortcomings of the standard (unconstrained) Markowitz framework is that small variations in the inputs of the model (i.e. in the vector of expected values or in the covariance matrix of the asset returns) often lead to large changes in the composition of the resulting portfolios [7]. Another important drawback is that the portfolios selected generally have poor out-of-sample performance (see [29] and references therein). To address these issues additional constraints can be considered in the model. For instance, it is possible to include no short-selling constraints [16], which restrict all portfolio weights to be non-negative. Besides improving the robustness and performance of the portfolios, these additional constraints reflect actual restrictions in real-world applications. In particular, it is necessary to take into account the impact of the transaction costs incurred when the portfolio is rebalanced. Furthermore, limiting the number of assets in which the portfolio invests makes the management of the portfolio simpler. When cardinality constraints are included the problem becomes NP-Complete [31]. Standard QP solvers can no longer be used to address the portfolio selection problem. Therefore, one needs to resort to other types of methods to find near-optimal solutions at a reasonable computational cost.

In this paper, we propose to use a memetic approach that combines a genetic algorithm (GA) [15] with an extended set encoding and quadratic programming (QP) to address the problem of portfolio optimization, taking into account transaction costs and other realistic constraints, such as cardinality constraints (restrictions on the maximum number of assets in the portfolio), minimum trading size constraints (restrictions on the minimum amount of assets that can be bought or sold), minimum and maximum bounds on variables or groups of variables (to limit the fraction of investment in a particular asset or group of assets) and no short-selling constraints (the portfolio weights are non-negative).

The proposed memetic approach is benchmarked against other portfolio selection algorithms in experiments that quantify both the in-sample and out-of-sample performance. In-sample performance measures are used to assess how effective the optimization algorithm is and to what extent do the constraints considered affect the value of the objective function that is being optimized. However, practitioners are primarily interested in the out-of-sample performance of the portfolio: Using the information that is available at the time of the investment, how does one allocate a fixed budget among different assets so that the expected future return of the portfolio is maximized, while minimizing the corresponding risk? In this respect, the results of the current investigation are in agreement with the observation that in-sample performance is generally not a good predictor of out-of-sample gains [28, 10, 2]. The situation is analogous to the problem of *overfitting* in supervised learning [3]: High predictive accuracy in the training data does not guarantee a good generalization performance (i.e. high predictive accuracy in unseen instances).

A novel contribution of the memetic approach proposed in this work is the use of a set encoding for the candidate solutions that specifies not only which assets are included in the portfolio but also the type of trade (buy/hold/sell) that is carried out for each asset to rebalance the portfolio. The RAR crossover operator, which was introduced in [38], is adapted in this work to this extended set representation. The adapted RAR crossover operation produces individuals that satisfy all the constraints in the problem, so that no penalty functions or repair mechanisms are needed. An additional contribution of the current investigation is to illustrate how cardinality constraints and transaction costs act as regularization terms. Including these constraints in the optimization problem generally improves the robustness and out-of-sample performance of the portfolios.

The paper is organized as follows: Previous approaches to the problem are reviewed in Section 2. In Section 3 the problem of mean-variance portfolio selection under transaction costs is described. Section 4 introduces the memetic approach proposed in this work to address the problem. The effectiveness of this approach is illustrated in experiments whose results are presented and discussed in Section 5. Finally, Section 6 summarizes the conclusions and perspectives of the current investigation.

2. Related work

The problem of optimal portfolio selection has been extensively studied in the literature [6]. It can be formulated in two different settings: Single-period (static) and multi-period (dynamic) portfolio selection. Single-period portfolio selection consists in allocating a fixed budget among a collection of assets during a period in which the composition of the portfolio is held constant. The objective is to achieve an optimal tradeoff between the expected return and the risk of the investment [26]. In a multi-period setting the goal is to identify optimal investment policies that involve dynamically trading of the portfolio assets up to a specified time horizon [27, 44, 4]. In this case one typically maximizes an utility function that takes into account the expected future returns and the risk of the porftolio up to the investment horizon. It is also possible to design a dynamic investment strategy using a myopic approach, in which the multi-period problem is cast as a sequence of consecutive single-period problems. If the distribution of asset returns is predictable an optimized multi-period strategy should outperform a myopic one. These types of advantages are known in the financial literature as hedging demands [6]: By deviating from the single-period portfolio choice one tries to hedge against future changes in the investment conditions. However, hedging demands appear to be difficult to realize in practice: As a consequence of the uncertainty in the prediction of the the time-varying distribution of asset returns [35] the differences in the out-of-sample performance of a myopic and a truly dynamic approach in actual multi-period portfolio selection problems are fairly small [10, 2].

The goal of this work is to extend the Markowitz mean-variance framework [26] to consider realistic constraints and, specially, transaction costs. The effects of transaction costs in a single period setting are generally small. However, because costs are non-negative, their effects accrue with time. In consequence, they eventually become a significant factor in the design of long-term investment strategies. Therefore, to assess the out-of-sample performance of the portfolios identified by the different investment strategies we adopt the "rolling window" procedure used in [28]: A sequence of investment decisions is considered. At the beginning of each investment period the composition of the portfolio is determined on the basis of the information on the asset returns in a time-window of fixed size that immediately precedes that period. We then compute and store the portfolio returns (including transaction costs) in the period under consideration. After that we slide the fixed-size window forward and select the portfolio that will be held in the following investment period. These steps are repeated until the final investment horizon is reached. Although the objective of these rolling window experiments is not to address the multi-period portfolio selection problem, empirical studies have shown that the differences between the performance of sequential single-period investment strategies and the corresponding optimized dynamic trading strategies are generally small [10, 2]. Therefore, myopic strategies, which are simpler to compute, are often preferred to truly multi-period ones in practice.

The extensions of the standard Markowitz framework to consider realistic constraints and transaction costs generally lead to complex (NP-complete) optimization problems that cannot be addressed using exact numerical techniques. In this work we propose metaheuristics to address this problem. The application of evolutionary and biologically inspired methods to financial problems has been the object of recent interest in the soft computing community [5]. One of the first investigations in which a genetic algorithm was used to address this problem is [8]. In that work the space of cardinality constrained portfolios was searched using a combination of a genetic algorithm [15], tabu search [12] and simulated annealing [19]. The candidate solutions were encoded using chromosomes composed of both discrete and real genes. No transaction costs were considered. As a consequence of the mixed encoding used, the crossover and neighborhood operators needed to handle both the discrete and the continuous constraints of the problem. This complicated the design of these operators. In the approach proposed in the current work the continuous part of the problem is handled separately by specialized techniques. In this manner the genetic algorithm can focus on the combinatorial search.

In [53] the cardinality constraints were handled using a clustering algorithm to reduce the size of the investment universe: After performing K-means clustering, the investor selects one asset from each cluster. The optimization is then carried out in the restricted space of the selected assets using a genetic algorithm with real-valued encoding, arithmetic variable-point crossover and real uniform mutation. Lower and upper bounds on the fraction of capital invested on each asset in the portfolio and concentration of capital constraints were handled by heuristic weight standardization algorithms. An optimization technique based on spin-glass models [13] was applied to the problem of optimal portfolio selection without cardinality or bound constraints in [17]. Multi-objective Evolutionary Algorithms (MOEAs) were used in [50] and [49]. A hybrid representation in which chromosomes include both discrete and continuous genes was found to be better suited to the problem than a purely continuous encoding. Similar conclusions were drawn in [22], where a general extension of evolution strategies [45] for mixed-integer optimization problems was proposed. Differential evolution [48] was adapted for multi-objective portfolio optimization with cardinality and other real-world constraints in [20]. In [24], a multi-objective evolutionary algorithm combined with a learning heuristic that identifies the most promising assets in the Pareto front was used to solve the problem with cardinality and round lot constraints. The authors combined a hybrid encoding with repair mechanisms for the continuous constraints. Another hybrid representation was used in [46] and [18] to address the closely related problem of tracking a financial index with a portfolio composed of a small number of assets.

An extension of the Markowitz model that includes transaction costs from both brokerage fees and market illiquidity was introduced in [37]. In [36] the standard complementary pivot algorithm of Markowitz [26] was extended to take into account concave piecewise linear transaction costs, turnover constraints and approximate minimum trading size constraints. Non-parametric universal portfolios were adapted in [34] to enhance their short-term performance. A non-linear programming technique was applied by Yoshimoto [56] to a portfolio selection problem with V-shaped transaction costs. That work showed that ignoring the transaction costs can result in inefficient portfolios. By contrast, considering transaction costs leads to the selection of portfolios that are more stable. Cardinality, turnover or minimum trading size constraints were not considered in that investigation. In [23] Lobo et. al. addressed non-convex portfolio optimization problems with transaction costs that include a fixed fee. They proposed an iterative heuristic algorithm that approximates the optimal portfolio by solving a series of convex relaxations of the original problem. The resulting portfolio was suboptimal but had the advantage of being an upper bound on the optimal solution. They also showed that in real-world cases the bound is generally tight, even for large problems. The same approach can also be used for index tracking. In [30], the authors proposed to rescale the objective function by the amount of wealth invested after transaction costs are subtracted. The resulting model is a fractional programming problem that can be addressed by convex optimization techniques. Best and Hlouskova [1] applied a modified quadratic programming active set algorithm to solve the mean-variance problem with transaction costs in an investment universe of size N. The transaction costs can be accounted for by defining a $3N$-dimensional optimization problem with $3N$ additional constraints. To reduce the complexity associated with the increase of the dimensionality of the optimization space, they proposed an algorithm that works in N-dimensions, in which the transaction costs were accounted for implicitly rather than explicitly. No cardinality or turnover constraints were considered in that work.

In [7] optimal portfolio selection was formulated as a regularized regression problem. The objective function included a penalty term proportional to the L_1 norm of the vector of portfolio weights $||\mathbf{w}||_1$ as in *lasso regression* [51]. As a consequence of the properties of the L_1 norm, when the weight of this type of penalty term is sufficiently large, the portfolios obtained are sparse and invest in only a subset of the assets available for investment. In general, these sparse portfolios are more stable than minimum variance portfolios obtained without the L_1-norm penalty. In the current work we consider an extension of this idea and include in the objective function an L_1 term that penalizes instead the differences between the portfolio weights before and after rebalancing $\left|\left|\mathbf{w} - \mathbf{w}^{(0)}\right|\right|_1$. This form of the penalty is similar to the term that appears in the objective function when transaction costs are considered. We show that the inclusion of this type of penalties leads to the selection of portfolios whose composition is more stable over time. As a result, they are simpler to manage, have lower rebalancing costs and generally exhibit good out-of-sample performance.

3. Portfolio Management with Transaction Costs

Consider the problem of managing a portfolio that invests in a universe of N assets. The portfolio can be rebalanced at times $t = 1, \ldots, T$. Its composition is held fixed in the interval $[t-1, t)$. At time t the portfolio is rebalanced with the goal of maximizing the expected return in the interval $[t, t+1)$, which is how the profit is measured, while minimizing the corresponding variance, which is taken as a measure of the risk of the investment. A self-financing constraint is imposed by requiring that the value of the investment at t, after rebalancing, plus the costs of the transactions made to modify the composition of the portfolio is equal to the value of the investment at t^-, before rebalancing. Assuming piecewise linear transaction costs and incorporating the self-financing constraint, the expected return of the portfolio is

$$\mathbb{E}\left[r_P(t)\right] = \mathbf{w}(t)^T \cdot \hat{\mathbf{r}} - \boldsymbol{\kappa}^T \cdot \left|\mathbf{w}(t) - \mathbf{w}^{(0)}(t)\right|, \tag{1}$$

where $\hat{\mathbf{r}} = \{\mathbb{E}\left[r_i(t)\right]\}_{i=1}^N$ is the vector of expected returns of the individual assets, $\boldsymbol{\kappa}$ is the vector of (non-negative) transaction costs and $\mathbf{w}^{(0)}(t)$ and $\mathbf{w}(t)$ are the asset weights before and after rebalancing at time t, respectively.

The risk of the investment is quantified in terms of the variance of the portfolio

$$\mathrm{Var}\left[r_P(t)\right] \quad = \quad \mathbf{w}^{\mathrm{T}}(t) \cdot \boldsymbol{\Sigma} \cdot \mathbf{w}(t), \tag{2}$$

where $\boldsymbol{\Sigma}$ is the $N \times N$ covariance matrix of the asset returns. A detailed derivation of these expressions is given in the Appendix.

Cardinality constraints set a limit K on number of assets on which the portfolio invests. These constraints are useful because, besides simplifying the management of the investment, they generally improve the robustness and stability of the portfolio. To encode these types of constraints it is convenient to introduce an N-dimensional vector of binary variables $\mathbf{z}$. The ith component of this vector specifies whether asset i is included in the final portfolio ($z_i = 1$) or not ($z_i = 0$). Since the investment horizon is fixed, in what follows we drop the time index and simply use $\mathbf{w}^{(0)}$ for the vector of portfolio weights prior to rebalancing and $\mathbf{w}$ for the vector of portfolio weights immediately after rebalancing. Using these conventions, the optimal portfolio is the solution of the constrained minimization problem

$$\min_{\mathbf{z},\mathbf{w}} \left[{\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \mathbf{\Sigma}^{[\mathbf{z},\mathbf{z}]} \cdot \mathbf{w}^{[\mathbf{z}]} - \alpha \left({\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \hat{\mathbf{r}}^{[\mathbf{z}]} - \boldsymbol{\kappa}^T \cdot \left| \mathbf{w} - \mathbf{w}^{(\mathbf{0})} \right| \right) \right] \tag{3}$$

$${\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \mathbf{1} + \boldsymbol{\kappa}^T \cdot \left| \mathbf{w} - \mathbf{w}^{(\mathbf{0})} \right| = 1 \tag{4}$$

$$\mathbf{a}^{[\mathbf{z}]} \le \mathbf{w}^{[\mathbf{z}]} \le \mathbf{b}^{[\mathbf{z}]}, \;\; \mathbf{a}^{[\mathbf{z}]} \ge \mathbf{0}, \;\; \mathbf{b}^{[\mathbf{z}]} \ge \mathbf{0} \tag{5}$$

$$\mathbf{l} \le \mathbf{A}^{[\mathbf{z}]} \cdot \mathbf{w}^{[\mathbf{z}]} \le \mathbf{u} \tag{6}$$

$$\mathbf{z}^{\mathrm{T}} \cdot \mathbf{1} \le K \tag{7}$$

$$w_i \ge w_i^{(0)} + P_i \;\;\text{or}\;\; w_i \le w_i^{(0)} - S_i \;\;\text{or}\;\; w_i = w_i^{(0)} \quad i = 1, \ldots, N. \tag{8}$$

The inputs of the optimization problem are $\hat{\mathbf{r}}$, the vector of expected asset returns, and $\mathbf{\Sigma}$, the covariance matrix of these returns, which are estimated from historical data. The column vector $\mathbf{w}^{[\mathbf{z}]}$ is obtained by removing from $\mathbf{w}$ those components i for which $z_i = 0$. Similarly, the matrix $\mathbf{A}^{[\mathbf{z}]}$ is obtained by eliminating the columns of $\mathbf{A}$ for which $z_i = 0$. Finally, $\mathbf{\Sigma}^{[\mathbf{z},\mathbf{z}]}$ is obtained by removing from $\mathbf{\Sigma}$ the rows and columns for which the corresponding indicator is zero ($z_i = 0$). The symbols $\mathbf{0}$ and $\mathbf{1}$ denote vectors whose components are all 0 or all 1, respectively. More details on the derivation of the optimization model (in particular the form of the expected return of the portfolio and the self-financing constraint (4)) are given in the Appendix.

The objective function consists of three terms: The first one is the variance of the portfolio, which is to be minimized. The second one corresponds to the expected return of the portfolio, which we wish to maximize and is therefore included with a negative sign. The last one corresponds to the adjustment of the expected returns due to transaction costs. The positive constant $\alpha > 0$ determines the importance of the terms corresponding to the cost-adjusted expected return in the objective function. Alternatively, one could minimize the variance subject to a fixed or minimum level of expected return, or maximize the expected return subject to an upper bound on the total variance of the portfolio. The problem can also be formulated as a multi-objective optimization problem with two independent objectives. All these different formulations are mathematically equivalent. As shown in Subsection 3.1, the formulation (3)-(8) is more convenient for the approach taken in this study because the term with the absolute value difference weighted by the constant α resembles a regularization term.

Equation (4) reflects the *self-financing* constraint, which ensures that the value of the portfolio before rebalancing is equal to the value of the portfolio after rebalancing plus the transaction costs incurred. *Minimum and maximum investment* constraints, which set a lower and an upper bound on the investment of each asset in the portfolio, are encoded in the restriction (5). In this constraint $\mathbf{a}$ and $\mathbf{b}$ are $N \times 1$ column vectors whose components are the lower and upper bounds on the portfolio weights, respectively. Inequality (6) corresponds to *capital concentration* constraints. The m-th row of the $M \times N$ matrix $\mathbf{A}$ is the vector of coefficients of the linear combination that defines this constraint. The $M \times 1$ column vectors $\mathbf{l}$ and $\mathbf{u}$ correspond to the lower and upper bounds of the M linear restrictions, respectively. Capital concentration constraints can be used, for instance, to limit the amount of capital invested in a group of assets, so that investor preferences for certain asset classes can be taken into account in the optimization. Expression (7) is the *cardinality constraint*, which sets a bound on the maximum number of assets that can be included in the final portfolio. Finally, the investor can impose *trading size* or *turnover* constraints (8). These constraints reflect the fact that the investor may not wish to modify the portfolio by buying or selling small quantities of assets [9]. Market restrictions that specify minimal trading volumes can be handled in a similar way. Trading size constraints are difficult to handle because they are disjunctive. The solution space is partitioned into multiple feasible regions that are separated by forbidden regions. Specifically, for each asset, only one out of three mutually exclusive alternatives can occur: (*i*) The change is greater than or equal to $S_i \geq 0$ when selling, (*ii*) an amount of asset greater than or equal to $P_i \geq 0$ is purchased, (*iii*) the asset is neither sold nor purchased.

3.1. Regularization

In the classical Markowitz model, the inputs of the optimization model (the vector of means and the covariance matrix of the asset returns) are typically estimated from historical data. These historical estimates can be poor predictors of future behavior. Furthermore, small fluctuations in the values of these inputs often induce large modifications of the estimated optimal portfolio. This is an undesirable instability [28] and renders the problem *ill-posed* in the Hadamard sense[1]. This lack of stability and sensitivity to the model inputs generally results in poor out-of-sample performance. Several authors have pointed out that *regularization* techniques can be a way to avoid instability and improve the generalization performance of the portfolios selected [16, 28]. Regularization of ill-posed problems was originally introduced by Tikhonov in the context of least-squares problems [52]. The main idea of Tikhonov regularization is to add a term proportional to the squared L_2 norm of the solution vector to the

[1]Loosely speaking, a problem is *well-posed* if it satisfies the following three requirements: *i*) A solution exists, *ii*) the solution is unique and *iii*) the solution is stable in the sense that a small variation in the problem inputs causes only a small variation in the final solution. An *ill-posed* problem is defined as a problem violating at least one of the above conditions.

objective function to be minimized. This term basically amounts to adding a positive quantity to the singular values of the coefficient matrix [33]. Since very small singular values are characteristic of ill-conditioned matrices, the stability of the solution is thus improved. This procedure is known as *ridge regression* in the statistics literature [14]. One of the major drawbacks of this technique is that the resulting solutions tend to have small non-zero components in all of the problem variables [51]. This makes the resulting solution difficult to interpret. An alternative is to add a penalty term proportional to the L_1 norm of the solution vector. The resulting method is known as *lasso* ("least absolute shrinkage and selection operator") [51]. This lasso penalty enforces sparse solutions in which some of the regression coefficients are exactly zero [7]. This makes the solution more robust and easier to interpret.

The term corresponding to the transaction costs in the objective function (3) can be seen as a kind of lasso penalty term. Norm-constrained portfolios in which the standard Markowitz framework is extended by including a penalty term proportional to some norm of the portfolio weight vector $\mathbf{w}$ in the cost function have been investigated in [28]. If an L_1 norm penalty is used, provided that its strength is sufficiently large, some coefficients in $\mathbf{w}$ are forced to be zero [51]. Therefore, increasing the weight of this penalty in the cost function tends to reduce the cardinality of the portfolio.

The L_1 penalty associated with transaction costs is of a different type. It is proportional to

$$\left|\mathbf{w} - \mathbf{w}^{(\mathbf{0})}\right|. \tag{9}$$

That is, this term penalizes deviations from the *initial portfolio* $\mathbf{w}^{(\mathbf{0})}$. The sparsifying effect of this L_1 penalty favors the selection of portfolios in which some of the components of $\mathbf{w} - \mathbf{w}^{(\mathbf{0})}$ are exactly zero. This means that there is a preference not to perform transactions unless they lead to large expected returns with a low associated risk. The result is a regularization effect that avoids large fluctuations in the composition of the portfolio. Note that such fluctuations are undesirable because they result in large transaction costs, which reduce the net return of the portfolio. Including this L_1 penalty can also be seen as a form of regularization that is expected to improve the out-of-sample performance of the portfolio and not simply a way of minimizing the costs involved in rebalancing the portfolio.

The observation that transaction costs in the portfolio selection problem can have a regularization effect suggests the possibility of minimizing a modified objective function

$$\min_{\mathbf{z},\mathbf{w}} \; {\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \boldsymbol{\Sigma}^{[\mathbf{z},\mathbf{z}]} \cdot \mathbf{w}^{[\mathbf{z}]} - \alpha {\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \hat{\mathbf{r}}^{[\mathbf{z}]} + \boldsymbol{\gamma}^{T} \cdot \left|\mathbf{w} - \mathbf{w}^{(\mathbf{0})}\right| \tag{10}$$

in which $\boldsymbol{\gamma}$ represents the strength of the L_1 penalty term, which could be different from the actual transaction costs. In Section 5 we perform experiments in which we set $\boldsymbol{\gamma} = \gamma\mathbf{1}$, with the value of the scalar constant $\gamma \in \mathbb{R}$ selected by cross validation. Typically, a value that is larger than the actual transaction costs is selected. We will refer to this portfolio selection strategy as the *lasso approach*.

4. A memetic approach to portfolio selection

The portfolio selection problem without transaction costs and without the constraints (7) and (8) can be solved in polynomial time using standard quadratic optimization techniques (for instance, the one described in [11]). These techniques guarantee that the global optimum is reached, provided that some standard assumptions on the objective function and the constraints (positive-definiteness of the Hessian, continuous derivatives, quadratic or linear constraints) hold. However, the piecewise linear transaction costs cannot be directly handled by a standard QP solver because they are non-differentiable. Furthermore, the optimization problem with cardinality or turnover constraints becomes NP-Complete [31]. Specifically, the inclusion of cardinality constraints means that one needs to solve the combinatorial optimization problem of selecting the optimal subset of $k \leq K$ assets from the original investment universe, where K is the upper bound on the number of assets that can be included in the final portfolio. Finally, the restrictions on the minimum trading size introduce further combinatorial complexity in the problem: One needs to know whether the portfolio rebalancing process involves buying, selling or holding the position in each of the assets.

In this work, we propose a *memetic* algorithm to address this mixed-integer optimization problem. Memetic algorithms [32] are a specific kind of *hybrid* metaheuristic techniques [39] in which evolutionary algorithms are combined with specific knowledge of the problem at hand. As expressed by the No-Free-Lunch theorems for optimization [55], no general-purpose algorithm can perform better than random search when averaged over all classes of optimization problems. Therefore, to design effective algorithms, it is necessary to introduce some kind of bias that incorporates in the search specific knowledge of the problem to be solved. A simple way of incorporating this knowledge is to perform a local optimization step right after mutation or recombination. In combinatorial problems, hill climbing heuristics are frequently used to improve the offspring [32].

The memetic approach proposed in this work handles the problem by treating the combinatorial and the continuous aspects of the optimization task separately. A genetic algorithm with an extended set representation is used to address the combinatorial aspect of the problem. This algorithm generates candidate solutions that determine the subset of assets of the specified cardinality to be included in the portfolio and the type of trades to be made when rebalancing the current portfolio. The fitness of this candidate solution is the optimal value of the objective function in the restricted universe of investment specified by the candidate solution proposed by the genetic algorithm. This subordinate optimization problem does not have cardinality or turnover constraints, which means that it can be solved using standard QP solvers. The main advantage of this pure combinatorial encoding compared to mixed encodings like those used in [50], [22] and [8], where chromosomes with both discrete and continuous components are used, resides in the fact that the GA can focus on solving the combinatorial optimization problem of finding the optimal subset of assets and the trades to be performed without having to handle the continuous constraints. This separation has been shown to increase the performance and effectiveness of cardinality-constrained portfolio selection algorithms [31] [42] [41].

In the extended set encoding the candidate solutions are represented as a subset of the appropriate cardinality. Assets that belong to this set are included in the rebalanced portfolio. There are transaction costs associated with the asset trades that are needed to build the new portfolio, characterized by the vector of weights $\mathbf{w}$, from the original portfolio, characterized by the vector of weights $\mathbf{w}^{(0)}$. For each element in the set we include an additional attribute that specifies whether the corresponding asset is sold, purchased or is left unchanged in the portfolio rebalancing operation. Including the information in the chromosome is advantageous for two reasons: First, it is a way of directly handling the turnover constraint (8). Once the information of the presence or absence of a trade and its direction for each asset is known, only one of the three inequalities in (8) is relevant. Since each of the inequalities is linear when considered in isolation, the selected inequality can be included in the set of linear constraints of the subordinate optimization problem. Second, the absolute values in the objective function and in the budget constraint (4) can be eliminated once this attribute is known by making the substitution

$$\left|\mathbf{w} - \mathbf{w}^{(\mathbf{0})}\right| = \sum_{i \in \text{Sold}} \left(w_i^{(0)} - w_i\right) + \sum_{i \in \text{Purchased}} \left(w_i - w_i^{(0)}\right). \tag{11}$$

In this manner, these terms become differentiable. Furthermore, there is no need to increase the number of variables from N to $3N$ as is usually done to eliminate the absolute values in the objective function. Therefore, the simplifications that result from using the information provided by the candidate solutions in the extended set encoding allow the use of standard QP solvers to address the subordinate optimization problem. Note that this approach remains valid even if the transaction costs take a more complicated form (for instance, if they are different for buying and selling or for higher transaction volumes to account for liquidity effects).

The combinatorial search takes place in the space

$$\Theta = \{(s,t) : s \in \cup_{k=1}^{K} C_k(N),\, t \in \mathcal{T}\} \tag{12}$$

where $C_k(N)$ is the set of subsets of $\{1, \ldots, N\}$ with cardinality $k \leq K$ and $\mathcal{T} = \{\text{'buy'}, \text{'hold'}, \text{'sell'}\}$ is the set of values of the attributes that determine the type of trade that is made to rebalance the portfolio. The size of the search space is exponential in N

$$|\Theta| = 3 \sum_{k=1}^{K} \binom{N}{k}. \tag{13}$$

The GA encodes the candidate solutions as sets of fixed cardinality. The algorithm is run for every possible value of the cardinality constraint in the range $k = 1, \dots, K$. The best among the solutions obtained is finally selected.

In the extended set representation, each element in the set has an additional attribute whose value is in $\mathcal{T}$. The mutation operator (see Algorithm 4.1) exchanges a randomly selected asset in the portfolio encoded by the candidate solutions with another asset that is not present in that candidate portfolio. If the new asset was not in the original portfolio, which is characterized by the vector of weights $\mathbf{w}^{(0)}$, the value of the trading attribute is set to 'buy'. Otherwise, a random value in $\mathcal{T}$ for the trading direction attribute is assigned to this new element. Note that this mutation operator does not change the cardinality of the portfolio. As mentioned above, the GA is run for each value of the cardinality constraint, and the best solution among these runs is taken.

Algorithm 4.1 Extended set mutation operator used

1. Let A be the input chromosome, and let $k = |A|$.
2. Choose an element $g \in A$ randomly with probability $1/k$.
3. Choose an element h in the complement set A^C randomly with probability $1/(N-k)$.
4. If $\mathbf{w}_h^{(0)} = 0$, set the trading attribute of h to 'buy'.
5. Otherwise, choose the trading attribute of h randomly with equal probability from the set $\mathcal{T}$.
6. $A' = A \setminus \{g\} \cup \{h\}$.
7. Return A'.

In [41], the performance of genetic algorithms that use different crossover operators specially designed for set encodings were compared in several cardinality constrained optimization problems. These included portfolio selection without transaction costs. The best overall results were obtained when the Random Assortment Recombination (RAR) [38] operator was used to generate offspring. In this work we propose to adapt this operator so that it can be applied to chromosomes with an extended set encoding. The resulting algorithm is referred to as extended RAR (eRAR). This extended version of RAR is detailed in Algorithm 4.2. The operator includes a positive integer parameter c that controls the amount of common information from the parents retained by the offspring. The RAR operator makes use of six sets: Set A is the intersection set, which contains assets that appear in both parents. Set B includes the assets not present in any parent. Sets C and D contain the assets present in only one parent. Set E is initially empty ($E = \emptyset$). An additional set G is then created with c copies of the assets from A and B and one copy from the assets in C and D. The elements in G retain the label of the set from which they originate. A child chromosome is generated by selecting a asset at random from G in each iteration. If the asset originally comes from A or C and is not in E, then it is included in the child. Otherwise, if it originated in B or in D, then it is included in set E. Note that C and D contain the same elements. However, set D is used to exclude assets that are present in only one parent, whereas set C is used to include these assets.

Algorithm 4.2 Extended Random Assortment Recombination algorithm (eRAR)

1. Let k be the desiderd cardinality of the child chromosome.
2. Create auxiliary sets A, B, C, D, E:
   - A = elements present in both parents.
   - B = elements not present in any of the parents.
   - $C \equiv D$ elements present in only one parent.
   - $E = \emptyset$.
3. Build set G with c copies of elements from A and B, and 1 copy of the elements in C and D.
4. Initialize child chromosome $\phi = \emptyset$.
5. While $|\phi| < k$ and $G \neq \emptyset$:
   - Extract $g \in G$ without replacement.
   - DetermineAttribute(g).
   - If $g \in A$ or $g \in C$, and $g \notin E$, $\phi = \phi \cup \{g\}$.
   - If $g \in B$ or $g \in D$, $E = E \cup \{g\}$.
6. If $|\phi| < k$, add elements not yet included chosen at random until chromosome is complete.

The process is terminated when the child has the specified cardinality or when $G = \emptyset$. If the latter happens, then the child is completed with assets selected at random from those which have not been included up to that moment. Note that this step allows the introduction in the child of assets not present in any parent. The extended version eRAR handles the additional attribute that specifies the direction of the trade for each asset in the rebalancing operation by means of the function DetermineAttribute(g), which is described in Algorithm 4.3.

When there is a disagreement between several of the additional attributes that determine the type of trade for that asset in the parents, we consider two strategies: (i) Either we pick the one that has the highest fitness among all possible combinations of attributes or (ii) the value of the attribute in which the parents disagree is selected at random. The best performance corresponds to (i). However, the computations are unfeasible for large values of the cardinality constraint. Therefore, in the experiments (ii) is used for values of the cardinality constraint higher than 15, since for higher values the computations were not feasible with the available resources.

The fitness of the candidate solution is defined in terms of the solution of

Algorithm 4.3 DetermineAttribute(g): Extended attribute selection in eRAR

1. If the asset g is not present in the original portfolio, which is characterized by the weight vector $\mathbf{w}^{(0)}$, then the value of the attribute is 'buy'.
2. Otherwise
   (a) If the asset g is not present in any parent, then the trading direction attribute is selected with equal probability in the set $\mathcal{T}$.
   (b) If the asset g is present in only one parent, then the trading direction attribute of g in the child is set to the same value as in the parent.
   (c) If asset g is present in both parents:
   - If the trading direction attribute in both parents is equal, the child has the same value of this attribute as its parents.
   - If the trading direction attribute is different in the parents the combination of attributes with the highest fitness is chosen.

the subordinate optimization problem (3)

$$\text{Fitness}(\mathbf{z}) = -\min_{\mathbf{w}} \left({\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \mathbf{\Sigma}^{[\mathbf{z},\mathbf{z}]} \cdot \mathbf{w}^{[\mathbf{z}]} - \alpha \left({\mathbf{w}^{[\mathbf{z}]}}^{\mathrm{T}} \cdot \hat{\mathbf{r}}^{[\mathbf{z}]} - \boldsymbol{\kappa}^{T} \cdot \left| \mathbf{w} - \mathbf{w}^{(\mathbf{0})} \right| \right) \right) \tag{14}$$

subject to (4)-(6) and one of the inequalities of (8) for each included asset. A standard QP solver [11] is used to address this subordinate optimization problem.

The advantages of separately handling the combinatorial and the continuous optimization aspects of the problem are twofold: First, no repair mechanisms are needed when the continuous constraints cannot be satisfied by the current chromosome. Repair mechanisms can be very costly and generally have a negative impact on the performance of the GA. Since the continuous constraints are handled by the QP solver, the GA does not need to check the feasibility of the obtained portfolios. Second, the crossover operator, which focuses on the combinatorial search, can be implemented efficiently [40].

In the next section the effectiveness of this approach is illustrated in a series of experiments on real financial data. In this empirical study the hybrid method is compared with a range of strategies. The main conclusion of the study is that cardinality constraints and transaction costs act as regularization terms that allow the selection of sparse portfolios that are stable, robust and generally exhibit good out-of-sample performance.

5. Empirical evaluation

In this section we present the results of an empirical evaluation of the hybrid method for portfolio optimization described in the previous section. The performance of this algorithm is compared with reference strategies for portfolio selection, such as the naïvely ($1/N$) diversified portfolio, the minimum variance portfolio and regularized portfolios obtained with lasso penalties. Special attention is given to the effects of transaction costs and cardinality constraints. The experiments are carried out on three different datasets compiled by Fama and French[2], some of which have been used in previous studies [7] [28]. These data consist of time series of non-annualized monthly returns from June 1971 until December 2009. The first dataset (FF48) includes 48 industry portfolios. The second one (FF100) is the intersections of 10 portfolios formed on size (market equity, ME) and 10 portfolios formed on the ratio of book equity to market equity (BE/ME). The third dataset (FF38) contains 38 industry portfolios different from those included in FF48. Dates with missing values are discarded. The optimizations are performed using an Intel Core Duo machine with 2 GHz clock speed and 2 GB RAM.

The study is divided into in-sample and out-of-sample evaluation. The goal of in-sample evaluation is to determine the quality of the memetic algorithm as an optimization method. The question is how close is the portfolio selected by this algorithm to the globally optimal portfolio. Given that the cardinality-constrained portfolio optimization problem is NP-hard, only results relative to the best known solution can be given in most cases. Several studies have shown that portfolios that are optimal in-sample can have poor out-of-sample performance [29]. The reason is that the inputs for the optimization are based on estimations that are insufficient or inadequate for prediction. Borrowing some terminology from machine learning, this discrepancy between in-sample (training) and generalization performance is referred to as *overfitting*. Overfitting is a result of erroneously identifying regularities in the data that are used to estimate the inputs to the optimization problem (training data) as patterns that are relevant to make predictions on independent test data. The reliance on these spurious patterns is misleading and hinders the generalization capacity of the system [3]. For this reason, in the second part of this section the out-of-sample performance of the selected portfolios is investigated. As will be shown, good in-sample performance (as an optimization method) does not in general correspond to good generalization (out-of-sample) performance.

5.1. In-sample evaluation

The in-sample performance is assessed using all the available data (approximately 400 monthly returns) to calculate the vector of estimated expected returns $\tilde{\mathbf{r}} = \{\tilde{r}_i\}_{i=1}^N$

$$\tilde{r}_i = \frac{1}{T}\sum_{t=1}^{T} r_i(t), \qquad i = 1, \ldots, N \tag{15}$$

where N is the number of assets in the investment universe and T the number of values available for estimation. The sample estimate $\tilde{\mathbf{\Sigma}}$ of the covariance matrix

[2] http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

is

$$\tilde{\Sigma}_{ij} = \frac{1}{T-1} \sum_{t=1}^{T} \left(r_i(t) - \tilde{r}_i\right) \left(r_j(t) - \tilde{r}_j\right) \;\; \forall i, j. \tag{16}$$

The hybrid metaheuristic introduced in the previous section is used to solve the optimization problem (3)-(8) using the sample estimates of the expected value (15) and the covariance matrix of the returns (16) as inputs. The GA uses a steady-state population of 100 individuals. Crossover is always performed. A child is generated by applying the eRAR operator with parameter $c = 1$ to two parents selected in separate binary tournaments. In each binary tournament two individuals are picked at random. The one with the highest fitness is then selected for crossover. The newly generated offspring replaces the worst individual of the original population. The mutation operator described in the previous section is applied with probability 10^{-2}. The termination condition of the GA is set to a fixed number of generations (4500). The values of these parameters were determined in a series of exploratory experiments.

A first set of optimizations is made to calculate the *efficient frontier* of Pareto optimal portfolios. The efficient frontier is the collection of portfolios whose returns have the lowest possible variance for a fixed value of the expected portfolio return. From the dual perspective, Pareto optimal portfolios have a maximum expected return for a fixed value of the variance. These portfolios are the solution of the collection of optimization problems obtained by using in (3)-(8) as objective function

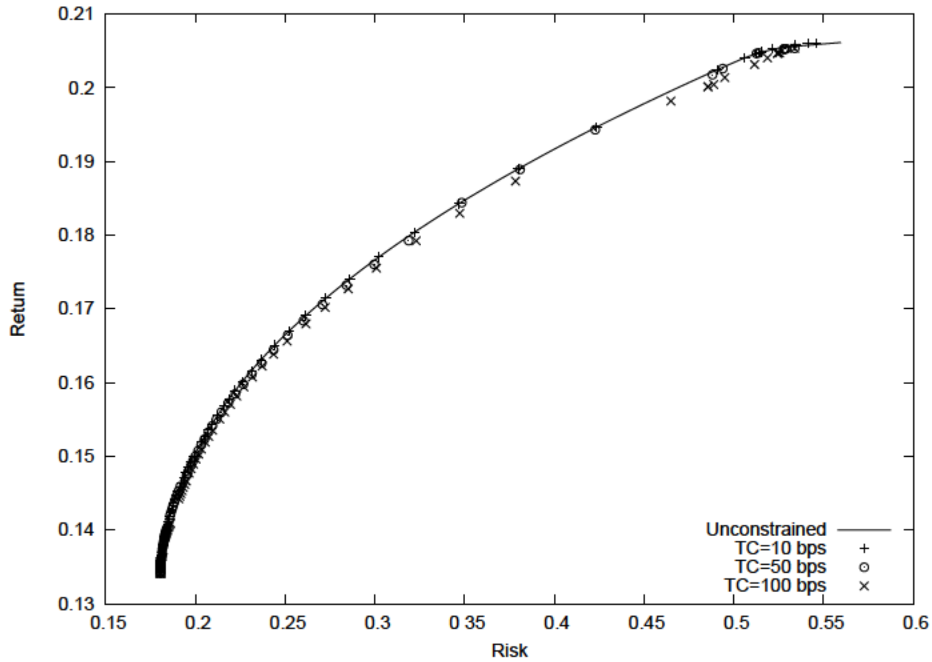
$$(1-\lambda)\mathbf{w}^{[\mathbf{z}]^{\mathrm{T}}} \cdot \tilde{\boldsymbol{\Sigma}}^{[\mathbf{z},\mathbf{z}]} \cdot \mathbf{w}^{[\mathbf{z}]} - \lambda \left(\mathbf{w}^{[\mathbf{z}]^{\mathrm{T}}} \cdot \tilde{\mathbf{r}}^{[\mathbf{z}]} - \boldsymbol{\kappa}^T \cdot \left| \mathbf{w} - \mathbf{w}^{(\mathbf{0})} \right| \right). \tag{17}$$

The efficient frontier of Pareto optimal portfolios is parameterized in terms of $\lambda \in [0, 1]$. The value $\lambda = 0$ corresponds to minimizing the variance. The value $\lambda = 1$ corresponds to maximizing the expected portfolio return, net of transaction costs. For the sake of simplicity, we assume equal transaction costs for all the assets $\{\kappa_i = \kappa,\, i = 1, \ldots N\}$. Taking into account different costs for different assets is straightforward and does not increase the complexity of the optimization problem. The efficient frontiers are then computed for several values of the transaction costs κ: 0, 10, 20, 30, 40, 50, and 100 basis points[3]. In all cases, the portfolios are restricted to invest in at most $K = 10$ different assets. We compute $N_F = 100$ portfolios in the efficient frontier by taking a grid of equidistant values of λ in the range $[0, 1]$. The efficient frontier that would be obtained with zero transaction costs if all the constraints, with the exception of the budget constraint (4), were removed is also computed for reference. The equally weighted $1/N$ portfolio is used as the initial portfolio

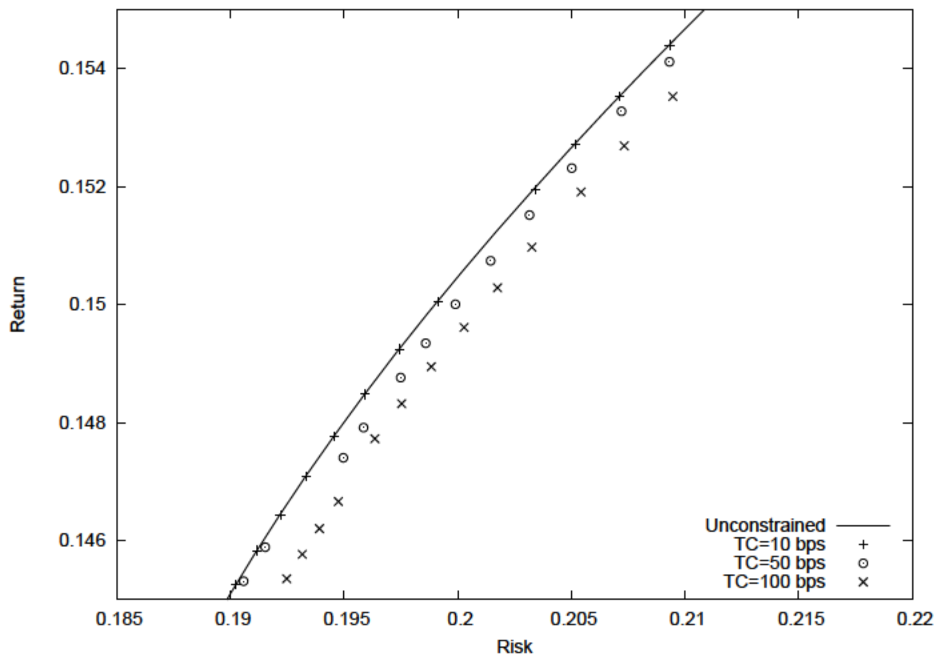
$$\mathbf{w}^{(0)} = \{1/N, 1/N, \ldots, 1/N\}. \tag{18}$$

Figure 1 displays the efficient frontiers obtained assuming different transaction costs. The results of these optimizations are summarized in Table 1. The

[3] 1 basis point (bps) = 0.01%



(a) Efficient frontiers



(b) Detail of the efficient frontiers

Figure 1: The efficient frontiers obtained by eRAR-GA in the FF48 dataset for different values of the transaction costs are displayed in 1(a). The detail given in 1(b) shows how the distance to the unconstrained efficient frontier increases with higher transaction costs.

Table 1: Comparison of in-sample results in the FF48 dataset using different values of the transaction costs.

Transaction costs	Best D	Success rate	Time (s)	Optimizations
0 bps	0.01378271	1.00	4262.4	$7.64 \cdot 10^7$
10 bps	0.13378017	1.00	4564.5	$8.91 \cdot 10^7$
20 bps	0.18238682	1.00	4487.2	$8.96 \cdot 10^7$
30 bps	0.27143231	1.00	4374.0	$9.05 \cdot 10^7$
40 bps	0.31587304	1.00	4379.6	$9.00 \cdot 10^7$
50 bps	0.36184243	1.00	4272.5	$8.95 \cdot 10^7$
100 bps	0.61696361	1.00	3920.7	$8.67 \cdot 10^7$

second column of this table displays the values of

$$D = \frac{1}{N_F} \sum_{i=1}^{N_F} \frac{\sigma_i^c - \sigma_i^*}{\sigma_i^*}, \tag{19}$$

which is a measure of the average relative horizontal distance between the actual and the unconstrained efficient frontiers. The value of σ_i^c in (19) is the standard deviation achieved for the ith portfolio on the actual efficient frontier, which is obtained considering all the constraints and transaction costs, and σ_i^* is the corresponding value on the unconstrained efficient frontier. The third column in Table 1 presents the success rate obtained by the algorithm. The success rate is the fraction of runs of the algorithm in which the best known solution at each point in the frontier is found. In our experiments, the algorithm is executed 5 times for each point on the frontier. The total run-time is given in the adjacent column. Finally, the last column shows the total number of quadratic optimizations performed in each execution.

The efficient frontier for the FF48 dataset is displayed in Figure 1(a) and, in greater detail, in Figure 1(b). As expected, the solutions that are optimal when transaction costs are considered are dominated by the solutions on the unconstrained efficient frontier. The distances between the constrained and the unconstrained efficient frontiers increase for higher transaction costs. The results displayed in Table 1 confirm these conclusions. The success rates, times and number of optimizations are similar in all cases. This means that the difficulty of the optimization problem is similar for the range of transaction costs considered.

In a second set of in-sample experiments we solve the optimization problem (3)-(8) using (15), (16) as inputs and compare the results to other benchmark strategies for portfolio optimization. The optimization is carried out with $\alpha = 2$ in (3). Similar conclusions are reached for other values of this parameter. The equally weighted $1/N$ portfolio is used as the initial portfolio for all the strategies. The goal is to rebalance this portfolio so that it satisfies the specified constraints and has the best performance in one investment period, which, in the experiments carried out, has a duration of one month. The evolutionary optimization methods (eRAR-GA with and without transaction costs) are run 20 times for each value of the transaction costs considered.

The performance is measured in terms of the expected return of the portfolios selected by the optimization procedure, taking into account the transaction costs

$$r_{exp} = \sum_{i=1}^{N} w_i \tilde{r}_i - \sum_{i=1}^{N} \kappa_i \left| w_i - w_i^{(0)} \right| \tag{20}$$

where $\mathbf{w} = \{w_i\}_{i=1}^{N}$ is the composition of the portfolio after rebalancing and $\tilde{r}_i$ are computed using (15). The Sharpe ratio [47] is used as a complementary performance measure. The Sharpe ratio is a measure of the risk-adjusted return of an investment. Usually the excess return over a benchmark (e.g. the risk-free rate) is used. For practical convenience, and since it does not affect the ranking of the portfolios, we directly use the returns in the numerator of the Sharpe ratio, without subtracting the risk-free rate. In terms of the sample estimate of the covariance matrix of the returns (16) the in-sample Sharpe ratio is

$$S_R = \frac{r_{exp}}{\sigma} = \frac{\sum_{i=1}^{N} w_i \tilde{r}_i - \sum_{i=1}^{N} \kappa_i \left| w_i - w_i^{(0)} \right|}{\sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} w_i \tilde{\Sigma}_{ij} w_j}}. \tag{21}$$

The performance of the portfolio selected by the GA algorithm with an extended set encoding is compared to portfolios that have been built using different benchmark investment strategies. Some of these strategies do not consider transaction costs explicitly in their formulation. It is therefore necessary to take into account the effect of transaction costs after the construction of the portfolio. To do this, assume that the composition of the portfolio immediately before rebalancing at time t is $\left\{ w_i^{(0)}(t), \quad i = 1, \dots N \right\}$, as in (47). The portfolio selected by strategy s, which does not take into account transaction costs, is characterized by the normalized weights

$$w_i^{(s)}(t), \quad i = 1, \dots N, \qquad \sum_{i=1}^{N} w_i^{(s)}(t) = 1 \tag{22}$$

after rebalancing at t. To take into account linear transaction costs, we use the self financing constraint

$$P(t^-) = P(t) + \sum_{i=1}^{N} \kappa_i \left| w_i^{(s)}(t) P(t) - w_i^{(0)}(t) P(t^-) \right|. \tag{23}$$

This implicit nonlinear equation can be solved to obtain $P(t)$, the value of the portfolio obtained by means of the investment strategy considered, as a function of the value of the portfolio before rebalancing $P(t^-)$. Taking into account the transaction costs, the net expected return is

$$r_{exp}^{(s)}(t) = \frac{P(t)}{P(t^-)}\left(\sum_{i=1}^{N} w_i^{(s)}(t)\tilde{r}_i + 1\right) - 1 \quad (24)$$
$$= \frac{P(t)}{P(t^-)}\left(\sum_{i=1}^{N} w_i^{(s)}(t)\tilde{r}_i - \left(\frac{P(t^-)}{P(t)} - 1\right)\right).$$

From the form of this expression, one can see that the effect of the transaction costs is, on the one hand, to lower the returns and, on the other hand, to reduce the amount of capital that is available for investment. The Sharpe ratio is

$$S_R^{(s)}(t) = \frac{\sum_{i=1}^{N} w_i^{(s)}(t)\tilde{r}_i - \left(\frac{P(t^-)}{P(t)} - 1\right)}{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} w_i^{(s)}\tilde{\Sigma}_{ij}w_j^{(s)}}}. \quad (25)$$

In this series of experiments, the performance of the proposed eRAR strategy is compared to the following five benchmark portfolios:

1. $1/N$: The naïvely diversified portfolio in which all N assets are given the same weight $1/N$. The transaction costs are ignored in the portfolio selection.
2. MinVar: The minimum variance portfolio. This portfolio is constructed by dropping the expected return constraint in the standard Markowitz model. The transaction costs are ignored in the portfolio selection.
3. No Card.: A portfolio built without the cardinality constraint but taking into account transaction costs. The problem can be formulated as a quadratic program in $3N$ dimensions by including two additional variables per asset: $d_i^+, d_i^- \in \mathbb{R}^+ \cup \{0\},\ i = 1, \ldots, N$. Two new linear constraints per variable need to be included:

$$w_i - d_i^+ \leq w_i^{(0)} \quad (26)$$

$$d_i^- + w_i \geq w_i^{(0)}. \quad (27)$$

   The terms corresponding to the transaction costs in the objective function and in the constraint (4) are replaced by

$$\sum_{i=1}^{N} \kappa_i(d_i^+ + d_i^-). \quad (28)$$

   This strategy is referred to as the standard Markowitz portfolio in the discussion.
4. Lasso: This type of portfolio is obtained using the lasso approach described in Subsection 3.1. The cardinality constraint is ignored. The value of $\boldsymbol{\gamma}^*$ used in the final optimization is estimated by leave-one-out cross-validation as follows: Let the training (in-sample) period be $[t_i, t_f]$.

For each $t = t_i, \ldots, t_f$ we leave the t-th return out and use the resulting training set to select an optimal portfolio according to those data. The portfolio $\mathbf{w}^{\gamma}(t)$, obtained using the value γ for the lasso penalty, is held on $[t, t+1)$. Its out-of-sample return in that period $(r^{\gamma}_{out}(t))$ is then recorded. As a result of this process, we have a time series $\{r^{\gamma}_{out}(t)\}_{t=t_i}^{t_f}$. We then calculate the mean return of this series $\hat{r}^{\gamma}_{out}$ and choose $\gamma^* = \max_{\gamma} \hat{r}^{\gamma}_{out}$. Using the returns in the first 60 months as training set, the value selected was $\boldsymbol{\gamma}^* = 3300$ bps for the FF48 dataset, $\boldsymbol{\gamma}^* = 3550$ bps for FF100 and $\boldsymbol{\gamma}^* = 2610$ bps for FF38.

5. IGNORETC: A portfolio that is selected by taking into account the cardinality constraint but ignoring transaction costs in the optimization. To this end, the parameter κ is set to 0. The optimization is carried out using the proposed hybrid GA approach.

In the tables in which the results of this empirical evaluation are presented the best value is highlighted in boldface and the second best value is underlined. Transaction costs are always considered in the measures of performance reported, even if they are ignored in the selection of the portfolio. Additionally, we perform non-parametric Wilcoxon sum rank tests [54] to compare the performance between the best method for every value of transaction costs against each of the other strategies (for each line the reference method is highlighted in boldface). Results in which the differences in performance are statistically significant at a 95% probability level are marked with an asterisk in the tables. The corresponding p-values, the success rates, means and standard deviations achieved by the eRAR-GA algorithms with and without transaction costs are provided as supplemental material in electronic form.

5.1.1. Dataset FF48

The in-sample results for the FF48 dataset are shown in Tables 2 and 3. In the absence of transaction costs, the strategy that obtains the best expected return is the standard Markowitz mean-variance portfolio (column "No Card."). When there are no transaction costs the standard Markowitz portfolio (i.e. without cardinality constraints) has the largest expected returns, followed the $1/N$ strategy. By contrast, when nonzero transaction costs are considered, the $1/N$ strategy, which does not incur transaction costs, has the best expected return. The second best results are obtained by the lasso strategy. This is reasonable because the value of $\gamma = 3300$ bps estimated by cross-validation is fairly large, which means that the lasso and the $1/N$ portfolios are very similar. In terms of Sharpe ratios, the best results without transaction costs are obtained by the portfolios selected by the eRAR strategies. With transaction costs up to 20 bps, the No Card. portfolio performs best. Above that value for the transaction costs the lasso strategy obtains the best results. Note that in terms of expected returns the strategy without cardinality constraints (No Card.) always obtains better results than the cardinality-constrained eRAR strategies. This is because the removal of a constraint necessarily improves the value of the optimum of the objective function.

Table 2: Comparison of expected in-sample returns for the different strategies in the FF48 dataset.

TC	1/N Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3300$ bps	eRAR Ignore TC	eRAR With TC
0	0.0150000*	0.011891*	**0.015951**	0.014763*	0.014812*	0.014812*
10	**0.015000**	0.010860*	0.014456*	0.014693*	0.013038*	0.012826*
20	**0.015000**	0.009830*	0.012975*	0.014622*	0.011267*	0.010786*
30	**0.015000**	0.008800*	0.011510*	0.014551*	0.009500*	0.008779*
40	**0.015000**	0.007770*	0.010060*	0.014480*	0.007737*	0.006776*
50	**0.015000**	0.006740*	0.008624*	0.014409*	0.005977*	0.004878*

Table 3: Comparison of in-sample Sharpe ratios for the different strategies in the FF48 dataset.

TC	1/N Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3300$ bps	eRAR Ignore TC	eRAR With TC
0	0.018795*	0.019705*	0.026014*	0.019448*	**0.026185**	**0.026185**
10	0.018795*	0.018016*	**0.023637**	0.019357*	0.023089*	0.022708*
20	0.018795*	0.016324*	**0.021270**	0.019265*	0.019989*	0.019127*
30	0.018795*	0.014628*	0.018916*	**0.019173**	0.016883*	0.015592*
40	0.018795*	0.012929*	0.016574*	**0.019080**	0.013773*	0.012053*
50	0.018795*	0.011227*	0.014243*	**0.018988**	0.010659*	0.008689*

5.1.2. Dataset FF100

Tables 4 and 5 summarize the in-sample results for the FF100 dataset. Without transaction costs, the standard Markowitz portfolio obtains the best in-sample returns, followed by the eRAR strategies. The lasso strategy is the best one when transaction costs are considered. The second best strategy is in this case the No Card. strategy, which, as in the previous case, obtains better results than the eRAR cardinality-constrained portfolios. In terms of Sharpe ratios, the best results for low transaction cost values are achieved by the eRAR strategies. However, the lasso strategy obtains the best Sharpe ratios with higher transaction costs.

5.1.3. Dataset FF38

The in-sample results for the FF38 dataset are summarized in Tables 6 and 7. As in the previous cases, the best strategy with zero transaction costs is the No Card. strategy. For higher values of the transaction costs, the 1/N strategy

Table 4: Comparison of in-sample expected returns for the different strategies in the FF100 dataset.

TC	1/N Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3550$ bps	eRAR Ignore TC	eRAR With TC
0	0.007800*	0.007800*	**0.012575**	0.011759*	0.012435*	0.012435*
10	0.007800*	0.007800*	0.010994*	**0.011714**	0.010574*	0.010732*
20	0.007800*	0.007800*	0.009460*	**0.011669**	0.008716*	0.009123*
30	0.007800*	0.007800*	0.007953*	**0.011624**	0.006863*	0.007377*
40	0.007800*	0.007800*	0.006487*	**0.011579**	0.005013*	0.006178*
50	0.007800*	0.007800*	0.005055*	**0.011534**	0.003167*	0.004344*

Table 5: Comparison of in-sample Sharpe ratios for the different strategies in the FF100 dataset.

TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3550$ bps	eRAR Ignore TC	eRAR With TC
0	0.010900*	0.010900*	0.022150*	0.016730*	**0.022714**	**0.022714***
10	0.010900*	0.010900*	0.019420*	0.016667*	0.019350*	**0.019612**
20	0.010900*	0.010900*	**0.016761**	0.016604*	0.015981*	0.016672*
30	0.010900*	0.010900*	0.014135*	**0.016541**	0.012605*	0.013493*
40	0.010900*	0.010900*	0.011561*	**0.016477**	0.009225*	0.011288*
50	0.010900*	0.010900*	0.009033*	**0.016414**	0.005838*	0.007945*

Table 6: Comparison of in-sample expected returns for the different strategies in the FF38 dataset.

TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 2610$ bps	eRAR Ignore TC	eRAR With TC
0	0.014690*	0.010892*	**0.015055**	0.014335*	0.014144*	0.014144*
10	**0.014690**	0.009579*	0.013615*	0.014213*	0.012424*	0.012746*
20	**0.014690**	0.008267*	0.012207*	0.014092*	0.010708*	0.011557*
30	**0.014690**	0.006955*	0.010824*	0.013970*	0.008996*	0.010683*
40	**0.014690**	0.005645*	0.009466*	0.013848*	0.007287*	0.009606*
50	**0.014690**	0.004336*	0.008120*	0.013726*	0.005581*	0.007930*

has the best expected returns. The lasso strategy, which is again very similar to the $1/N$ strategy is the second best for higher transaction costs. The same observation as in the previous cases holds for the No Card. and the eRAR strategies: The model with no cardinality constraints obtains better expected returns. In terms of Sharpe ratios, the No Card. portfolios obtain the best results without transaction costs, followed by the eRAR strategies.

5.2. Out-of-sample evaluation

The out-of-sample performance of the different strategies is evaluated in a simulated investment exercise. We are given a collection of N assets from which an investment portfolio can be built. As in the in-sample case, the data available consist of time series of returns for each of these assets $\left\{\{r_i(t)\}_{i=1}^N\right\}_{t=1}^T$. We fix a time horizon $t_{tr} \leq T$ that determines the amount of training data. The expected returns used as input in the optimization are estimated from these

Table 7: Comparison of in-sample Sharpe ratios for the different strategies in the FF38 dataset.

TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 2610$ bps	eRAR Ignore TC	eRAR With TC
0	0.018516*	0.019355*	**0.025376**	0.019506*	0.025366*	0.025366*
10	0.018516*	0.017044*	**0.022996**	0.019343*	0.022320*	0.022949
20	0.018516*	0.014728*	0.020658*	0.019179*	0.019270*	**0.020786***
30	0.018516*	0.012408*	0.019016*	0.018962*	0.016216*	**0.019093***
40	0.018516*	0.010084*	0.016080*	**0.018852**	0.013158*	0.017183*
50	0.018516*	0.007755*	0.013822*	**0.018688**	0.010095*	0.014290*

training data

$$\tilde{r}_i = \frac{1}{t_{tr}} \sum_{t=1}^{t_{tr}} r_i(t) \qquad i = 1, \ldots, N. \tag{29}$$

The sample estimate of the covariance matrix of these returns is

$$\tilde{\Sigma}_{ij} = \frac{1}{t_{tr} - 1} \sum_{t=1}^{t_{tr}} \left(r_i(t) - \tilde{r}_i\right)\left(r_j(t) - \tilde{r}_j\right) \ \forall i, j. \tag{30}$$

The equally weighted portfolio

$$\mathbf{w}_s^{(0)}(t_{tr}^-) = \{1/N, 1/N, \ldots, 1/N\} \tag{31}$$

is the initial portfolio (at time t_{tr}^-, before the first portfolio rebalancing) for all the strategies analyzed. From the results of additional exploratory experiments, one concludes that the effect of the composition of the initial portfolio in the final performance measures is small. Furthermore, the transient regime during which the initial composition of the portfolio has a significant effect on the evolution is fairly short-lived. For simplicity, equal transaction costs are assumed for all assets $\{\kappa_i = \kappa;\ i = 1, \ldots, N\}$. We then select a portfolio $\mathbf{w}^s(t_{tr})$ using each of the strategies considered. The composition of this portfolio is held fixed for the period $[t_{tr}, t_{tr} + 1)$. Even if the composition of the portfolio does not change, the portfolio weights evolve during this period because of changes in the market prices of its constituents. The training data window is then shifted by one month. The portfolio is rebalanced at $t_{tr} + 1$ using as inputs the expected means and covariance matrix of the asset returns estimated on the training data from the shifted time window. The process is repeated until the last period of data available.

Consider the portfolio selected by strategy s after rebalancing at time t. This portfolio is characterized by the vector of weights $\mathbf{w}^{(s)}(t) = \left\{w_i^{(s)}(t)\right\}_{i=1}^{N}$. As described in the section on in-sample evaluation, when no transaction costs are considered to select these weights (that is, for strategies $1/N$, MinVar, Lasso and IgnoreTC) the cost-adjusted return of the portfolio in the period $[t, t+1)$ is

$$r^{(s)}(t) \quad = \quad \frac{P(t)}{P(t^-)} \left(\sum_{i=1}^{N} w_i^{(s)}(t) r_i(t) - \left(\frac{P(t^-)}{P(t)} - 1 \right) \right),$$

where $r_i(t)$ are the actual returns for the ith asset in that period, $P(t^-)$ is the value of the portfolio before rebalancing at t, and $P(t)$ is the value of the portfolio after rebalancing.

When the portfolio weights $\mathbf{w}^{(s)}(t)$ are computed taking into account transaction costs (that is, for No Card. and eRAR), the portfolio return in $[t, t+1)$ is

$$r^{(s)}(t) = \sum_{i=1}^{N} w_i^{(s)}(t) r_i(t) - \sum_{i=1}^{N} \kappa_i \left| w_i^{(s)}(t) - w_i^{(0)}(t) \right|, \tag{32}$$

where $w_i^{(0)}(t)$ are the portfolio weights immediately before rebalancing at time t.

The accumulated return in the testing (out-of-sample) period $[t_{tr}+1, T]$ is

$$R_{acc}^{(s)}(t_{tr}+1, T) = \frac{\mathbb{E}\left[P(T)\right]}{P(t_{tr})} - 1 = \prod_{t=t_{tr}+1}^{T} (1 + r^{(s)}(t)) - 1. \tag{33}$$

The average Sharpe ratio is

$$S_{av}^{(s)}(t_{tr}+1, T) = \frac{\text{Av}\left[\left\{R^{(s)}(t)\right\}_{t=t_{tr}+1}^{T}\right]}{\text{Stdev}\left[\left\{R^{(s)}(t)\right\}_{t=t_{tr}+1}^{T}\right]}. \tag{34}$$

In this expression the numerator represents the sample average and the denominator the sample standard deviation of the time series of portfolio returns. To quantify the amount of trading that is performed, the average turnover in terms of normalized weights

$$T^s(t_{tr}+1, T) = \frac{1}{T - t_{tr} + 1} \sum_{t=t_{tr}}^{T-1} \sum_{i=1}^{N} \left| \frac{w_i^{(s)}(t+1)}{\sum_{j=1}^{N} w_j^{(s)}(t+1)} - \frac{w_i^{(s)}(t)}{\sum_{j=1}^{N} w_j^{(s)}(t)} \right| \tag{35}$$

is also computed.

In the experiments performed, 5 years of data are used for estimating the inputs to the optimization in each time window. The first training period is from June 1971 until June 1976. The first testing period is July 1976. The training period is then shifted one month, so that it includes data from July 1971 until July 1976. The return of the selected portfolio in August 1976 is then computed and stored. The rolling window experiment is repeated until the data are exhausted (December 2009). The performances of the portfolios selected by the genetic algorithm with set encoding, cardinality constraints and transaction costs are compared with the five benchmark portfolios described in the section on in-sample results. The parameters used in the GA optimizations are the same as those used for in-sample evaluation (Section 5.1). The eRAR-GA methods are run 15 times for each time window. We report also the results for the Passive strategy, in which the composition of the portfolio is held constant. Initially the N assets have the same weight $(1/N)$. Even though the composition of the portfolio does not change, the asset weights change because of the evolution of their market prices. It is interesting to benchmark against the passive strategy because it does not involve any rebalancing and therefore does not incur transaction costs. We have also carried out statistical tests to determine whether the differences in performance between the proposed strategy (eRAR with TC) and each of the other strategies are statistically significant. Differences in returns are tested applying the non-parametric Wilcoxon signed-rank test [54] directly to the paired portfolio return values. For the Sharpe ratios we use the method proposed by Ledoit and Wolf in [21] [4]. Differences that are

[4] The R code is available at `http://www.econ.uzh.ch/faculty/wolf/publications.html`.

statisitically significant at a 95% probability level are marked with an asterisk in the tables. The p-values are presented in a separate electronic companion for reference.

5.2.1. Dataset FF48

Table 8 displays the accumulated returns for the different strategies in the FF48 dataset. The corresponding average Sharpe ratios are shown in Table 9. The values reported correspond to an investment period from June 1976 until December 2009. The accumulated returns should therefore be interpreted as the accumulated profit (final minus initial portfolio values) at the end of December 2009 that results from an investment of 1€ at the beginning of June 1976. The results are calculated for transaction costs that range between 0 and 50 bps.

The first important observation is that the portfolio built using the eRAR strategy but without taking into account transaction costs when rebalancing the portfolio has in most cases a lower accumulated return than the same strategy with transaction costs, which has smaller values of the average turnover as well. In fact, this strategy has larger accumulated return than all the portfolios that are selected using strategies that ignore transaction costs. The minimum variance portfolios exhibit better performance than $1/N$ portfolios both in terms of accumulated return and of average Sharpe ratios. The passive portfolio has fairly high expected returns, larger than the standard mean-variance, the min-variance and the $1/N$ portfolios, and only slightly lower than the lasso portfolio. Another important observation is that including cardinality constraints generally improves the out-of-sample performance. Nonetheless, the cardinality constraint by themselves are not sufficient. One needs to take into account also transaction costs. The eRAR strategy that considers transaction costs has the best accumulated return in most cases.

The average values of the Sharpe ratios are presented in Table 9. In spite of the differences observed in the accumulated returns, all the portfolios, except for the ones selected by the passive and the $1/N$ strategies, have similar values of this performance measure. The reason is that portfolios without cardinality constraints generally have lower accumulated returns but also lower variances, as a result of diversification. The best values of the average Sharpe ratio correspond to the MinVar strategy. This suggests that minimizing the in-sample variance is an effective strategy to minimize the out-of-sample variance. The second largest values of the Sharpe ratios are achieved by the eRAR strategy with transaction costs.

The values of the average turnover of the different portfolios are shown in Table 10. The $1/N$ strategy has the largest overall turnover: The portfolio needs to be continuously rebalanced to compensate the changes in portfolio weights resulting from the changes in the market prices of the assets in the portfolio. This explains the poor performance of this portfolio when transaction costs are taken into account. The passive strategy, which does not involve any rebalancing, has zero turnover. The turnover for the lasso strategy, in which trades are penalized, is fairly small. As expected, in the strategies that take into account transaction costs in the selection of optimal portfolio weights (No Card.,

Table 8: Accumulated returns for the different strategies in the FF48 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3300$ bps	eRAR Ignore TC	eRAR With TC
0	95.670663	86.276413	89.350936*	86.813818*	97.189846	127.804675	**127.856728**
10		73.408415	88.132098*	83.336660*	97.104098	119.443385*	**125.431925**
20		62.436569	86.929599*	80.114928*	97.018307	111.703791	**111.723312**
30		53.081622*	85.743219*	77.145857*	96.932477	104.395984	**111.135888**
40		45.105426*	84.572742*	74.190147*	96.846614	97.428673	**104.769528**
50		38.304883*	83.417955	71.356740*	96.760665	91.039471	**99.743073**

Table 9: Average Sharpe ratios for the different strategies in the FF48 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3300$ bps	eRAR Ignore TC	eRAR With TC
0	0.263390	0.269102	**0.364336**	0.322772*	0.304987	0.345993	0.346015
10	0.263390*	0.260463*	**0.363313**	0.320258*	0.304931	0.341572	0.344822
20	0.263390	0.251818*	**0.362288**	0.317803*	0.304874	0.337197	0.339437
30	0.263390	0.243168*	**0.361262**	0.315423*	0.304818	0.332791	0.338209
40	0.263390	0.234514*	**0.360235**	0.312971*	0.304761	0.328241	0.334632
50	0.263390	0.225856*	**0.359207**	0.310449*	0.304704	0.323804	0.330771

eRAR with TC), the average turnover decreases with increasing transaction costs.

To further investigate the regularization effects of cardinality constraints and of lasso penalties we compare the out-of-sample accumulated returns for lasso portfolios with cardinality constraints $K = 10$, $K = 20$ and without a cardinality constraint. Figure 2 displays the accumulated return of these lasso portfolios as a function of the value of $\boldsymbol{\gamma}$ used in the optimization. Using either cardinality constraints or high lasso penalties ($\gamma \approx 2500$ bps) are useful strategies that can be used to select portfolios with good out-of-sample performance. However, using *both* cardinality constraints and a high lasso penalty seems to be detrimental for the out-of-sample performance. From these results we conclude that including both types of regularization is not an effective strategy in the problems investigated.

To illustrate the evolution of the porftolios that are selected when both transaction costs and cardinality constraints are considered, we present results in an investment universe of $N = 3$ assets. The portfolio is restricted to have

Table 10: Average turnover for the different strategies in the FF48 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3300$ bps	eRAR Ignore TC	eRAR With TC
0	0.000000	0.474748	0.044916	0.092204	0.002220	0.189553	0.189676
10				0.088368			0.140837
20				0.085043			0.123523
30				0.082032			0.106768
40				0.079307			0.101795
50				0.076869			0.093230

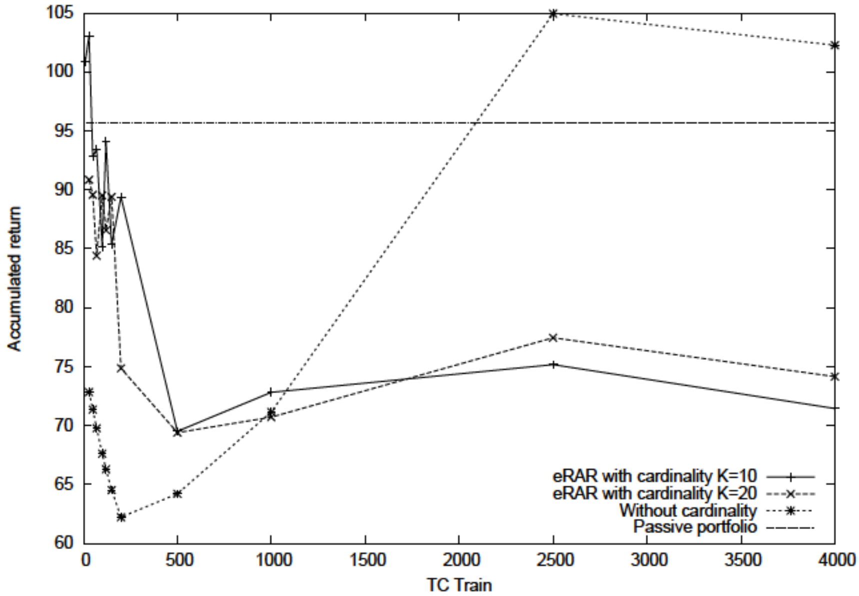


Figure 2: Accumulated returns as a function of the transaction costs used for training. Transaction costs in test are set to 50 bps.

at most $K \leq 2$ assets at a given instant. Figure 3 displays the evolution of the portfolio for transaction costs $\kappa = 0, 25, 50, 75, 150, 250$ bps. The parameters of the GA are the same as in the previous subsection. The best result of 5 executions is reported. Several features of the evolution of the investment are noteworthy: For low costs the assets included in the portfolio change often to take advantage of local trends. For larger transaction costs, the changes in the portfolio weights are smaller and the positions in a given asset are held longer. The composition is modified only if the trend that has been detected is sufficiently strong. In the experiments performed this is illustrated by the fact that the composition of the portfolios changes when the transaction costs are low (0-75 bps). In contrast, when the transaction costs are higher (150 and 250 bps), the portfolios invest in the same two assets during the whole investment period.

5.2.2. Dataset FF100

The out-of-sample performance measures for the FF100 dataset are presented in Tables 11, 12 and 13. Since the number of assets is much higher in this case, we use a larger cardinality constraint of $K = 25$. The best results are obtained by the lasso and the passive strategies. Portfolios that do not consider transaction costs have lower accumulated returns. The fact that these portfolios have lower variance (because they are more diversified) means that the average Sharpe ratios are only slightly lower. The performance achieved by portfolios that are selected by strategies that ignore transaction costs significantly deteri-

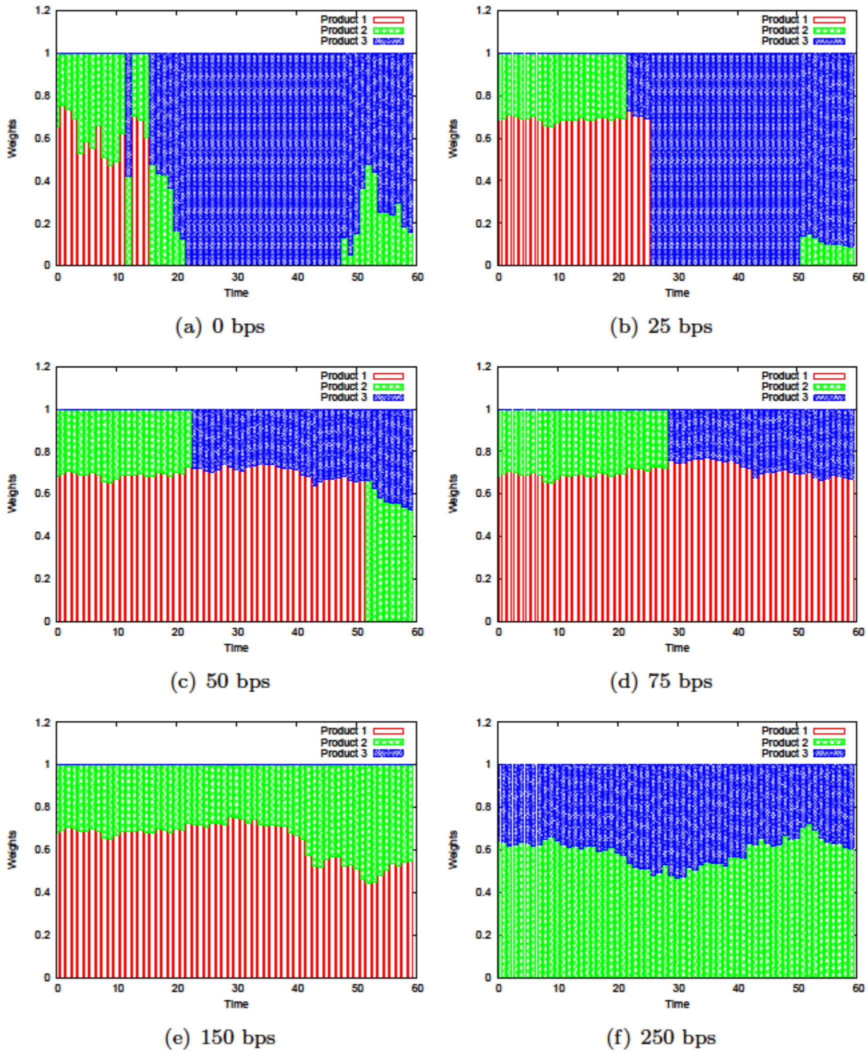


Figure 3: Evolution of portfolio weights obtained by eRAR for different transaction costs in a 3 asset universe.

Table 11: Accumulated returns for the different strategies in the FF100 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3550$ bps	eRAR Ignore TC	eRAR With TC
0	110.500376	48.913566	48.913595	58.043442	**116.864211**	65.425594	65.425594
10		41.973967	48.471808	55.467468	**116.820847**	57.817103*	61.665953
20		35.998316*	48.033867	53.010446*	**116.777430**	51.076857*	58.791746
30		30.852828*	47.599738	50.854126*	**116.733955**	45.106145*	60.410304
40		26.422274*	47.169388	49.061071*	**116.690428**	39.817437*	61.116005
50		22.607402*	46.742783	47.423537*	**116.646838**	35.133122*	62.641121

Table 12: Average Sharpe ratios for the different strategies in the FF100 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3550$ bps	eRAR Ignore TC	eRAR With TC
0	0.301013	0.253470	0.253470	0.276608	**0.320164**	0.278810	0.278810
10		0.244719	0.252947	0.273972	**0.320139**	0.271551	0.275028
20		0.235964	0.252424	0.271277	**0.320115**	0.264273*	0.272166
30		0.227206	0.251901	0.268755	**0.320091**	0.256978*	0.275201
40		0.218443	0.251377	0.266597	**0.320066**	0.249666*	0.276256
50		0.209678	0.250854	0.264584	**0.320042**	0.242338*	0.279815

orates with increasing transaction costs. In contrast, the eRAR portfolios with transaction costs exhibit good overall performance, although inferior to the passive or the lasso strategies. From these results one concludes that it is crucial to take into account the effects of transaction costs in the optimization.

5.2.3. Dataset FF38

Out-of-sample performance measures for the dataset FF38 are given in Tables 14, 15 and 16. The conclusions are similar to those obtained from the analysis of the results in the FF100 dataset. The lasso strategy is the best strategy in terms of accumulated out-of-sample returns. The portfolio selected by eRAR with transaction costs also has large accumulated returns, although they are lower than the lasso. As in the previous cases, eliminating the cardinality constraint or ignoring transaction costs leads to the selection of unstable portfolios that have lower accumulated returns. In contrast, the average Sharpe ratios do not exhibit this effect. As a matter of fact, the minimum variance portfolio has the best average Sharpe ratio. This is because these types of portfolio

Table 13: Average turnover for the different strategies in the FF100 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 3550$ bps	eRAR Ignore TC	eRAR With TC
0	0.000000	0.444945	0.024699	0.122999	0.001260	0.335823	0.335823
10				0.116488			0.256265
20				0.110776			0.213480
30				0.105754			0.186941
40				0.101169			0.162803
50				0.097206			0.149543

Table 14: Accumulated returns for the different strategies in the FF38 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 2610$ bps	eRAR Ignore TC	eRAR With TC
0	85.489189	88.216696	83.930429	68.433229*	**102.642278**	100.823671	100.823671
10	85.489189	76.294301	82.615493	65.320255*	**102.530729**	95.403599	96.335757
20	85.489189	65.965154	81.320665	62.357596*	**102.419171**	90.270322	91.046090
30	85.489189	57.016317	80.045640	59.576223*	**102.307587**	85.408754	86.152814
40	85.489189	49.263322	78.790119	57.023553*	**102.196015**	80.804599	85.493012
50	85.489189	42.546352	77.553806	54.635601*	**102.084424**	76.444307	80.215254

Table 15: Average Sharpe ratios for the different strategies in the FF38 dataset.

TC	Passive No TC	$1/N$ Ignore TC	MinVar Ignore TC	No Card. With TC	Lasso $\gamma = 2610$ bps	eRAR Ignore TC	eRAR With TC
0	0.257746*	0.269579	**0.384662**	0.317762*	0.314303	0.342412	0.342412
10	0.257746	0.261837*	**0.383389**	0.314790*	0.314234	0.338644	0.338829
20	0.257746*	0.254092*	**0.382114**	0.311862*	0.314165	0.334869	0.334972
30	0.257746	0.246343*	**0.380837**	0.308960*	0.314095	0.331087	0.330807
40	0.257746	0.238591*	**0.379557**	0.306183*	0.314025	0.327297	0.331758
50	0.257746	0.230837*	**0.378275**	0.303434*	0.313955	0.323500	0.326053

are more diversified, and, in consequence, tend to have a lower variance.

5.3. Discussion

When transaction costs are not considered, the naïvely diversified ($1/N$) portfolio has good out-of-sample performance ([29]). However, the performance of the $1/N$ portfolio quickly deteriorates when transaction costs are considered. The reason is that a very active trading strategy is needed to compensate for the changes in portfolio weights that result from the evolution of the market prices of the assets in the portfolio. This strategy has large turnover rates and incurs high transaction costs. A better benchmark when transaction costs are considered is the passive strategy. Since no rebalancing is performed, one does not incur any transaction costs. Asymptotically, for long investment periods, the portfolio is dominated by the best performing assets. This means that, in practice, the expected return from this investment is large. However, the variance of the portfolio returns is also large because of the lack of diversification.

Table 16: Average turnover for the different strategies in the FF38 dataset.

<table>
<tr><th>TC</th><th>Passive
No TC</th><th>$1/N$
Ignore TC</th><th>MinVar
Ignore TC</th><th>No Card.
With TC</th><th>Lasso
$\gamma = 2610$ bps</th><th>eRAR
Ignore TC</th><th>eRAR
With TC</th></tr>
<tr><td>0</td><td rowspan="6">0.000000</td><td rowspan="6">0.420368</td><td rowspan="6">0.050512</td><td>0.059326</td><td rowspan="6">0.002825</td><td rowspan="6">0.160068</td><td>0.160068</td></tr>
<tr><td>10</td><td>0.047544</td><td>0.120016</td></tr>
<tr><td>20</td><td>0.038947</td><td>0.108162</td></tr>
<tr><td>30</td><td>0.032244</td><td>0.101079</td></tr>
<tr><td>40</td><td>0.027461</td><td>0.092896</td></tr>
<tr><td>50</td><td>0.023389</td><td>0.087525</td></tr>
</table>

From the results of the empirical study carried out, the observation that in-sample performance is not necessarily a good estimate of the out-of-sample performance is confirmed. To obtain good out-of-sample performance one needs to include some form of regularization in the optimization. This regularization can be in the form of terms in the objective function that penalize excessive portfolio rebalancing in response to spurious trends in the training data, or of cardinality constraints. Exploratory experiments show that including both types of regularization does not seem to be an effective strategy. Nonetheless, a more extensive evaluation should be carried out to provide further evidence of this observation. Besides the passive strategy, the best out-of-sample returns are obtained by portfolios that are built using regularization: The lasso strategy and the eRAR strategy that takes into account the actual transaction costs and considers cardinality constraints as well. In terms of Sharpe ratios, the differences between regularized and non-regularized strategies are smaller. In particular, the average Sharpe ratios of minimum variance portfolios are generally among the best. These portfolios are well diversified and, in general, the variance of the out-of-sample returns is small. The standard mean-variance optimal portfolio, which has excellent in-sample performance has poor out-of-sample performance in all the cases investigated. This can be ascribed to some form of overfitting to the training data [29].

6. Conclusions and future work

In this work, a novel memetic algorithm is introduced to address the problem of optimal portfolio selection with piecewise linear transaction costs, cardinality constraints and minimum trading size restrictions. The algorithm is based on handling the combinatorial and the continuous aspects of the optimization separately.

The combinatorial problem of selecting the subset of assets of the specified cardinality in which the portfolio invests is addressed by a genetic algorithm in which candidate solutions are represented as extended sets. The elements in the set correspond to the assets included in the portfolio. For each asset, the chromosome considers an additional attribute that indicates the type of trade ('buy','hold','sell') that needs to be made in the portfolio rebalancing. The exploration and exploitation capabilities of the algorithm are enforced using specially designed crossover and mutation operators that take advantage of the specific features of the problem. For the selection process, the candidate solutions are evaluated by solving the portfolio optimization problem defined in the restricted investment universe under the conditions specified by the corresponding chromosome. This is a quadratic programming problem that can be solved by a standard quadratic optimization algorithm.

One of the main contributions of this research is the adaptation of the RAR crossover operator to manipulate the additional attributes in the genetic representation that specify the direction of the trades in each asset during rebalacing. This extended RAR crossover (eRAR) allows to handle the transaction costs, cardinality constraints and minimum trading size restrictions in such a way that the offspring generated are always feasible.

One possibility to improve the results of the proposed memetic approach is to implement other set-based crossover operators that exploit the structure of the problem. In particular, a novel crossover operator based on sets was proposed in [43]. This algorithm can be adapted to work with extended sets in a similar fashion as RAR. Additionally, dimensionality-reduction techniques similar to the ones proposed in [42] could be applied to this problem.

Analyzing the results of the extensive empirical evaluation performed, we conclude that it is important to incorporate transaction costs explicitly in the optimization to obtain portfolios that have good in-sample, but specially out-of-sample performance. Another conclusion of this study is that cardinality constraints can also improve the out-of-sample performance of the portfolio. In general, portfolios with cardinality constraints have better out-of-sample performance than portfolios that invest in all assets. In summary, both transaction costs and cardinality constraints can be seen as regularization strategies that allow the identification of stable and robust portfolios with good out-of-sample performance.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

[1] M.J. Best and J. Hlouskova. Quadratic programming with transaction costs. *Computers and Operations Research*, 25(35):18–33, 2008.

[2] Daniele Bianchi and Massimo Guidolin. Can long-run dynamic optimal strategies outperform fixed-mix portfolios? Evidence from multiple data sets. *European Journal of Operational Research*, 236(1):160–176, 2014.

[3] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[4] Stephen P. Boyd, Mark T. Mueller, Brendan O'Donoghue, and Yang Wang. Performance bounds and suboptimal policies for multi-period investment. *Foundations and Trends in Optimization*, 1(1):1–72, 2014.

[5] A. Brabazon and M. O'Neill. *Biologically Inspired Algorithms for Financial Modelling.* Springer-Verlag, Berlin, 2006.

[6] Michael W. Brandt. *Portfolio choice problems.* Handbook of Financial Econometrics, Volume 1: Tools and Techniques. North Holland, 2010.

[7] J. Brodie, I. Daubechies, C. De Mol, D. Giannone, and I. Loris. Sparse and stable Markowitz portfolios. *Proceedings of the National Academy of Sciences*, 106(30):12267–12272, July 2009.

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

[9] Y. Crama and M. Schyns. Simulated annealing for complex portfolio selection problems. Technical report, Groupe d'Etude des Mathematiques du Management et de l'Economie 9911, Universie de Liege., 1999.

[10] Bart Diris, Franz Palm, and Peter Schotman. Long-term strategic asset allocation: An out-of-sample evaluation. *Management Science*, accepted for publication.

[11] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. Inertia-controlling methods for general quadratic programming. *SIAM Review*, 33:1–36, 1991.

[12] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13:533–549, 1986.

[13] A.K. Hartmann and H. Rieger. *Optimization Algorithms in Physics*. Wiley-VCH, New York, 2002.

[14] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[15] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[16] R. Jagannathan and T. Ma. Risk reduction in large portfolios: Why imposing the wrong constraint helps. *The Journal of Finance*, 58(4):1651–1683, 2003.

[17] M. V. Jahan and M. R. Akbarzadeh-Totonchi. From local search to global conclusions: Migrating spin glass-based distributed portfolio selection. *IEEE Transactions on Evolutionary Computation*, 14(4):591–601, 2010.

[18] R. Jeurissen and J. van den Berg. Optimized index tracking using a hybrid genetic algorithm. In *Proceedings of the IEEE World Congress on Evolutionary Computation (CEC2008)*, pages 2327–2334, 2008.

[19] S. Kirkpatrick, C. D. Gelatt, and Jr. M. P. Vecchi. Optimization by simulated annealing. *Science*, 4598:671–679, 1983.

[20] T. Krink and S. Paterlini. Multiobjective optimization using differential evolution for real-world portfolio optimization. *Comput. Manag. Sci.*, 8:157–179, 2011.

[21] O. Ledoit and M. Wolf. Robust performance hypothesis testing with the Sharpe ratio. *Journal of Empirical Finance*, 15:850–859, 2008.

[22] R. Li, M. Emmerich, J. Eggermont, T. Bäck, M. Schütz, J. Dijkstra, and J.H.C. Reiber. Mixed integer evolution strategies for parameter optimization. *Evolutionary Computation*, 21(1):29–64, 2013.

[23] M. Lobo, M. Fazel, and S. Boyd. Portfolio optimization with linear and fixed transaction costs. *Annals of Operations Research, special issue on financial optimization*, 152 (1):376–394, July 2007.

[24] K. Lwin, R. Qu, and G. Kendall. A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization. *Applied Soft Computing*, 2014.

[25] H. Markowitz. Portfolio selection. *Journal of Finance*, 7:77–91, 1952.

[26] H. Markowitz. *Mean-variance analysis in portfolio choice and capital markets*. Basil Blackwell, 1987.

[27] Robert C Merton. Lifetime Portfolio Selection under Uncertainty: The Continuous-Time Case. *The Review of Economics and Statistics*, 51(3):247–57, August 1969.

[28] V. De Miguel, L. Garlappi, F. J. Nogales, and R. Uppal. A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms. *Management Science*, 55(5):798–812, 2009.

[29] V. De Miguel, L. Garlappi, and R. Uppal. Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy? *Review of Financial Studies*, 5(22):1915–1953, 2009.

[30] J.E. Mitchell and S. Braun. Rebalancing an investment portfolio in the presence of convex transaction costs. Technical report, Department of Mathematical Sciences, Rensselaer Polytechnic Institute, 2004.

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

[32] P. Moscato and C. Cotta. A gentle introduction to memetic algorithms. In *Handbook of Metaheuristics*, pages 105–144. Kluwer Academic Publishers, 2003.

[33] A. Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review*, 40(3):636–666, 1998.

[34] P. O'Sullivan and D. Edelman. Adaptive universal portfolios. *The European Journal of Finance*, 2013.

[35] Ľuboš Pástor and Robert F. Stambauc. Are stocks really less volatile in the long run? *The Journal of Finance*, 67(2):431–478, 2012.

[36] A.F. Perold. Large-scale portfolio optimization. *Management Science*, 30(10):1143–1160, 1984.

[37] G.A. Pogue. An extension of the Markowitz portfolio selection model to include variable transactions' costs, short sales, leverage policies and taxes. *The Journal of Finance*, 25(5):1005–1027, 1970.

[38] N. J. Radcliffe. Genetic set recombination. In *Foundations of Genetic Algorithms*. Morgan Kaufmann Pulishers, 1993.

[39] G. R. Raidl. A unified view on hybrid metaheuristics. In *Proceedings of the Hybrid Metaheuristics Workshop*, volume 4030 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2006.

[40] R. Ruiz-Torrubiano. *Cardinality constraints and dimensionality reduction in optimization problems.* PhD thesis, Computer Science Department, Universidad Autónoma de Madrid, 2012.

[41] R. Ruiz-Torrubiano, S. García-Moratilla, and A. Suárez. Optimization problems with cardinality constraints. In *Computational Intelligence in Optimization: Implementations and Applications*. Springer Verlag, 2010.

[42] R. Ruiz-Torrubiano and A. Suarez. Hybrid approaches and dimensionality reduction for portfolio selection with cardinality constraints. *IEEE Computational Intelligence Magazine*, 5(2):92–107, 2010.

[43] R. Ruiz-Torrubiano and A. Suárez. The transrar crossover operator for genetic algorithms with set encoding. In *Proceedings of the 13th annual conference on Genetic and Evolutionary Computation (GECCO 2011), Dublin, Ireland*, pages 489–496, 2011.

[44] Paul A Samuelson. Lifetime Portfolio Selection by Dynamic Stochastic Programming. *The Review of Economics and Statistics*, 51(3):239–46, August 1969.

[45] H.P. Schwefel. *Evolution and Optimum Seeking: The Sixth Generation.* John Wiley & Sons, Inc. New York, 1993.

[46] J. Shapcott. Index tracking: genetic algorithms for investment portfolio selection. Technical report, EPCC-SS92-24, Edinburgh, Parallel Computing Centre, 1992.

[47] W. F. Sharpe. The Sharpe ratio. *The Journal of Portfolio Management*, 21(1):49–58, 1994.

[48] R. Storn and K. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.

[49] F. Streichert and M. Tamaka-Tamawaki. The effect of local search on the constrained portfolio selection problem. In *Proceedings of the IEEE World Congress on Evolutionary Computation (CEC2006)*, pages 2368–2374, Vancouver, Canada, 16-21 July 2006.

[50] F. Streichert, H. Ulmer, and A. Zell. Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC2004)*, volume 1, pages 932–939, 2004.

[51] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, 58(1):267–268, 1996.

[52] A. N. Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Soviet Mathematics Doklady*, 4:1035–1038, 1963.

[53] G. A. Vijayalakshmi and T. Michel. Evolutionary optimization of constrained k-means clustered assets for diversification in small portfolios. *IEEE Transactions on Evolutionary Computation*, 13(5):1030–1053, 2009.

[54] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

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

[56] A. Yoshimoto. The mean-variance approach to portfolio optimization subject to transaction costs. *Journal of the Operations Research Society of Japan*, 39(1):99–117, 1996.

Appendix

In this Appendix we provide a detailed derivation of the expressions of the expected return (1) and risk (2) of the portfolio when a self-financing constraint is imposed and the effects of piecewise linear transaction costs are considered.

Consider the problem of managing a portfolio that invests in a universe of N assets and can be rebalanced at times $t = 1, \ldots, T$. Let $\left\{\mathbf{S}(t) = \{S_i(t)\}_{i=1}^N\right\}_{t=1}^T$ be the time series of the price of the assets. The composition of the portfolio in the interval $[t-1, t)$ is given by the column vector

$$\mathbf{x}(t-1) = \{x_i(t-1)\}_{i=1}^N, \tag{36}$$

where $x_i(t-1)$ denotes the number of shares of asset i held in the portfolio. The amount of capital invested in asset i at time τ in the interval $t-1 \leq \tau < t$ is $x_i(t-1)S_i(\tau)$. The composition of the portfolio $\mathbf{x}(t-1)$ is held constant in the interval $[t-1, t)$. However, as a result of the evolution of the market prices of the assets the total value of the portfolio changes with time

$$P(\tau) = \sum_{i=1}^{N} x_i(t-1) S_i(\tau), \quad t-1 \leq \tau < t. \tag{37}$$

Let

$$P(t^-) = \sum_{i=1}^{N} x_i(t-1) S_i(t). \tag{38}$$

be the value of the portfolio at t^-, the time immediately *before* rebalancing at the end of the interval $[t-1, t)$. At time t the portfolio is rebalanced with the goal of maximizing the expected return in the interval $[t, t+1)$, which is how the profit is measured, while minimizing the corresponding variance, which is taken as a measure of the risk of the investment. The new portfolio has a different composition $\mathbf{x}(t) = \{x_i(t)\}_{i=1}^N$ that is held constant during the period $[t, t+1)$. Its value in this interval is

$$P(\tau) = \sum_{i=1}^{N} x_i(t) S_i(\tau), \quad t \leq \tau < t+1. \tag{39}$$

An alternative way of specifying the composition of the portfolio is to use the vector of weights $\mathbf{w}(\tau) = \{w_i(\tau)\}_{i=1}^N$. The components of this vector are the fraction of $P(t^-)$, the wealth available for investment at time t^-, allocated to each of the assets

$$w_i(\tau) = \frac{x_i(t) S_i(\tau)}{P(t^-)} = \frac{x_i(t) S_i(\tau)}{\sum_{j=1}^{N} x_j(t-1) S_j(t)}, \quad t \leq \tau < t+1, \quad i = 1, \ldots, N. \tag{40}$$

These weights satisfy the inequality constraint

$$\sum_{i=1}^{N} w_i(t) \leq 1, \tag{41}$$

with equality only if the transaction costs are zero.

The rebalancing is made with the restriction that the portfolio is self-financing

$$P(t^-) = P(t) + \Phi(\mathbf{x}, t), \tag{42}$$

where $P(t^-)$ is the value of the portfolio before rebalancing, $P(t)$ the value of the portfolio after rebalancing and $\Phi(\mathbf{x}, t)$ are the costs incurred in the transactions that are needed to rebalance the portfolio. In this work we assume piecewise linear transaction costs

$$\Phi(\mathbf{x}, t) = \sum_{i=1}^{N} \kappa_i \left| x_i(t) S_i(t) - x_i(t-1) S_i(t) \right|, \tag{43}$$

where κ_i is the fee associated with buying or selling one euro worth of asset i. The generalization of (43) to consider different selling and buying costs is straightforward. Note that the linearity assumption typically holds for small transactions. For larger transactions liquidity costs must be considered, and the transaction costs generally become non-linear. Nevertheless, these can be approximated by a piecewise linear function [37].

Using the explicit form of the transaction costs (43) the self-balancing constraint (42) becomes

$$P(t^-) = \sum_{i=1}^{N} x_i(t)S_i(t) + \sum_{i=1}^{N} \kappa_i \left| x_i(t)S_i(t) - x_i(t-1)S_i(t) \right|. \tag{44}$$

Dividing both sides by $P(t^-)$ one obtains

$$1 = \frac{\sum_{i=1}^{N} x_i(t)S_i(t)}{\sum_{j=1}^{N} x_j(t-1)S_j(t)} + \sum_{i=1}^{N} \kappa_i \left| \frac{x_i(t)S_i(t)}{\sum_{j=1}^{N} x_j(t-1)S_j(t)} - \frac{x_i(t-1)S_i(t)}{\sum_{j=1}^{N} x_j(t-1)S_j(t)} \right|. \tag{45}$$

Rewriting this expression in terms of $\mathbf{w}(t) = \{w_i(t)\}_{i=1}^{N}$ the self-financing constraint becomes

$$\sum_{i=1}^{N} w_i(t) + \sum_{i=1}^{N} \kappa_i \left| w_i(t) - w_i^{(0)}(t) \right| = 1, \tag{46}$$

where the vector of weights immediately before rebalancing is

$$\begin{aligned} w_i^{(0)}(t) &\equiv \frac{x_i(t-1)S_i(t)}{\sum_{j=1}^{N} x_j(t-1)S_j(t)}, \quad i = 1, \ldots, N, \\ \sum_{i=1}^{N} w_i^{(0)}(t) &= 1. \end{aligned} \tag{47}$$

The goal of the optimization is to minimize the risk of the portfolio and maximize the expected return. In terms of the returns of the individual assets in the period $[t, t+1)$

$$r_i(t) = \frac{S_i(t+1)}{S_i(t)} - 1, \quad i = 1, \ldots, N, \tag{48}$$

the return of the portfolio in that interval is

$$\begin{aligned} r_P(t) &= \frac{P((t+1)^-)}{P(t^-)} - 1 = \frac{\sum_{i=1}^{N} x_i(t)S_i(t+1)}{\sum_{i=1}^{N} x_i(t-1)S_i(t)} - 1 = \sum_{i=1}^{N} w_i(t) \frac{S_i(t+1)}{S_i(t)} - 1 \\ &= \sum_{i=1}^{N} w_i(t)r_i(t) + \sum_{i=1}^{N} w_i(t) - 1 = \sum_{i=1}^{N} w_i(t)r_i(t) - \sum_{i=1}^{N} \kappa_i \left| w_i(t) - w_i^{(0)}(t) \right|. \end{aligned} \tag{49}$$

The expected value of the portfolio return is

$$\mathbb{E}\left[r_P(t)\right] = \sum_{i=1}^{N} w_i(t)\hat{r}_i - \sum_{i=1}^{N} \kappa_i \left| w_i(t) - w_i^{(0)}(t) \right|, \tag{50}$$

where $\{\hat{r}_i = \mathbb{E}\left[r_i(t)\right]\}_{i=1}^{N}$ are the expected returns of the individual assets, which in the Markowitz framework are assumed to be constant inputs to the model.

The risk of the investment is quantified in terms of the variance of the portfolio

$$\mathrm{Var}\left[r_P(t)\right] = \mathbf{w}^{\mathrm{T}}(t) \cdot \mathbf{\Sigma} \cdot \mathbf{w}(t), \tag{51}$$

where $\mathbf{\Sigma}$ is the $N \times N$ covariance matrix of the asset returns.