Optimization of Index-Based Portfolios

INAUGURALDISSERTATION

zur Erlangung der Würde eines Doctor rerum oeconomicarum
der Wirtschafts- und Sozialwissenschaftlichen Fakultät der Universität Bern

Oliver Strub

Betreuer: Prof. Dr. Norbert Trautmann
Professur für Quantitative Methoden der BWL
Departement Betriebswirtschaftslehre
Schützenmattstrasse 14, 3012 Bern

Bern, März 2018

Die Fakultät hat diese Arbeit am 24. Mai 2018 auf Antrag der beiden Gutachter Prof. Dr. Renata Mansini und Prof. Dr. Norbert Trautmann als Dissertation angenommen, ohne damit zu den darin ausgesprochenen Auffassungen Stellung nehmen zu wollen.

Contents

Introduction

An investment fund is a financial vehicle that pools capital collected from different investors. A professional portfolio manager invests the collected capital on behalf of the investors in a portfolio of assets, e.g., stocks or bonds. A financial index, such as the Standard & Poor's 500 index, reflects the overall development of the assets that constitute that index. Such indices often serve as benchmarks for evaluating the performance of passive and active investment funds. The objective of a passive investment fund, also called index-tracking fund, is to track the return of a given benchmark index as closely as possible, i.e., to obtain a low tracking error, whereas the objective of an active investment fund is to achieve an excess return over a benchmark index, i.e., to obtain an outperformance. Passive and active investment funds are economically important; over US-$ 16 trillion in net assets were managed through such funds in the US at the end of the year 2016[1]. In that year, a total of over 400 new investment funds entered and over 600 funds exited the US investment fund industry; these figures indicate the strong competition in this industry.

To be competitive, the portfolio manager of a passive or an active fund needs to determine how to form the fund's portfolio to achieve a low tracking error or a high outperformance, respectively, without violating a set of restrictions on the composition of the portfolio that are imposed by regulations, investment guidelines, stock exchanges, or the investors. These restrictions include, e.g., a maximum number of assets that may be included in the portfolio to limit the management costs, a prescribed range for the weight of each asset in the portfolio, and a maximum amount that can be spent for fixed and proportional transaction costs. Hence, the portfolio manager faces various decisions, such as whether an asset is selected for inclusion in the portfolio and which amount is invested in a selected asset. The problem of determining the best feasible index-based portfolio, i.e., a portfolio of a passive or active fund that satisfies all the considered restrictions and achieves the lowest possible tracking error or the highest possible outperformance, respectively, constitutes a challenging optimization problem.

[1] Investment Company Institute Fact Book (2017)

To optimize an index-based portfolio, exact and heuristic optimization methods can be applied. Exact methods include algorithms that can be applied to a formulation of the optimization problem as a mathematical program. Such exact methods guarantee to find the best feasible portfolio, but they often require a substantial computational time that may be prohibitive in practice. By contrast, heuristic optimization methods aim to find a good, but not necessarily the best, feasible portfolio in a short computational time.

This thesis consists of three papers on the optimization of index-based portfolios. In each paper, we consider a different type of investment fund, and develop a novel optimization method to determine better portfolios in terms of the tracking error or the outperformance within a limited computational time compared with the state-of-the-art optimization methods from the literature.

In the first paper, we consider index-tracking funds as described above. We formulate the optimization problem of determining the portfolio for such an index-tracking fund as a mathematical program. This mathematical program includes a novel optimization criterion that considers the trade-off between transaction costs and tracking error. Based on this novel mathematical programming formulation, a commercial solver, which is based on an exact method, can be applied to determine a portfolio. In a computational experiment based on a set of real-world problem instances, the portfolios determined by the novel mathematical programming formulation achieved a considerably lower tracking error within a limited computational time than the portfolios determined by two state-of-the-art mathematical programming formulations.

In the second paper, we consider a specific type of investment funds that are regulated by the European Union. These funds are known as Undertakings for Collective Investments in Transferable Securities (UCITS). UCITS funds have become increasingly popular, resulting in a total corresponding amount of assets under management that is comparable to the scale of the US investment fund industry. UCITS funds are subject to regulatory restrictions imposed by the UCITS directive of the European Parliament. These restrictions include a short-selling prohibition and the 5/10/40 concentration rule. This concentration rule states that the weight of each selected asset must not exceed a lower threshold of 5%, except that the weights of some assets may be increased up to a middle threshold of 10%, provided that the sum of the weights exceeding the lower threshold does not exceed an upper threshold of 40%. We formulate the optimization problem of determining the portfolio for a UCITS fund that aims to track the return of an index as a mathematical program. Furthermore, we present a two-stage optimization method for this optimization problem. In the first stage, we apply a genetic algorithm, which treats subsets of the assets that are available for investment as individuals, to construct a good feasible portfolio in a short computational time. In this genetic algorithm, we use a new representation of subsets, which is the first to exhibit all of the following four desirable properties that enhance the effectiveness of genetic algorithms: feasibility, efficiency, locality, and heritability. In the second stage, we apply a local-search heuristic to improve the best feasible portfolio obtained in the first stage. In a computational experiment based on real-world data, the two-stage optimization method yields better feasible portfolios in terms of the tracking error compared with an exact method based on the new mathematical programming formulation.

In the third paper, we consider a novel type of investment funds, so-called enhanced index-tracking funds. Enhanced index-tracking funds aim to track the return of an index as closely as possible while outperforming that index by a small positive excess return. These funds are attractive to investors, especially when the index is large and thus well diversified. We deal with the optimization problem of determining a portfolio for an enhanced index-tracking fund that is benchmarked against a large regional or global index. For this optimization problem, we present two matheuristic approaches based on a novel mathematical programming formulation. Matheuristics are a recent type of optimization methods that combine exact and heuristic methods. We applied both matheuristics in a computational experiment to a set of novel problem instances that are based on large regional and global indices with up to more than 9,000 stocks. The results of this computational experiment indicate that, within a limited computational time, both matheuristics yield better feasible portfolios than other optimization methods do in terms of the tracking error and the excess return.

Although the optimization methods presented in this thesis were developed for the optimization of index-based portfolios, they can be applied, with only minor modifications, to other optimization problems that involve the selection of a small number of elements from a larger set. For example, the presented optimization methods could easily be adapted to the feature-selection problem that arises in machine learning and consists of selecting a small subset of relevant features to construct the best possible prediction or classification model.

Paper I

Optimal construction and rebalancing of index-tracking portfolios [2]

Oliver Strub Philipp Baumann

Department of Business Administration
University of Bern

Contents

[2]Published in European Journal of Operational Research Vol. 264, No. 1, 370–387, (DOI:10.1016/j.ejor.2017.06.055)

Abstract

Index funds aim to track the performance of a financial index, such as, e.g., the Standard & Poor's 500 index. Index funds have become popular because they offer attractive risk-return profiles at low costs. The index-tracking problem considered in this paper consists of rebalancing the composition of the index fund's tracking portfolio in response to new market information and cash deposits and withdrawals from investors such that the index fund's tracking accuracy is maximized. In a frictionless market, maximum tracking accuracy is achieved by investing the index fund's entire capital in a tracking portfolio that has the same normalized value development as the index. In the presence of transaction costs, which reduce the fund's capital, one has to manage the trade-off between transaction costs and similarity in terms of normalized value developments. Existing mathematical programming formulations for the index-tracking problem do not optimize this trade-off explicitly, which may result in substantial transaction costs or tracking portfolios that differ considerably from the index in terms of normalized value development. In this paper, we present a mixed-integer linear programming formulation with a novel optimization criterion that directly considers the trade-off between transaction costs and similarity in terms of normalized value developments. In an experiment based on a set of real-world problem instances, the proposed formulation achieves a considerably higher tracking accuracy than state-of-the-art formulations.

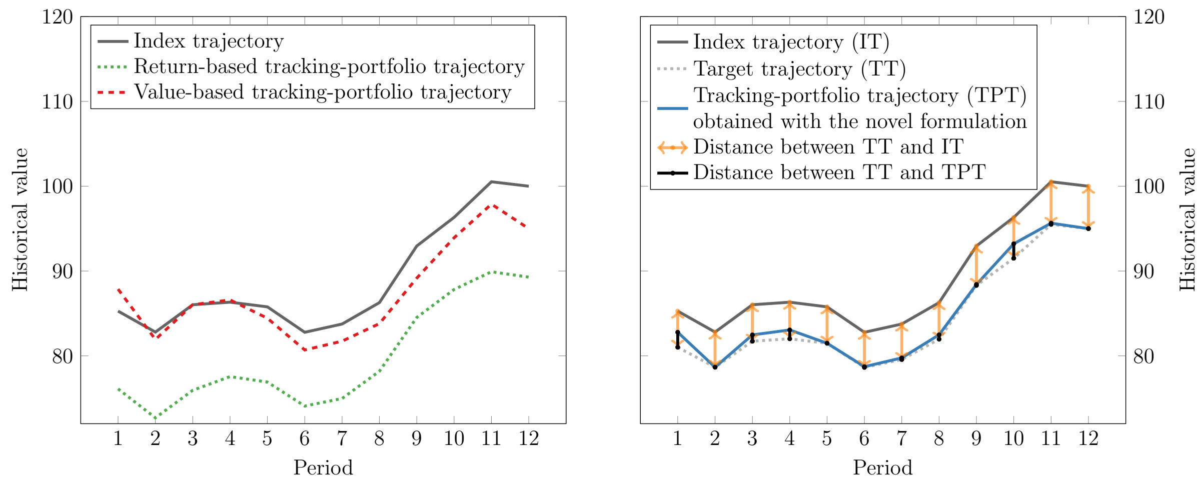
1.1 Introduction

An investment fund is a professionally managed investment vehicle that pools funds from different investors to invest in a portfolio of securities such as stocks or bonds. Over the past decade, a specific type of investment fund, so-called index funds, have become increasingly popular. At the end of 2015, over US-$2 trillion were invested in US-based index funds (cf. Investment Company Institue, 2016). Index funds aim to track the future value development over a given investment horizon of a benchmark portfolio, which represents an investment of the fund's entire capital in a financial index, such as the Standard & Poor's 500 index. Index funds require less investment research and often yield higher net returns compared to the investment funds that aim to outperform the benchmark portfolio (cf., e.g., Busse et al., 2010; Montfort et al., 2008; Malkiel, 1995).

The investors of an index fund demand a high tracking accuracy, i.e., small deviations between the future value developments of the benchmark portfolio and the index fund. In a frictionless market, the tracking accuracy is maximized when the index fund invests its entire capital in a tracking portfolio that has the same normalized shape as the index. The normalized shape refers to the shape of the value development that is normalized to a value of one today. The most intuitive index-tracking approach is to invest in all index constituents according to the index composition, which is known as full replication. Full replication ensures that the normalized shapes of the tracking portfolio and the index coincide. However, in the presence of market frictions, full replication causes a substantial amount of transaction costs that reduce the fund's capital, especially for indices with many constituents (cf., e.g., Beasley et al., 2003). This reduction of the fund's capital negatively impacts the tracking accuracy. Often, a higher tracking accuracy can be obtained by balancing the trade-off between the transaction costs and the similarity in terms of normalized shape.

The index-tracking problem considered in this paper consists of revising (rebalancing in the following) the composition of the index fund's tracking portfolio in response to new market information and cash changes that occur due to deposits and withdrawals from investors. Purchasing or selling assets to rebalance the tracking portfolio causes fixed and proportional transaction costs. The objective is to maximize the tracking accuracy subject to various practical portfolio constraints, including minimum transaction values that are imposed by many stock exchanges and short-selling restrictions that prohibit selling securities not currently owned by the fund. The initial construction of the tracking portfolio is a special case of the index-tracking problem in which the existing tracking portfolio consists of cash only. In practice, index-fund managers solve the index-tracking problem periodically over the fund's lifetime, which allows to account for changed market environments and cash changes.

Various mathematical programming formulations for the index-tracking problem have been proposed in the literature. These formulations determine a tracking portfolio by minimizing an objective function, referred to as tracking error, that measures the difference between the historical performance of the tracking portfolio and the index. The underlying idea is that, even though a low tracking error in the past does not guarantee a high tracking accuracy in the future, tracking portfolios with a lower tracking error tend to have a higher tracking accuracy in the future. Based on the specific tracking-error function used, existing formulations are divided into the two classes value-based formulations and return-based formulations (cf. Gaivoronski et al., 2005). The value-based formulations measure the tracking error as the distance between the historical trajectories (value developments) of the tracking portfolio and the index (cf., e.g., Konno and Wijayanayake, 2001; Guastaroba and Speranza, 2012). The values of the historical index trajectory correspond to the values of the index scaled by a constant factor such that the value at the end of the last historical period is equal to the capital of the index fund. The end of the last historical period corresponds to the point in time when the portfolio is rebalanced. The return-based formulations measure the tracking error as a function of the differences between the historical returns of the tracking portfolio and the index (cf., e.g., Andriosopoulos and Nomikos, 2014). Intuitively, minimizing this return-based tracking error can be viewed as maximizing the similarity between the normalized shapes of the historical index trajectory and the historical tracking-portfolio trajectory. In the presence of transaction costs, both classes of formulations determine tracking portfolios with an unsatisfactory tracking accuracy. Return-based formulations tend to determine tracking portfolios whose historical normalized shape is similar to the historical normalized shape of the index, but whose historical trajectory lies considerably below the historical index trajectory because the distance between the trajectories is not penalized in the objective function (cf. Figure 1.1a). Value-based formulations tend to determine tracking portfolios whose historical trajectories are overall closer to the historical trajectory of the index, but whose normalized shape differs considerably from the normalized shape of the index (cf. Figure 1.1a). By deviating from the normalized shape of the index, the tracking portfolio can reduce the distance between its trajectory and the trajectory of the index that is inevitably caused by transaction costs. However, to achieve a high tracking accuracy, both a small distance between the trajectories and similar normalized shapes are required. Another disadvantage of existing return-based and value-based formulations is that they lead to substantial rebalancing costs when applied periodically over the fund's lifetime. Return-based formulations rebalance heavily because the associated transaction costs are not penalized in the objective function. Value-based formulations lead to substantial rebalancing because the rebalancing costs affect the distance between the trajectories only in the last historical period and can be overcompensated by a reduced distance in the previous periods. The above mentioned drawbacks can be addressed by combining value-based and return-based tracking-error measures in a single optimization criterion and limiting the rebalancing costs. However, since return-based formulations calculate the historical returns of the tracking portfolio either by using a non-convex function (cf., e.g., Beasley et al., 2003) or by assuming constant weights of the components in the tracking portfolio over time (cf., e.g., Canakgoz and Beasley, 2009), the resulting optimization criterion would represent a non-convex function or would rely on simplifying assumptions. Moreover, weighting the two measures and choosing an appropriate limit on the rebalancing



(a) Existing formulations: large distance or different shapes

(b) Novel formulation: small distance and similar shapes

Figure 1.1: Drawbacks of existing formulations and advantages of the proposed formulation.

costs would require an instance-specific fine tuning by the user.

In this paper, we propose a new value-based mixed-integer linear programming (MILP) formulation. The main feature of this formulation is that it determines a target trajectory internally instead of using the historical index trajectory as done by existing value-based formulations. The target trajectory has the same returns as the index, but its level may vary. As shown in Figure 1.1b, the objective function then minimizes both, the distance between the tracking-portfolio trajectory and the target trajectory, and the distance between the target trajectory and the index trajectory. Minimizing the former distance can be viewed as maximizing a similarity measure between the normalized shape of the tracking portfolio and the normalized shape of the index. In contrast to return-based formulations, which use a different similarity measure, the proposed similarity measure can be formulated as a linear function without assuming constant portfolio weights over time. Minimizing the latter distance leads to tracking portfolios with low rebalancing costs. Minimizing both distances simultaneously corresponds to optimizing the trade-off between rebalancing costs and similarity in terms of normalized shape. The proposed formulation represents a generalization of existing value-based formulations because if the distance between the target trajectory and the index trajectory is forced to be zero, e.g., when transaction costs are ignored, the proposed objective function is equivalent to the tracking-error measure of the existing value-based formulations. In addition to the novel optimization criterion, we introduce the following three modelling techniques: a) a valid inequality that leads to a tighter linear programming relaxation, b) a procedure that builds on an idea of Filippi et al. (2016) to strengthen the proposed MILP formulation by removing redundant decision variables and constraints, and c) a novel modelling technique to compute the actual transaction values and costs rather than just upper bounds as done in existing formulations (cf., e.g., Canakgoz and Beasley, 2009). The latter allows to consider a wide range of practical portfolio constraints such as the minimum transaction values imposed by many stock exchanges. These modelling techniques can be applied to related portfolio-optimization problems and are therefore of general interest.

In a computational experiment, we compared the proposed MILP formulation to a value-based and to a return-based MILP formulation from the literature. For the first time, we also compared the MILP formulations to the full-replication strategy, which demonstrates the benefits that can be obtained in practice by using optimization rather than a naïve approach. For the computational experiment, we used a set of problem instances from the literature (cf. Beasley et al., 2003; Canakgoz and Beasley, 2009; Guastaroba et al., 2009) and a set of newly constructed problem instances that are all based on real-world stock-market data. We applied the MILP formulations periodically in each period of a given investment horizon in two scenarios, one with stochastic cash changes and one without cash changes, and under three investment settings that differ in the considered set of portfolio constraints. To evaluate the tracking accuracy, we used as main criterion the mean-absolute deviation (MAD) between the value developments over the investment horizon of the index fund and the benchmark portfolio. This criterion reflects the trade-off between rebalancing costs and similarity in terms of normalized shape. As a secondary criterion, we used the root-mean squared error (RMSE) between the returns of the index fund and the index over the investment horizon. The results of our experiment indicate that the proposed formulation leads to the highest tracking accuracy in terms of MAD and RMSE. The comparison of the MILP formulations with the full-replication strategy demonstrates that, in practice, using optimization is preferable to using a naïve approach, particularly when the index includes many constituents and when frequent cash changes occur.

The remainder of this paper is organized as follows. In Section 1.2, we describe the index-tracking problem in detail and discuss the related literature. In Section 1.3, we present the proposed MILP formulation. In Section 1.4, we review the two MILP formulations that we used as benchmarks for the proposed MILP formulation in the computational experiment. In section 1.5, we describe the design of the computational experiment. In Section 1.6, we illustrate the design of the computational experiment and the advantages of the proposed MILP formulation by means of an example. In Section 1.7, we report the computational results. In Section 1.8, we provide some concluding remarks and directions for future research.

1.2 Optimization problem and related literature

In this section, we state the optimization problem (cf. Subsection 1.2.1) and review the related literature (cf. Subsection 1.2.2).

1.2.1 The problem of tracking an index

In practice, the index-tracking problem considered in this paper must be solved periodically over a given investment horizon to account for changed market environments and cash changes. In the following, we state the index-tracking problem that arises at the end of a particular period of the investment horizon.

Given are the values of a financial index and the prices of the index constituents for a set of historical periods, the composition of the current tracking portfolio, and information on current cash changes. Positive and negative cash changes, e.g., deposits and withdrawals from investors, increase and decrease the fund's capital, respectively. The total capital of the fund consists of this cash change, the value of the current tracking portfolio, the excess cash that has not been invested in the tracking portfolio in the previous period, and the interest earnings from the excess cash. Note that the construction of an initial tracking portfolio is a special case of the index-tracking problem in which no current tracking portfolio exists and the fund's capital arises from the initial cash change only.

Sought is the number of units of each stock that will represent the tracking portfolio in the next period. The objective is to minimize a tracking error that is measured as some function of the deviation between the historical performance of the tracking portfolio and the index.

It is assumed that transactions can be executed based on the closing market prices and cause proportional and fixed transaction costs. Analogously to Beasley et al. (2003), Canakgoz and Beasley (2009), and Guastaroba and Speranza (2012), it is further assumed that fractional values for the number of units of each stock are valid and that short sales are not allowed. Typically, additional constraints on the composition of the tracking portfolio are considered in index tracking. In this paper, we consider three investment settings that differ in the considered constraints (see Table 1.1). These investment settings can be characterized based on whether they are restrictive (R) or non-restrictive (NR) and based on whether tracking portfolios are self-financing (SF) or externally financed

Table 1.1: Investment settings for the index-tracking problem.

	Investment setting		
Constraints	*R-EF*	*R-SF*	*NR-SF*
Self-financing portfolios	✗	✓	✓
Upper bound on portfolio cardinality	✓	✓	✗
Bounds on portfolio value of each stock	LB and UB	LB and UB	✗
Bounds on transaction value of each stock	UB	UB	LB and UB

(*EF*). We consider an investment setting to be restrictive if the cardinality of the tracking portfolio is restricted by an upper bound on the number of stocks in the tracking portfolio either imposed directly or indirectly through the definition of ranges for the investment in individual stocks. A tracking portfolio is said to be self-financing if transaction costs must be paid by the fund's capital. In contrast, if transaction costs are paid out of a separate account, the tracking portfolio is externally financed. In the following, we describe the three investment settings:

- *R-EF*: This restrictive investment setting was proposed in Guastaroba and Speranza (2012). The transaction costs are paid out of an external account. Furthermore, it is assumed that the user imposes an upper bound on the tracking-portfolio cardinality. The investment in each included stock must lie within a user-defined range. Finally, the value of each transaction (purchase or sale of a stock) cannot exceed a user-defined upper bound.

- *R-SF*: This setting was also proposed in Guastaroba and Speranza (2012). The only difference from the first setting is that the tracking portfolios are self-financing.

- *NR-SF*: In this setting, the tracking portfolios are also self-financing. As in the first two settings, the value of each transaction cannot exceed a user-defined upper bound. The major difference compared to the restrictive investment settings is that no user-defined bounds on the portfolio cardinality and no user-defined ranges for the investment in each included stock are specified. Another difference is that a lower bound on the value of each transaction can also be specified. This lower bound allows consideration of minimum transaction values, which are often imposed by stock exchanges. The motivation for this third investment setting is that it provides more flexibility and may thus lead to a higher tracking accuracy.

1.2.2 Related literature

Based on how the tracking error is measured, the existing mathematical programming formulations for the index-tracking problem can be categorized into the two classes *return-based* (cf. Subsection 1.2.2.1) and *value-based* (cf. Subsection 1.2.2.2) formulations (cf. Gaivoronski et al., 2005). All formulations of the latter class employ a linear function to measure the tracking error. In contrast, several formulations of the former class use a tracking-error measure in which the historical returns of the tracking portfolio are computed as a non-convex function of the decision variables. Due to the non-convexity, general-purpose mathematical programming solvers such as CPLEX and Gurobi are not applicable for these formulations and heuristic solution methods must be applied instead (cf., e.g., Andriosopoulos and Nomikos, 2014; Beasley et al., 2003; Chiam et al., 2013; Maringer and Oyewumi, 2007). In this paper, we focus on the convex formulations.

1.2.2.1 Return-based formulations

The return-based formulations determine the composition of the tracking portfolio by minimizing a function of the historical tracking-portfolio returns and the historical index returns. Specifically, the function quantifies the total difference between the returns over all historical periods. The convex formulations from this class, on which we focus here, differ with respect to the objective function or the practical portfolio constraints.

Rudolf et al. (1999) propose a quadratic objective function that minimizes the sum of the squared differences between the returns of the tracking portfolio and the returns of the index without considering transaction costs. In addition to the quadratic objective function, four linear objective functions are presented that are based on absolute differences. Gaivoronski et al. (2005) consider a setting with transaction costs and self-financing tracking portfolios. For this setting, they present different types of linear and quadratic objective functions to measure the tracking error. Takeda et al. (2013) introduce an objective function that minimizes the average of the squared differences between the historical returns of the tracking portfolio and the historical returns of the index. They introduce a term into this objective function that penalizes large differences between the weights of the assets. Hence, equally-weighted portfolios are preferred. The authors argue that such portfolios might achieve a higher tracking accuracy in the future. Sant'Anna et al. (2017) also use the objective function of Takeda et al. (2013), but impose an upper bound on the absolute difference between the return of the tracking portfolio and the return of the index that must not be exceeded in any of the historical periods. The purpose of this upper bound is to avoid large return differences in single periods.

All approaches of this class rely on the simplifying assumption that the weights of the stocks in the tracking portfolio are constant over time. This assumption is required to compute the tracking-portfolio returns as a convex function of the decision variables. The assumption is also used in other convex formulations although they do not directly minimize a tracking-error measure that is based on the differences between the tracking-portfolio returns and the index returns. For example, Canakgoz and Beasley (2009) propose to construct a tracking portfolio such that a regression of its returns on the index returns yields an intercept (alpha) of zero and a slope (beta) of one. In their formulation, they compute alpha and beta under the assumption of constant portfolio weights. The assumption of constant weights is also used in other optimization problems in finance, e.g., in the problems of constructing an absolute return portfolio (cf. Valle et al., 2014), a factor neutral portfolio (cf. Valle et al., 2015), and a mean-VaR (value-at-risk) portfolio (cf. Lwin et al., 2017). Because the assumption of constant portfolio weights is rarely satisfied in reality due to individual price changes of the stocks, formulations from this class may lead to tracking portfolios with an unsatisfactory future tracking accuracy. Moreover, a low future tracking accuracy can occur even if the assumption of constant weights is satisfied or if the actual tracking-portfolio returns are considered by employing a non-convex function. As mentioned in the introduction, the reason for the low future tracking accuracy is that minimizing a return-based tracking error may result in substantial rebalancing costs and a large distance between the historical trajectories of the index and the tracking portfolio.

1.2.2.2 Value-based formulations

The value-based formulations do not rely on the assumption of constant portfolio weights. As tracking error, they measure the distance between the historical trajectories of the tracking portfolio and the index. The existing formulations from this class primarily differ with respect to the considered practical portfolio constraints.

Konno and Wijayanayake (2001) use the mean-absolute deviation (MAD) to quantify the distance between the historical trajectories of the tracking portfolio and the index. They consider non-convex transaction costs, a user-defined limit on the tracking error, and so-called minimal transaction unit constraints. These constraints require the transaction value of each stock to be an integer multiple of some minimal transaction value. Gaivoronski et al. (2005) propose, in addition to the return-based objective functions mentioned in the previous subsection, different value-based objective functions. They also present an approach to decide whether to rebalance the current portfolio based on new market information. In this approach, a new optimal candidate portfolio is computed first. Then, the tracking error of the current portfolio is compared to that of the candidate portfolio. If the difference between the tracking errors of the two portfolios is larger than a specific threshold, the candidate portfolio is selected as the new current portfolio. Hence, no partial rebalancing of the current portfolio towards the candidate portfolio is considered, which might be expensive in terms of transaction costs as the two portfolios could be structurally very different. Guastaroba and Speranza (2012) minimize the MAD introduced by Konno and Wijayanayake (2001) for the two investment settings *R-EF* and *R-SF* described above.

If the tracking portfolio is self-financing and its composition is rebalanced, then the distance between the historical trajectory of the tracking portfolio and the historical trajectory of the index cannot be zero. This is because the tracking-portfolio trajectory must lie below the index trajectory in the last historical period due to the scaling factor that is used to determine the index trajectory. By deviating from the normalized historical trajectory of the index, the tracking portfolio can reduce the distance between its trajectory and the trajectory of the index in the previous periods. A drawback of the value-based formulations is that the different normalized historical trajectories can lead to an unsatisfactory tracking accuracy in the future. Moreover, as mentioned in the introduction, the value-based formulations lead to substantial rebalancing costs that can also cause a low future tracking accuracy.

1.3 New mixed-integer linear programming formulation

In this section, we present the new MILP formulation for the index-tracking problem that arises at the end of a particular period T of the investment horizon as described in the previous section. In Subsection 1.3.1, we present the constraints for the proposed non-restrictive and self-financing investment setting *NR-SF*. In Subsection 1.3.2, we present the novel optimization criterion. In Subsection 1.3.3, we show how the proposed formulation can be strengthened by adding a valid inequality and removing some redundant variables and constraints. The nomenclature used in this paper is provided in Tables 1.2 and 1.3.

1.3.1 Constraints

We consider the problem of tracking a financial index with n stocks in the proposed non-restrictive and self-financing investment setting *NR-SF* (see Table 1.1 for a description of the investment setting). Let X_{jT} be the main decision variables that correspond to the number of units of each stock $j \in J$ in the tracking portfolio after rebalancing, and let

Table 1.2: Sets, parameters, and decision variables used in the MILP formulations.

	Sets and parameters:	
	T	End of the current period (today) at which the index-tracking problem has to be solved
	n	Number of index constituents
	J	Set of index constituents ($= \{1, \ldots, n\}$)
	q_{jt}/I_t	Historical closing price of stock $j \in J$/index at $t \in \{1, \ldots, T\}$
	κ_T	Cash change at T
	Y_{jT}	Number of units of stock $j \in J$ in the tracking portfolio before rebalancing at T
	c_{T-1}	Excess cash at $T-1$
	r_T	Interest rate on the excess cash for period p_T
	C_T	Capital of the fund at T ($C_T = \kappa_T + \sum_{j \in J} Y_{jT} q_{jT} + c_{T-1}(1 + r_T)$)
	γ_T	Percentage of C_T denoting the available funds in the external account for transaction costs
	$\varepsilon_{jT}/\delta_{jT}$	Percentage of C_T denoting the minimum/maximum value of stock $j \in J$ if included in the tracking portfolio at T
	ζ_{jT}/η_{jT}	Percentage of C_T denoting the minimum/maximum transaction value for stock $j \in J$ if traded at T
	k	Maximum cardinality of the tracking portfolio
	c_j^f	Fixed transaction costs for trading stock $j \in J$
	c_j^b/c_j^s	Proportional transaction costs for purchasing/selling stock $j \in J$ as a percentage of the transaction value
	Continuous decision variables:	
*	G_{jT}	Total transaction costs for stock $j \in J$ at T
*	v_{jT}^b/v_{jT}^s	Value purchased/sold of stock $j \in J$ at T
*	X_{jT}	Number of units of stock $j \in J$ in the tracking portfolio after rebalancing at T
*	u_{tT}/d_{tT}	Non-negative variables that represent the absolute upside/downside deviation at $t \in \{1, \ldots, T\}$
	Binary decision variables:	
*	z_{jT}	$= 1$, if $X_{jT} > 0$; $= 0$, otherwise ($j \in J$)
*	w_{jT}	$= 1$, if $Y_{jT} \neq X_{jT}$; $= 0$, otherwise ($j \in J$)
*	w_{jT}^b	$= 1$, if $Y_{jT} < X_{jT}$; $= 0$, otherwise ($j \in J$)
*	w_{jT}^s	$= 1$, if $Y_{jT} > X_{jT}$; $= 0$, otherwise ($j \in J$)

Table 1.3: Additional notation used in this paper that is not listed in Table 1.2.

$\mathbf{x}$	Vector of decision variables $[X_{1T} \; \dots \; X_{nT}]$
$f(\mathbf{x})$	Proposed optimization criterion
$g(\mathbf{x})$	Optimization criterion used in value-based formulations from the literature
$Q_t(\mathbf{x})$	Value of the tracking portfolio at time $t \in \{1, \dots, T\}$
p_t	Period t that starts at time $t-1$ and ends at time t
S	Start of the investment horizon
E	End of the investment horizon
τ	Time progress in periods since S
P_t	Value of the index fund at time t, $t \in \{S+1, \dots, E\}$
P_t^{FR}	Value of an index fund that invests according to the full-replication strategy at time t, $t \in \{S+1, \dots, E\}$
B_t	Value of the benchmark portfolio at time t, $t \in \{S+1, \dots, E\}$

q_{jT} be the price of each stock $j \in J$ at the end of period T. The value of the tracking portfolio after rebalancing is computed as $\sum_{j \in J} q_{jT} X_{jT}$. Since the tracking portfolio is self-financing, its value after rebalancing plus the transaction costs spent for rebalancing ($\sum_{j \in J} G_{jT}$) must not exceed the capital of the index fund (C_T):

$$\sum_{j \in J} (q_{jT} X_{jT} + G_{jT}) \leq C_T \tag{1.1}$$

To compute the transaction costs G_{jT} for each stock $j \in J$, we introduce the non-negative continuous variables v_{jT}^b and v_{jT}^s that represent the value of each stock purchased and sold, respectively. The following set of constraints is used to determine the values of v_{jT}^b and v_{jT}^s. Parameter Y_{jT} denotes the number of units of each stock in the tracking portfolio before rebalancing:

$$v_{jT}^b - v_{jT}^s = q_{jT}(X_{jT} - Y_{jT}) \qquad (j \in J) \tag{1.2}$$

The binary variables w_{jT}^b and w_{jT}^s are equal to one if stock $j \in J$ is purchased and sold, respectively, and zero otherwise. The following constraints (1.3) and (1.4) assign the appropriate values to these binary variables and simultaneously impose lower and upper bounds to ensure that the value of stock $j \in J$ purchased or sold is not below the minimum transaction value ($\zeta_{jT} C_T$) and not above the maximum transaction value ($\eta_{jT} C_T$).

$$\zeta_{jT} C_T w_{jT}^b \leq v_{jT}^b \leq \eta_{jT} C_T w_{jT}^b \qquad (j \in J) \tag{1.3}$$

$$\zeta_{jT} C_T w_{jT}^s \leq v_{jT}^s \leq \eta_{jT} C_T w_{jT}^s \qquad (j \in J) \tag{1.4}$$

Constraints (1.5) prevent the binary variables w_{jT}^b and w_{jT}^s to be equal to one at the same time. These constraints together with constraints (1.3) and (1.4) ensure that for each stock $j \in J$ the value purchased and the value sold cannot be strictly positive at the same time, and that constraints (1.2) assign the actual transaction values of the stocks to the variables v_{jT}^b and v_{jT}^s.

$$w_{jT}^b + w_{jT}^s \leq 1 \qquad (j \in J) \tag{1.5}$$

Without the constraints (1.5), both variables w_{jT}^b and w_{jT}^s could be equal to one for some stock $j \in J$, which means that both variables v_{jT}^b and v_{jT}^s could be strictly positive at the same time. Hence, without the constraints (1.5), the sum of v_{jT}^b and v_{jT}^s represents an upper bound on the transaction value of stock j rather than the exact transaction value. This upper bound is sufficient to model to maximum transaction values that are considered in the literature, but is not sufficient to model the minimum transaction values that are considered in the proposed investment setting.

The transaction costs G_{jT} related to each stock $j \in J$ include proportional costs for purchasing the stock ($c_j^b v_{jT}^b$), proportional costs for selling the stock j ($c_j^s v_{jT}^s$), and fixed costs for trading the stock ($c_j^f(w_{jT}^b + w_{jT}^s)$). The transaction costs G_{jT} are computed as follows:

$$G_{jT} = c_j^b v_{jT}^b + c_j^s v_{jT}^s + c_j^f(w_{jT}^b + w_{jT}^s) \qquad (j \in J) \tag{1.6}$$

The domains of the decision variables are specified as follows:

$$w_{jT}^b \in \{0,1\},\ w_{jT}^s \in \{0,1\} \qquad (j \in J) \tag{1.7}$$

$$v_{jT}^b \geq 0,\ v_{jT}^s \geq 0,\ X_{jT} \geq 0,\ G_{jT} \geq 0 \qquad (j \in J) \tag{1.8}$$

1.3.2 Objective function

In this subsection, we present the novel optimization criterion. To simplify the notation, let $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ be the vector of decision variables, i.e., $\mathbf{x} = \begin{bmatrix} X_{1T} & \dots & X_{nT} \end{bmatrix}$, and let $Q_t(\mathbf{x})$ denote the value of the tracking portfolio at the historical time points $t \in \{1, \dots, T\}$ computed as $Q_t(\mathbf{x}) = \sum_{j \in J} q_{jt} X_{jT}$. We assume that all historical index values and stock prices are strictly positive ($I_t > 0,\ q_{jt} > 0,\ j \in J,\ t \in \{1, \dots, T\}$), which means that the value of the tracking portfolio will also be strictly positive in each historical period if we invest in the stocks, i.e., $Q_t(\mathbf{x}) > 0$, $t \in \{1, \dots, T\}$, if $\|\mathbf{x}\|_1 = \sum_{j \in J} |X_{jT}| > 0$.

The objective is to minimize the value of the following function $f(\mathbf{x})$:

$$f(\mathbf{x}) = \sum_{t=1}^{T} \left| Q_t(\mathbf{x}) - I_t \frac{Q_T(\mathbf{x})}{I_T} \right| + \sum_{t=1}^{T} \left(I_t \frac{C_T}{I_T} - I_t \frac{Q_T(\mathbf{x})}{I_T} \right) \tag{1.9}$$

We obtain the following linear formulation to minimize $f(\mathbf{x})$ by introducing the non-negative auxiliary decision variables u_{tT} and d_{tT}, which capture the absolute positive and negative values of $\left| Q_t(\mathbf{x}) - I_t \frac{Q_T(\mathbf{x})}{I_T} \right|$ at time $t \in \{1, \ldots, T\}$, respectively:

$$NEW \begin{cases} \text{Min.} \displaystyle\sum_{t=1}^{T} \left(d_{tT} + u_{tT} + I_t \frac{C_T - Q_T(\mathbf{x})}{I_T} \right) & (1.10) \\ \text{s.t. } u_{tT} - d_{tT} = Q_t(\mathbf{x}) - I_t \dfrac{Q_T(\mathbf{x})}{I_T} \quad (t \in \{1, \ldots, T\}) & (1.11) \\ d_{tT}, u_{tT} \geq 0 \quad (t \in \{1, \ldots, T\}) & (1.12) \end{cases}$$

Minimizing (1.10) subject to constraints (1.11) and (1.12) is equivalent to minimizing $f(\mathbf{x})$ because in an optimal solution, u_{tT} and d_{tT} will never be both strictly positive. Hence, in an optimal solution, we have for all $t \in \{1, \ldots, T\}$ that $u_{tT} + d_{tT} = \left| Q_t(\mathbf{x}) - I_t \frac{Q_T(\mathbf{x})}{I_T} \right|$ (see the proof by Konno and Wijayanayake, 2001, for a similar objective function).

The objective function $f(\mathbf{x})$ is composed of two terms.

The first term of $f(\mathbf{x})$ captures the distance between the historical tracking-portfolio trajectory ($Q_t(\mathbf{x}),\ t \in \{1, \ldots, T\}$) and the historical target trajectory, which the optimization model determines internally by scaling the index values to the value of the tracking portfolio at time T ($I_t \frac{Q_T(\mathbf{x})}{I_T},\ t \in \{1, \ldots, T\}$). In contrast, existing value-based formulations from the literature use the historical index trajectory that is obtained by scaling the index values to the capital of the fund at time T ($I_t \frac{C_T}{I_T},\ t \in \{1, \ldots, T\}$). Using the historical index trajectory has the drawback that a considerable difference between the normalized historical trajectory of the tracking portfolio ($\frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})},\ t \in \{1, \ldots, T\}$) and the normalized historical trajectory of the index ($\frac{I_t}{I_T},\ t \in \{1, \ldots, T\}$) can arise (cf. Figure 1.1a in Section 1.1). The reason for this difference is that the distance between the historical tracking-portfolio trajectory at time T ($Q_T(\mathbf{x})$) and the historical index trajectory at time T (C_T) cannot be zero when the tracking portfolio is rebalanced due to the transaction costs and the budget constraint (1.1). By using the historical target trajectory, we can overcome the mentioned drawback. To see this, note that the first term of $f(\mathbf{x})$ is equal to the distance between the normalized historical trajectory of the tracking portfolio and the normalized historical trajectory of the index multiplied by the value of the tracking portfolio at time T:

$$\sum_{t=1}^{T}\left|Q_t(\mathbf{x})-I_t\frac{Q_T(\mathbf{x})}{I_T}\right|=\sum_{t=1}^{T}\left|\frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})}-\frac{I_t}{I_T}\right|Q_T(\mathbf{x}) \tag{1.13}$$

If we invest in some of the stocks, i.e., if $\|\mathbf{x}\|_1 > 0$, then it follows from (1.13) that the normalized historical trajectories of the tracking portfolio and the index are identical if and only if the historical tracking-portfolio trajectory and the historical target trajectory are identical, i.e,

$$\sum_{t=1}^{T}\left|Q_t(\mathbf{x})-I_t\frac{Q_T(\mathbf{x})}{I_T}\right|=0 \iff \sum_{t=1}^{T}\left|\frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})}-\frac{I_t}{I_T}\right|=0 \tag{1.14}$$

Hence, minimizing the first term of $f(\mathbf{x})$ can be viewed as maximizing a measure of the similarity between the normalized historical trajectories of the tracking portfolio and the index. Also, minimizing existing non-convex return-based tracking-error functions can be viewed as maximizing a measure of the similarity between the normalized historical trajectories of the index and the tracking portfolio, because the normalized historical trajectories are identical if and only if the historical returns of the tracking portfolio and the index are identical. This relation is shown in Proposition 1 (cf. Appendix 1.A for the proof). However, it is important to note that the similarity between the normalized historical trajectories in the existing non-convex return-based formulations is measured differently than in the proposed formulation. Hence, minimizing the first term of $f(\mathbf{x})$ is not equivalent to minimizing the return differences.

Proposition 1. *If $\|\mathbf{x}\|_1 > 0$, then*

$$\frac{Q_t(\mathbf{x})}{Q_{t-1}(\mathbf{x})}-1=\frac{I_t}{I_{t-1}}-1\ (t\in\{2,\ldots,T\}) \iff \sum_{t=1}^{T}\left|\frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})}-\frac{I_t}{I_T}\right|=0$$

By combining (1.14) and Proposition 1, we obtain the following implications:

$$\sum_{t=1}^{T}\left|Q_t(\mathbf{x})-I_t\frac{Q_T(\mathbf{x})}{I_T}\right|=0 \iff$$
$$\frac{Q_t(\mathbf{x})}{Q_{t-1}(\mathbf{x})}=\frac{I_t}{I_{t-1}},\ (t\in\{2,\ldots,T\}) \iff \sum_{t=1}^{T}\left|\frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})}-\frac{I_t}{I_T}\right|=0 \tag{1.15}$$

Regarding the first term of $f(\mathbf{x})$, we can conclude that minimizing the differences between the historical returns of the index and the tracking portfolio and minimizing the first term of $f(\mathbf{x})$ can both be viewed as maximizing a measure of the similarity between the normalized historical trajectories of the tracking portfolio and the index. The advantage of the proposed similarity measure, i.e., the first term of $f(\mathbf{x})$, is that it can be formulated as a linear function without relying on the assumption of constant portfolio weights over time.

The second term of $f(\mathbf{x})$ penalizes the high rebalancing costs that can occur when one focuses on the normalized historical trajectories only. The second term is a function of the difference between the capital of the fund (C_T) and the value of the tracking portfolio at time T ($Q_T(\mathbf{x})$). This difference cannot be negative because of the budget constraint (1.1). Specifically, the second term of $f(\mathbf{x})$ captures the distance between the historical index trajectory and the historical target trajectory. The second term counterbalances the preference by the first term of $f(\mathbf{x})$ for tracking portfolios with a small value $Q_T(\mathbf{x})$. This preference exists because for two tracking portfolios with a positive first term of $f(\mathbf{x})$ and identical normalized historical trajectories, the tracking portfolio with the smaller value $Q_T(\mathbf{x})$ leads to a smaller value of the first term of $f(\mathbf{x})$ (cf. (1.13)). In an extreme case, the solution $\mathbf{x} = \mathbf{0}$, with $\mathbf{0}$ denoting the zero vector with appropriate dimension, leads to a value of zero for the first term of $f(\mathbf{x})$. The second term is chosen such that the solution $\mathbf{x} = \mathbf{0}$ cannot be optimal if we can find a feasible tracking portfolio ($\mathbf{x}^*$) whose historical trajectory is closer to the historical target trajectory than the historical trajectory of a tracking portfolio that is not invested in the stocks at all. We assume here that it is always possible to find such a feasible tracking portfolio $\mathbf{x}^*$.

Proposition 2. $\exists\ \mathbf{x}^* : \sum_{t=1}^{T} \left| Q_t(\mathbf{x}^*) - I_t \frac{Q_T(\mathbf{x}^*)}{I_T} \right| < \frac{Q_T(\mathbf{x}^*)}{I_T} \sum_{t=1}^{T} I_t$
$\Rightarrow$ $\mathbf{x} = \mathbf{0}$ *is not optimal to the objective function (1.9).*

The proof of Proposition 2 can be found in Appendix 1.A.

To conclude, minimizing the proposed optimization criterion $f(\mathbf{x})$ corresponds to optimizing the trade-off between rebalancing costs and the similarity in terms of normalized historical trajectories.

A special feature of the proposed optimization criterion $f(\mathbf{x})$ is that, when transaction costs can be ignored, it reduces to the existing value-based tracking-error measure that was used by Guastaroba and Speranza (2012), Konno and Wijayanayake (2001), and Filippi et al. (2016), i.e., $g(\mathbf{x}) = \sum_{t=1}^{T} \left| Q_t(\mathbf{x}) - I_t \frac{C_T}{I_T} \right|$. To see this, note that without transaction costs, the entire capital of the fund can be invested in the tracking portfolio and the budget constraint (1.1) can be written as $Q_T(\mathbf{x}) = C_T$. By replacing $Q_T(\mathbf{x})$ with C_T in $f(\mathbf{x})$, we obtain that $f(\mathbf{x})$ is equal to $g(\mathbf{x})$, and hence, it follows that minimizing

$f(\mathbf{x})$ is equivalent to minimizing $g(\mathbf{x})$, i.e.,

$$Q_T(\mathbf{x}) = C_T \Rightarrow \text{Min. } f(\mathbf{x}) \equiv \text{Min. } g(\mathbf{x}) \tag{1.16}$$

1.3.3 A strengthened formulation

In this subsection, we strengthen the proposed MILP formulation by removing redundant variables and constraints, and by adding a valid inequality that leads to a tighter linear programming relaxation.

The procedure for removing redundant variables and constraints builds on an idea presented by Filippi et al. (2016), who made the following observation for the *R-EF* investment setting: if some stock is not included in the current tracking portfolio, then this stock is always included in the tracking portfolio after rebalancing if it is traded. Based on this observation, the size of the MILP formulation presented in Filippi et al. (2016) was reduced substantially.

Similarly, for the *NR-SF* setting, we can make the following observation: a stock that is not included in the current portfolio cannot be sold because short selling is not allowed ($Y_{jT} = 0 \Rightarrow v^s_{jT} = w^s_{jT} = 0,\ j \in J$). Thus, we can remove the binary and the continuous variables w^s_{jT} and v^s_{jT}, respectively, and replace the continuous variables v^b_{jT} by $q_{jT}X_{jT}$ for all stocks that are not included in the current tracking portfolio ($j \in J : Y_{jT} = 0$). Below, we provide for the respective constraints of the investment setting *NR-SF* their strengthened version.

$$(1.2): v^b_{jT} - v^s_{jT} = q_{jT}(X_{jT} - Y_{jT}) \qquad (j \in J : Y_{jT} > 0) \tag{1.17}$$

$$(1.1): (1.1)\ \text{(unchanged)}$$

$$(1.6): G_{jT} = c^b_j v^b_{jT} + c^s_j v^b_{jT} + c^f_j (w^b_{jT} + w^s_{jT}) \qquad (j \in J : Y_{jT} > 0) \tag{1.18}$$

$$(1.6): G_{jT} = q_{jT} c^b_j X_{jT} + c^f_j w^b_{jT} \qquad (j \in J : Y_{jT} = 0) \tag{1.19}$$

$$(1.3): \zeta_{jT} C_T w^b_{jT} \le v^b_{jT} \le \eta_{jT} C_T w^b_{jT} \qquad (j \in J : Y_{jT} > 0) \tag{1.20}$$

$$(1.3): \zeta_{jT} C_T w^b_{jT} \le q_{jT} X_{jT} \le \eta_{jT} C_T w^b_{jT} \qquad (j \in J : Y_{jT} = 0) \tag{1.21}$$

$$(1.4): \zeta_{jT} C_T w^s_{jT} \le v^s_{jT} \le \eta_{jT} C_T w^s_{jT} \qquad (j \in J : Y_{jT} > 0) \tag{1.22}$$

$$(1.5): w^b_{jT} + w^s_{jT} \le 1 \qquad (j \in J : Y_{jT} > 0) \tag{1.23}$$

$$(1.7)/(1.8): v^b_{jT} \ge 0, w^s_{jT} \in \{0,1\}, v^s_{jT} \ge 0 \qquad (j \in J : Y_{jT} > 0) \tag{1.24}$$

$$(1.7)/(1.8): w^b_{jT} \in \{0,1\}, X_{jT} \ge 0, G_{jT} \ge 0 \qquad (j \in J) \tag{1.25}$$

To further strengthen the formulation, we provide a valid inequality, which is derived from two upper bounds on the value of the tracking portfolio at time T ($\sum_{j \in J} q_{jT} X_{jT}$). The first upper bound exists because the purchasing amounts of the stocks are limited through parameter η_{jT}. The second upper bound exists because the total portfolio value must not exceed the index fund's capital minus the total transaction costs. Both upper bounds depend on the number of purchased stocks $b = \sum_{j \in J} w^b_{jT}$, yet in opposite directions. The first upper bound increases with increasing b, while the second upper bound decreases with increasing b because of the fixed transaction costs. Due to the opposite impact of b on the two upper bounds, it is possible to compute a value for b that achieves the maximum total tracking portfolio value. Since we know that b must be integer, we can devise the following valid inequality (cf. Appendix 1.A for the proof):

Proposition 3. *Let* $c^b = \min\{c^b_j\} > 0$, $c^f = \min\{c^f_j\} > 0$, $\eta = \max\{\eta_{jT}\} > 0$, *and* $b = \frac{\kappa_T + c_{T-1}(1+r_T)}{(1+c^b)\eta C_T + c^f}$, *then we obtain the following upper bound on the value of the tracking portfolio in the investment setting NR-SF.*

$$\begin{aligned}
&\sum_{j \in J} q_{jT} X_{jT} \leq \\
&\max\left\{\min\left\{\sum_{j \in J} q_{jT} Y_{jT} + \eta C_T \lceil b \rceil, \sum_{j \in J} q_{jT} Y_{jT} + \frac{\kappa_T + c_{T-1}(1+r_T) - c^f \lceil b \rceil}{1+c^b}\right\},\right. \\
&\left.\min\left\{\sum_{j \in J} q_{jT} Y_{jT} + \eta C_T \lfloor b \rfloor, \sum_{j \in J} q_{jT} Y_{jT} + \frac{\kappa_T + c_{T-1}(1+r_T) - c^f \lfloor b \rfloor}{1+c^b}\right\}\right\}
\end{aligned} \tag{1.26}$$

As an example, let us assume that we construct an initial tracking portfolio. Hence, the value of the current tracking portfolio and the excess cash from the previous period are both zero, i.e., $\sum_{j \in J} q_{jT} Y_{jT} = 0$ and $c_{T-1} = 0$. In this case, the capital of the fund consists of the cash change only, i.e., $C_T = \kappa_T$. Thus, we have $b = \frac{\kappa_T}{(1+c^b)\eta\kappa_T + c^f}$, and the valid inequality (1.26) reduces to:

$$\sum_{j \in J} q_{jT} X_{jT} \leq \max\left\{\min\left\{\eta\kappa_T \lceil b \rceil, \frac{\kappa_T - c^f \lceil b \rceil}{1+c^b}\right\}, \min\left\{\eta\kappa_T \lfloor b \rfloor, \frac{\kappa_T - c^f \lfloor b \rfloor}{1+c^b}\right\}\right\}$$

Let us further assume that $\eta = 0.1$, $\kappa_T = 100'000$, $c^f = 12$, and $c^b = 0.01$. Hence, we have $b \approx 9.89$, and

$$\sum_{j \in J} q_{jT} X_{jT} \leq \max\{\min\{100'000, 98'891\}, \min\{90'000, 98'903\}\} = 98'891$$

In this example, no more than 98.891% of the cash inflow can be invested in the tracking portfolio.

1.4 Benchmark mixed-integer linear programming formulations

In this section, we briefly describe the benchmark MILP formulations for our computational experiment. In Subsection 1.4.1, we state the constraints that correspond to the investment settings *R-EF* and *R-SF*. The variables that are used to model these investment settings differ from the variables used to model the proposed investment setting *NR-SF* as follows: the binary variables w^b_{jT} and w^s_{jT} are not required anymore, and the binary variables w_{jT} and z_{jT} are introduced. The variables w_{jT} are equal to one if some stock $j \in J$ is traded, and zero otherwise. The variables z_{jT} are equal to one if some stock $j \in J$ is included in the tracking portfolio, and zero otherwise. In Subsection 1.4.2, we present two state-of-the-art tracking-error measures that have been used in the literature. The two tracking-error measures, which we refer to as *RET* and *VAL*, represent the return-based and the value-based class, respectively.

1.4.1 Constraints

The following constraints represent the investment setting *R-EF* presented by Guastaroba and Speranza (2012):

$$
\textit{R-EF}\begin{cases}
\sum_{j\in J} q_{jT}X_{jT} \leq C_T & & (1.27)\\
G_{jT} = c^b_j v^b_{jT} + c^s_j v^s_{jT} + c^f_j w_{jT} & (j \in J) & (1.28)\\
\sum_{j\in J} G_{jT} \leq \gamma_T C_T & & (1.29)\\
v^b_{jT} - v^s_{jT} = q_{jT}(X_{jT} - Y_{jT}) & (j \in J) & (1.2)\\
v^b_{jT} + v^s_{jT} \leq \eta_{jT} C_T w_{jT} & (j \in J) & (1.30)\\
\varepsilon_{jT} C_T z_{jT} \leq q_{jT}X_{jT} \leq \delta_{jT} C_T z_{jT} & (j \in J) & (1.31)\\
\sum_{j\in J} z_{jT} \leq k & & (1.32)\\
w_{jT} \in \{0,1\},\ z_{jT} \in \{0,1\} & (j \in J) & (1.33)\\
v^b_{jT} \geq 0,\ v^s_{jT} \geq 0,\ X_{jT} \geq 0,\ G_{jT} \geq 0 & (j \in J) & (1.34)
\end{cases}
$$

The budget constraint (1.27) ensures that the value of the tracking portfolio does not exceed the fund's capital (investment budget). Constraints (1.28) compute the total transaction costs for each stock $j \in J$. These costs include proportional costs for purchasing stock j ($c_j^b v_{jT}^b$), proportional costs for selling stock j ($c_j^s v_{jT}^s$), and fixed transaction costs for trading stock j ($c_j^f w_{jT}$). Constraint (1.29) guarantees that the total transaction costs do not exceed the balance of the external account for transaction costs. Constraints (1.2) from the investment setting *NR-SF* compute for each stock $j \in J$ the value purchased (v_{jT}^b) and the value sold (v_{jT}^s) with respect to the current portfolio. Note that the variables v_{jT}^b and v_{jT}^s can both be strictly positive, and thus, the variables v_{jT}^b and v_{jT}^s can over-estimate the actual transaction values. Hence, the transaction costs determined by constraints (1.28) represent an upper bound on the actual transaction costs. This upper bound is sufficient for ensuring that the actual transaction costs do not exceed the balance of the external account. However, when we want to consider minimum transaction values as in the investment setting *NR-SF*, a computation of the exact transaction values is required. Constraints (1.30) impose upper bounds on the transaction value of stock $j \in J$ and prevent the binary variables w_{jT} from having a value of zero if stock j is traded. Constraints (1.31) assign a value of one to the binary variables z_{jT} if the value of stock $j \in J$ in the tracking portfolio is greater than zero, and a value of zero otherwise. Constraints (1.31) also ensure that the value of each selected stock in the tracking portfolio is in the range $[\varepsilon_{jT} C_T, \delta_{jT} C_T]$. Constraint (1.32) imposes an upper bound on the portfolio cardinality. The domains of the variables are defined by (1.33) and (1.34).

Guastaroba and Speranza (2012) propose a second investment setting in which the tracking portfolios are self-financing. The following constraints represent this investment setting:

$$\textit{R-SF} \begin{cases} (1.1) \\ (1.2) \\ (1.28) \\ (1.30) - (1.34) \end{cases}$$

Note that the budget constraint (1.27) is replaced by (1.1) and that constraint (1.29) is dropped because the external account to finance the transaction costs is no longer needed.

1.4.2 Objective functions

Sant'Anna et al. (2017) and Takeda et al. (2013) minimize the tracking error computed as the average of the squared differences between the historical returns of the tracking portfolio and the historical returns of the index. Because we focus on mixed-integer linear programming formulations in this paper to ensure a fair comparison between the different formulations in the computational experiment, we decided to use the average absolute differences between the historical returns of the tracking portfolio and the historical returns of the index as the tracking-error measure to represent the return-based formulations. Under the assumption of constant weights, the weight of each stock $j \in J$ is computed by $\frac{q_{jT}X_{jT}}{C_T}$ and the objective is to minimize the following tracking-error measure:

$$\frac{1}{T-1}\sum_{t=2}^{T}\left|\sum_{j\in J}\left(\frac{q_{jT}X_{jT}}{C_T}\frac{q_{jt}}{q_{j,t-1}}\right)-\frac{I_t}{I_{t-1}}\right| \tag{1.35}$$

A linear formulation to minimize this tracking-error measure can be obtained as follows:

$$RET \begin{cases} \text{Min. } \dfrac{1}{T-1}\displaystyle\sum_{t=2}^{T}(d_{tT}+u_{tT}) & & (1.36) \\ \text{s.t. } d_{tT}-u_{tT}=\dfrac{I_t}{I_{t-1}}-\displaystyle\sum_{j\in J}\left(\dfrac{q_{jt}q_{jT}X_{jT}}{q_{j,t-1}C_T}\right) & (t\in\{2,\dots,T\}) & (1.37) \\ d_{tT},u_{tT}\geq 0 & (t\in\{2,\dots,T\}) & (1.38) \end{cases}$$

The value-based formulations are represented by the tracking-error measure used by Konno and Wijayanayake (2001) and Guastaroba and Speranza (2012). This tracking-error measure is calculated as the sum of the absolute differences between the historical value developments of the tracking portfolio and the index as follows:

$$\sum_{t=1}^{T}\left|\sum_{j\in J}q_{jt}X_{jT}-I_t\frac{C_T}{I_T}\right| \tag{1.39}$$

A linear formulation to minimize (1.39) is obtained as follows:

$$VAL \begin{cases} \text{Min. } \displaystyle\sum_{t=1}^{T}(d_{tT}+u_{tT}) & & (1.40) \\ \text{s.t. } u_{tT}-d_{tT}=\displaystyle\sum_{j\in J}q_{jt}X_{jT}-I_t\dfrac{C_T}{I_T} & (t\in\{1,\dots,T\}) & (1.41) \\ d_{tT},u_{tT}\geq 0 & (t\in\{1,\dots,T\}) & (1.42) \end{cases}$$

1.5 Experimental design

In practice, the index-fund managers have to solve the index-tracking problem periodically over a given investment horizon to account for changed market environments and cash changes. Our computational experiment to test the formulations from Sections 1.3 and 1.4 is designed to reflect this situation of index-fund managers.

In the proposed experimental design, we use a discrete time horizon that starts at time 0 and ends at time E. The time horizon is divided into the set of periods $\{\, p_t \mid t \in \{1, \ldots, E\} \,\}$. Each period p_t starts at time point $t-1$ and ends at t. The goal is to track the index over an investment horizon that starts at $S < E$, ends at E, and consists of the set of periods $\{\, p_t \mid t \in \{S+1, \ldots, E\} \,\}$. Figure 1.2 illustrates the time and the investment horizon.

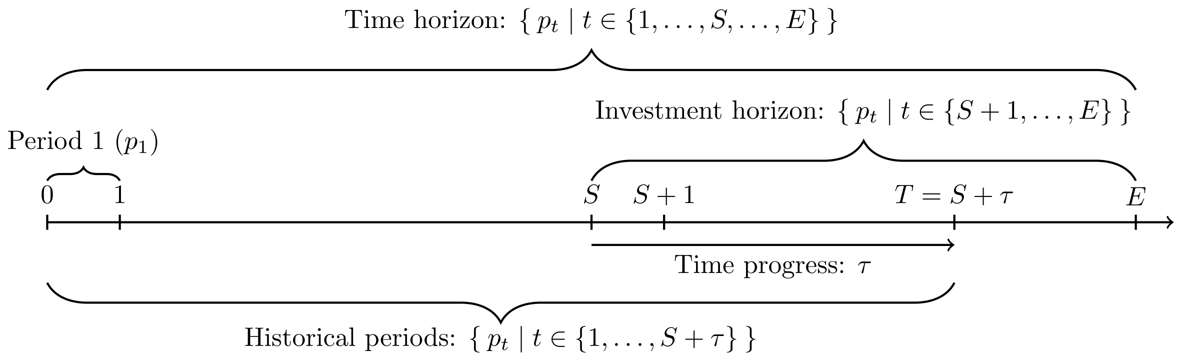


Figure 1.2: Decision situation after τ periods have passed since the start of the investment horizon S.

The index-fund managers solve the index-tracking problem sequentially at the beginning of each period of the investment horizon. Starting at time S, the composition of the initial tracking portfolio is determined based on the market information collected from the set of historical periods $\{\, p_t \mid t \in \{1, \ldots, S\} \,\}$. As time progresses, at each succeeding time point $S + \tau$, $\tau \in \{1, \ldots, E - S - 1\}$, the composition of the tracking portfolio can be rebalanced in response to cash changes and the market information from the updated set of historical periods $\{\, p_t \mid t \in \{1, \ldots, S + \tau\} \,\}$, which also includes the new information that has been revealed in period $p_{S+\tau}$.

Following Guastaroba and Speranza (2012), we consider a time horizon that spans 156 periods; each period corresponds to one week. The last 52 periods of the time horizon represent the investment horizon. Hence, the tracking portfolio's composition has to be determined successively at each time point of the investment horizon $T \in \{104, \ldots, 155\}$. We assume that an initial tracking portfolio has to be constructed at the beginning of the investment horizon at time point $S = 104$. The index fund's capital at S consists of the initial cash change κ_S only, which we assume to be 100. For the one-period interest rate for cash r_T, we use a value of zero in all periods. As mentioned in Section 1.2, we consider the three investment settings *R-EF*, *R-SF*, and *NR-SF*, which differ with respect to the practical portfolio constraints. For the practical portfolio constraints, we chose the specific parameter values according to the most recent study that used the investment

Table 1.4: Parameter values for the computational experiment.

$S = 104,\ E = 156,\ c_{S-1} = 0,\ \kappa_S = 100$	
$Y_{jS} = 0,\ c_j^f = 0.012,\ c_j^b = 0.01,\ c_j^s = 0.01$	$(j \in J)$
$r_T = 0,\ \gamma_T = 0.015$	$(T \in \{S, \ldots, E-1\})$
$\varepsilon_{jT} = \zeta_{jT} = 0.01,\ \delta_{jT} = \eta_{jT} = 0.1$	$(j \in J,\ T \in \{S, \ldots, E-1\})$

setting *R-EF* (cf. Filippi et al., 2016). For the investment setting *NR-SF*, we had to additionally define the minimum transaction value for each stock (ζ_{jT}). For simplicity, we chose the same value for this parameter as for the parameter that determines the minimum value of each stock in the tracking portfolio (ε_{jT}). Table 1.4 lists all parameter values of the computational experiment.

Figure 1.3 illustrates the experiment using a flowchart. The MILP formulations are first applied for period $T = 104$ to determine the initial composition of the tracking portfolio. Then, T is incremented ($T := T + 1$), and the values of the parameters are updated as follows:

$$Y_{jT} := X_{j,T-1}$$

$$c_{T-1} \begin{cases} := \sum_{j \in J} q_{j,T-1} Y_{j,T-1} - \sum_{j \in J} q_{j,T-1} X_{j,T-1} - \\ \quad \sum_{j \in J: X_{j,T-1} > Y_{j,T-1}} c_j^b q_{j,T-1} \left(X_{j,T-1} - Y_{j,T-1} \right) - \\ \quad \sum_{j \in J: Y_{j,T-1} > X_{j,T-1}} c_j^s q_{j,T-1} \left(Y_{j,T-1} - X_{j,T-1} \right) - \sum_{j \in J: X_{j,T-1} \neq Y_{j,T-1}} c_j^f + \kappa_{T-1}, \\ \quad \text{if the tracking portfolio is self-financing;} \\ := \sum_{j \in J} q_{j,T-1} Y_{j,T-1} + \kappa_{T-1} - \sum_{j \in J} q_{j,T-1} X_{j,T-1}, \text{otherwise} \end{cases}$$

$$C_T := \kappa_T + (1 + r_T) c_{T-1} + \sum_{j \in J} Y_{jT} q_{jT}$$

This update is performed after each rebalancing decision until the last period of the investment horizon has been reached. Then, we evaluate the tracking accuracy over the entire investment horizon by calculating the mean-absolute deviation (MAD) between the values of the fund and the benchmark portfolio, and by calculating the root-mean squared error (RMSE) between the returns of the index and the index fund. The value of the benchmark portfolio at time $t \in \{S + 1, \ldots, E\}$ is computed as $B_t = (B_{t-1} + \kappa_{t-1}) \frac{I_t}{I_{t-1}}$, with $B_S = C_S$. This means that the capital of the benchmark portfolio ($B_{t-1} + \kappa_{t-1}$) grows, according to its definition, exactly as the index ($\frac{I_t}{I_{t-1}}$) in every period. Based on the assumption that the units of each stock after rebalancing represent the tracking portfolio in the next period, the value of the fund is computed as $P_t = \sum_{j \in J} X_{j,t-1} q_{jt} + (1 + r_t) c_{t-1}$.

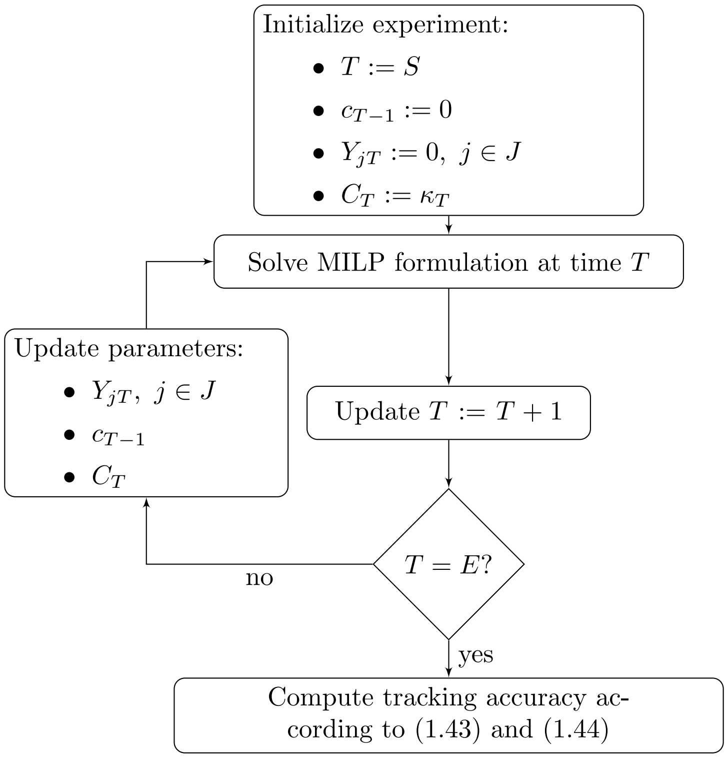


Figure 1.3: Flowchart of the experiment.

The MAD and the RMSE are then calculated as

$$\text{MAD} = \frac{1}{E-S} \sum_{t=S+1}^{E} |B_t - P_t| \tag{1.43}$$

$$\text{RMSE} = \sqrt{\frac{1}{E-S} \sum_{t=S+1}^{E} \left(\ln\left(\frac{B_t}{B_{t-1}}\right) - \ln\left(\frac{P_t}{P_{t-1}}\right) \right)^2} \tag{1.44}$$

1.6 Illustrative example

In this section, we give an example of how the proposed MILP formulation (cf. Section 1.3) and the two existing MILP formulations (cf. Section 1.4) are applied to track an index. The purpose of this example is threefold. First, it gives insight into why existing formulations may fail to construct the best possible tracking portfolio. Second, it illustrates the design of the computational experiment presented in Section 1.5. Third, it is helpful for the readers to reproduce the results presented in Section 1.7.

We consider an index composed of $n = 4$ stocks, and a time horizon that spans $E = 12$ weeks. The investment horizon starts at $S = 6$. Except for S, E, and the parameters δ_{jT}

and η_{jT}, which we set to 0.5 for all stocks $j \in \{1, \ldots, 4\}$ and for all decisions over the investment horizon at the time points $T \in \{6, \ldots, 11\}$, we used the parameters as defined in the experimental design (see Table 1.4 in Section 1.5).

The closing prices of the four stocks q_{jt} and the value of the index I_t at the end of each of the 12 weeks are shown in Table 1.5. The prices q_{jt} stem from the first four stocks of the smallest problem instance introduced by Beasley et al. (2003), which contains weekly prices of the stocks in the Hang Seng index. We computed the index values based on the assumption that each of the four stocks is represented by the same number of units in the index. The number of units was chosen such that the index has a value of 100 at the start of the investment horizon $S = 6$, i.e., each stock is represented by approximately 1.7015 units. Table 1.5 also shows the cash changes κ_t that occur during the investment horizon and the values of the benchmark portfolio B_t, which are computed based on the index values and the cash changes. The last column of Table 1.5 presents the values of a self-financing tracking portfolio that is invested according to the full-replication strategy. Based on the assumption that all stocks cause the same transaction costs (i.e., $c_j^b = c^b$, $c_j^s = c^s$, $c_j^f = c^f$, $j \in J$), the values in column P_t^{FR} are calculated for the periods $t = S+1, \ldots, E$ as follows (with $P_S^{FR} = 0$):

$$P_t^{FR} = \begin{cases} P_{t-1}^{FR} \frac{I_t}{I_{t-1}}, & \text{if } \kappa_{t-1} = 0 \\ \left(P_{t-1}^{FR} + \frac{\kappa_{t-1} - nc^f}{1+c^b} \right) \frac{I_t}{I_{t-1}}, & \text{if } \kappa_{t-1} > 0 \\ \left(P_{t-1}^{FR} + \frac{\kappa_{t-1} - nc^f}{1-c^s} \right) \frac{I_t}{I_{t-1}}, & \text{if } \kappa_{t-1} < 0 \end{cases} \tag{1.45}$$

The values $\frac{\kappa_{t-1} - nc^f}{1+c^b}$ and $\frac{\kappa_{t-1} - nc^f}{1-c^s}$ denote the maximum possible investment based on the cash deposit and the minimum possible divestment based on the cash withdrawal from investors, respectively, such that the fund's entire capital is invested in the tracking portfolio, i.e., such that the fund has no excess cash.

We applied all three formulations from Sections 1.3 and 1.4 under each of the three investment settings presented in Section 1.2. Table 1.6 summarizes the numerical results. For each combination of model formulation and investment setting, the table presents the tracking accuracy measured by the MAD and the RMSE according to (1.43) and (1.44), respectively, the average absolute difference between the weights of the stocks in the index and the weights of the stocks in the tracking portfolio (WD), the average transaction costs spent (TC), the average portfolio cardinality (PC), and the average excess cash (EC) over the investment horizon.

We first discuss the results for the investment setting *R-EF*, in which transaction costs are paid out of an external account. Under this setting, it is reasonable to follow the

Table 1.5: Stock prices (q_{jt}), index values (I_t), cash flows (κ_t), benchmark-portfolio values (B_t), and portfolio values of a self-financing full-replication strategy (P_t^{FR}) for the illustrative example.

t	q_{1t}	q_{2t}	q_{3t}	q_{4t}	I_t	κ_t	B_t	P_t^{FR}
1	9.34	14.38	7.37	29.47	103.02	–	–	–
2	9.87	14.18	7.40	27.34	100.03	–	–	–
3	9.96	14.59	7.77	28.76	103.93	–	–	–
4	9.16	15.74	7.63	28.76	104.29	–	–	–
5	9.25	16.29	7.49	27.87	103.62	–	–	–
6	8.98	15.62	7.54	26.63	100.00	100	–	–
7	9.07	15.74	7.67	26.98	101.18	0	101.18	100.13
8	8.73	16.99	8.02	27.51	104.23	0	104.23	103.15
9	9.25	18.98	8.66	29.11	112.29	100	112.29	111.13
10	9.34	18.76	9.41	30.89	116.37	0	220.00	217.72
11	10.93	18.90	9.23	32.31	121.44	-50	229.59	227.21
12	10.93	18.98	9.85	31.24	120.81	–	178.66	175.73

full-replication strategy because the index only contains four stocks and the transaction costs do not decrease the index fund's capital. However, only the formulations *VAL* and *NEW* determined a tracking portfolio that follows the full-replication strategy. This can be observed from the corresponding MAD, RMSE, and WD values, which are all zero for these tracking portfolios. In contrast, the composition of the tracking portfolio that was obtained with formulation *RET* differs from the index composition even though all four stocks are included in the tracking portfolio. The different composition results from the underlying assumption of formulation *RET* that the portfolio weights are constant during the historical periods. Another observation is that when formulation *RET* is used, a fraction of the index fund's capital is not invested in the tracking portfolio (excess cash). This occurs because formulation *RET* does not consider the distance between the historical trajectories of the tracking portfolio and the index. The resulting excess cash further decreases the tracking accuracy.

Next, we consider the investment settings *R-SF* and *NR-SF*, in which the tracking portfolios are self-financing. Under these two settings, all formulations are bound to have a positive MAD because the transaction costs must be paid out of the index fund's capital. However, it is still reasonable to follow the full-replication strategy, as the index is small and the transaction costs are relatively low. From all three formulations, only *NEW* determines a tracking portfolio that follows the full-replication strategy as can be seen from $WD = 0$. The composition of the tracking portfolio determined by formulation *RET* is

Table 1.6: Results for the illustrative example: MAD according to (1.43), RMSE according to (1.44), average absolute difference between the weights of the stocks in the tracking portfolio and in the index (WD), average transaction costs (TC), average portfolio cardinality (PC), and average excess cash (EC) over all rebalancing decisions.

	Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*		
	VAL	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*
MAD	0.00	0.25	0.00	2.75	2.07	1.81	2.77	1.96	1.81
RMSE [%]	0.00	0.18	0.00	0.64	0.52	0.49	0.64	0.51	0.49
WD [%]	0.00	1.78	0.00	2.61	1.79	0.00	2.59	1.75	0.00
TC	0.44	0.49	0.44	0.52	0.49	0.44	0.52	0.49	0.44
PC	4.00	4.00	4.00	3.83	4.00	4.00	3.83	4.00	4.00
EC	0.00	1.36	0.00	0.00	1.03	0.00	0.00	0.88	0.00

similar to the one obtained for the *R-EF* investment setting. In contrast to the investment setting *R-EF*, the composition of the tracking portfolio determined by formulation *VAL* now also differs from the index composition even though the tracking portfolio includes all four stocks after the second rebalancing. This occurs because formulation *VAL* selects the tracking portfolio whose historical trajectory has minimum distance to the historical trajectory of the index. The composition of this minimum-distance tracking portfolio is quite different from the index composition, which results in a relatively low tracking accuracy.

1.7 Computational experiment

In this section, we use the proposed optimization criterion *NEW* and the two existing optimization criteria *RET* and *VAL* in the three investment settings *R-EF*, *R-SF*, and *NR-SF* to rebalance the tracking portfolios during the entire investment horizon according to the experimental design described in Section 1.5 for a set of 54 problem instances. For each rebalancing decision, we limited the running time of the solver to 100 seconds. In practice, a longer running-time limit could be used because the index-fund managers do not have to take the decisions within seconds when the portfolio is rebalanced weekly. We set the time limit to 100 seconds to limit the maximum running time of our computational analysis. Notice that there are 54 instances, 52 rebalancing decisions per instance, and nine combinations of optimization criteria and investment settings, resulting in 25'272 optimization problems to solve. Hence, with a time limit of 100 seconds, the total running time for our computational experiment is bounded by approximately one month. We

Table 1.7: Benchmark indices for the problem instances.

Instance	Index	n	k	Source	Time horizon
1/28	Hang Seng	31	10	Beasley et al. (2003)	03/1992–03/1995
2/29	DAX100	85	10	Beasley et al. (2003)	03/1992–03/1995
3/30	FTSE100	89	10	Beasley et al. (2003)	03/1992–03/1995
4/31	S&P100	98	10	Beasley et al. (2003)	03/1992–03/1995
5/32	Nikkei225	225	10	Beasley et al. (2003)	03/1992–03/1995
6/33	S&P500	457	40	Canakgoz and Beasley (2009)	NA
7/34	Russell2000	1,319	70	Canakgoz and Beasley (2009)	NA
8/35	Russell3000	2,152	90	Canakgoz and Beasley (2009)	NA
9/36	SMI	20	10	This paper	07/2012–07/2015
10/37	Hang Seng	49	10	This paper	07/2012–07/2015
11/38	EUROSTOXX50	50	10	This paper	07/2012–07/2015
12/39	FTSE100	96	10	This paper	07/2012–07/2015
13/40	S&P100	99	10	This paper	07/2012–07/2015
14/41	NASDAQ100	101	10	This paper	07/2012–07/2015
15/42	DAX100	102	10	This paper	07/2012–07/2015
16/43	SPI	198	10	This paper	07/2012–07/2015
17/44	Nikkei225	220	10	This paper	07/2012–07/2015
18/45	S&PASX300	254	10	This paper	07/2012–07/2015
19/46	S&P500	489	40	This paper	07/2012–07/2015
20/47	FTSE All Share	567	40	This paper	07/2012–07/2015
21/48	STOXXEURO600	575	40	This paper	07/2012–07/2015
22/49	S&P1200	1,179	70	This paper	07/2012–07/2015
23/50	NASDAQ Composite	2,140	90	This paper	07/2012–07/2015
24/51	FTSE100 (down-down)	100	10	Guastaroba et al. (2009)	2000–2003
25/52	FTSE100 (down-up)	100	10	Guastaroba et al. (2009)	2001–2004
26/53	FTSE100 (up-down)	100	10	Guastaroba et al. (2009)	1998–2001
27/54	FTSE100 (up-up)	100	10	Guastaroba et al. (2009)	1995–1998

implemented the MILP formulations in Java and used the Gurobi solver 7.0 for the optimization. All computations were performed on an HP Z800 workstation with two 3.1GHz Intel Xeon CPUs and 128 GB of RAM. In Subsection 1.7.1, we describe the problem instances. In Subsection 1.7.2, we discuss the numerical results.

1.7.1 Problem instances

Our test set consists of 54 problem instances that are all derived from real-world data. Each instance consists of the closing prices of n stocks for 156 periods, the index values for all 156 periods, the cash changes that occur during the last 52 periods (during the investment horizon) and parameter k, which denotes the maximum number of stocks that are allowed to be included in the tracking portfolio. Parameter k is only relevant for the investment settings *R-EF* and *R-SF*.

The first 27 problem instances differ from the second 27 problem instances only with respect to the cash changes κ_T that occur at the time points $T = S+1, \dots, E-1$. For the first 27 problem instances, we assumed that no cash changes occurred ($\kappa_T = 0,\ T = S+1, \dots, E-1$). For the last 27 test instances, we assumed that the cash changes followed a normal distribution with mean 20 and standard deviation 25 ($\kappa_T \sim \mathcal{N}(20, 25), T \in \{S+1, \dots, E-1\}$). This means that each cash change at the time points $T = S+1, \dots, E-1$ was simulated by drawing a random number from the distribution $\mathcal{N}(20, 25)$. Chiam et al. (2013) also used the normal distribution to simulate the cash changes, but they used a mean of zero and a rather small standard deviation. Because such small cash changes make it difficult to investigate the effects of cash changes on the tracking accuracy, we used a higher standard deviation. To reduce the probability that the fund's capital reaches a value of zero due to cash out-flows, we used a positive mean. In the following, we describe the first 27 problem instances in more detail.

Instances 1–8 correspond to the instances that were introduced by Beasley et al. (2003) and Canakgoz and Beasley (2009). These eight instances, which can be downloaded from the OR-Library (cf. Beasley, 1990), contain weekly closing prices and index values of 291 weeks from eight different market indices. Only stocks that were listed in the index during all 291 weeks were included. Following Guastaroba and Speranza (2012), we construct the problem instances by selecting the stock prices and index values from the first 156 periods of their instances. For parameter k, we chose the same values as in Guastaroba and Speranza (2012).

Instances 9–23 are new instances that we constructed for this paper. Following Beasley et al. (2003) and Canakgoz and Beasley (2009), we used DATASTREAM to obtain 156 weekly closing prices of the stocks that constituted different stock-market indices from July 2012 to July 2015. Analogous to Beasley et al. (2003) and Canakgoz and Beasley (2009), we also disregard stocks with incomplete price data during this period. For k, we selected an integer from the set $\{10, 40, 70, 90\}$ according to the first eight problem instances such that instances with a similar number of stocks n have the same k.

Instances 24–27 correspond to the instances that were introduced by Guastaroba et al. (2009). These four instances, which can be downloaded at the website `http://or-brescia.unibs.it/instances`, contain weekly closing prices and index values of 156 weeks from the FTSE100 index. The four instances are characterized by either a downward or an upward market trend in the first 104 weeks and either a downward or an upward market trend in the following 52 weeks. For example, instance 26 is characterized by an upward trend in the first 104 weeks and a downward trend in the following 52 weeks due to the burst of the dot-com bubble. Hence, this instance can be used to examine the tracking accuracy in periods of market stresses. The value for parameter k is set to ten for all of the four instances.

Table 1.7 lists for each instance the name of the index, the number of stocks n, parameter k, the source, and, if known, the time horizon from which the data were collected.

1.7.2 Results

We applied all three formulations under each investment setting to all 54 problem instances. In addition, we computed the tracking accuracy of the self-financing full-replication strategy *FR* for each problem instance. The values of the full-replication tracking portfolio were calculated according to (1.45). Tables 1.8 and 1.9 present the MAD according to (1.43) of the resulting index funds over the investment horizon for the problem instances without cash changes (instances 1–27) and the problem instances with cash changes (instances 28–54), respectively. The lowest MAD per instance and investment setting is shown in bold. From Tables 1.8 and 1.9, we can gain the following main insights:

- The proposed formulation *NEW* achieved the best average tracking accuracy in the investment settings *R-SF* and *NR-SF* with self-financing tracking portfolios. In the investment setting *R-EF*, all three formulations achieved a similar tracking accuracy.
- All formulations achieved a higher average tracking accuracy in the non-restrictive investment setting *NR-SF* than in the restrictive investment setting *R-SF*. The difference in tracking accuracy increased when cash changes occurred. This finding highlights the impact of user-defined parameters such as k on the tracking accuracy. A careless selection of these parameters may significantly reduce the tracking accuracy.
- The tracking accuracy of the self-financing full-replication strategy decreased as the size of the problem instance increased in terms of number of stocks n. Moreover, as shown in Table 1.9, the tracking accuracy of the full-replication strategy significantly decreased when cash changes occurred. This decrease in tracking accuracy is caused by transaction costs. Whenever a cash change occurs, each stock must be traded (either purchased or sold) to ensure that the fund has no excess cash and that the stock weights in the tracking portfolio coincide with the stock weights in the index. Hence, fixed and proportional transaction costs are paid for each stock. These transaction costs negatively impact the tracking accuracy and increase with an increasing number of stocks in the index and with an increasing frequency of cash changes. The following example illustrates this effect of cash changes: Suppose the capital of the fund is USD 100’000, the index consists of 2’000 stocks, and the fixed

Table 1.8: MAD (cf. (1.43)) without cash changes over the investment horizon.

	Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*			
Instance	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*FR*
1	3.60	3.10	**1.17**	6.18	**1.74**	4.73	5.69	4.71	**0.96**	1.13
2	1.68	**0.50**	1.15	10.88	7.26	**2.64**	7.65	11.40	**3.57**	1.94
3	1.62	**1.55**	3.23	**5.58**	11.45	6.27	7.37	20.08	**0.54**	1.87
4	3.53	**0.94**	4.86	8.09	9.51	**6.86**	10.62	22.61	**1.28**	2.11
5	**2.53**	4.21	2.80	10.39	13.94	**3.87**	14.20	32.11	**4.47**	3.84
6	**2.25**	4.81	2.71	21.20	40.17	**2.74**	19.34	30.08	**2.16**	7.73
7	4.68	**2.87**	4.06	16.76	60.87	**6.37**	13.98	17.44	**3.19**	18.89
8	3.56	6.95	**2.90**	19.84	65.83	**2.58**	16.61	19.15	**1.29**	31.85
9	1.35	1.48	**1.03**	5.79	8.25	**3.15**	5.05	3.71	**3.67**	1.25
10	2.31	2.14	**1.29**	13.93	5.88	**1.71**	7.38	12.33	**0.72**	1.66
11	1.64	**1.35**	2.75	10.03	6.97	**3.43**	8.14	12.07	**2.86**	1.62
12	2.99	**0.98**	4.74	7.95	8.44	**0.87**	9.53	17.39	**1.32**	2.09
13	4.61	1.93	**0.97**	7.66	6.74	**4.01**	8.93	24.84	**3.17**	2.22
14	**1.34**	3.76	3.29	8.32	10.32	**3.86**	12.85	18.12	**4.20**	2.37
15	1.20	1.51	**0.91**	9.30	**5.53**	6.58	7.91	11.34	**1.89**	2.30
16	3.71	2.43	**1.55**	7.54	8.19	**7.09**	7.38	7.24	**4.34**	3.40
17	1.37	3.93	**1.32**	11.43	15.20	**6.45**	16.98	27.05	**4.68**	4.13
18	2.84	3.38	**2.05**	**5.97**	15.30	9.28	8.98	18.40	**2.06**	4.05
19	1.52	**0.49**	2.34	14.94	35.42	**3.49**	13.68	33.79	**1.74**	6.99
20	**0.56**	0.64	0.64	15.15	27.42	**1.81**	16.71	25.69	**2.38**	7.65
21	3.30	**0.93**	1.05	18.42	36.72	**2.30**	17.80	31.02	**1.31**	8.16
22	**0.83**	1.46	0.94	17.59	57.18	**3.40**	16.25	28.30	**2.45**	14.73
23	**2.20**	6.93	3.71	34.31	64.20	**5.11**	22.69	27.41	**15.80**	27.97
24	3.55	**3.38**	4.43	6.99	9.08	**4.72**	6.76	14.55	**0.51**	1.92
25	5.75	**2.08**	5.26	11.16	**10.83**	11.62	8.13	15.91	**2.58**	2.67
26	**4.53**	11.37	9.64	11.45	**2.89**	5.38	6.77	8.02	**4.15**	2.00
27	**1.65**	7.32	4.30	13.70	17.16	**2.98**	13.19	20.71	**4.34**	2.61
∅	**2.62**	3.05	2.78	12.24	20.83	**4.57**	11.50	19.09	**3.02**	6.26

Table 1.9: MAD (cf. (1.43)) with cash changes over the investment horizon.

	Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*			
Instance	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*FR*
28	**4.15**	10.82	5.98	17.78	**4.45**	28.44	15.48	8.96	**6.82**	15.79
29	4.89	**2.73**	8.14	35.14	18.87	**16.74**	20.60	22.08	**11.61**	33.24
30	10.43	**8.81**	8.88	48.96	47.85	**6.13**	27.99	39.64	**4.38**	34.75
31	4.72	6.93	**4.23**	71.74	30.14	**19.00**	30.31	33.58	**6.76**	38.59
32	**5.35**	5.37	14.46	82.60	60.06	**15.19**	32.02	40.47	**9.30**	76.34
33	25.92	8.45	**7.80**	27.11	69.44	**15.25**	39.41	37.02	**7.03**	160.33
34	**12.54**	30.52	19.07	48.11	137.02	**40.33**	35.95	14.15	**10.34**	– *
35	22.70	22.86	**9.49**	**16.96**	146.76	18.68	29.38	15.06	**9.47**	– *
36	**7.37**	7.97	7.60	22.40	32.45	**4.83**	15.65	13.36	**10.03**	13.80
37	12.90	8.94	**7.19**	50.70	10.65	**6.63**	21.05	28.20	**7.25**	23.96
38	**4.40**	9.37	10.34	46.50	27.35	**22.99**	25.25	31.20	**11.40**	24.60
39	**2.97**	8.63	6.32	57.17	34.72	**19.22**	29.42	31.51	**9.20**	38.07
40	**3.54**	5.93	11.88	40.70	43.42	**10.32**	29.09	41.95	**10.52**	39.35
41	9.28	11.53	**8.87**	37.52	34.17	**12.35**	32.56	30.21	**8.74**	41.22
42	17.02	**9.75**	21.84	50.11	20.43	**15.79**	29.08	27.55	**7.78**	43.87
43	18.58	9.45	**9.26**	45.31	36.00	**24.16**	19.95	14.86	**12.71**	71.75
44	**6.05**	12.04	15.33	87.47	64.02	**10.71**	41.16	32.40	**16.41**	86.78
45	**4.40**	10.37	12.85	43.79	40.84	**29.80**	24.60	25.98	**12.26**	89.52
46	5.14	3.29	**3.23**	43.65	116.49	**10.30**	26.53	34.93	**10.25**	166.75
47	4.36	1.97	**1.87**	37.95	57.06	**10.53**	28.84	28.37	**10.73**	190.95
48	7.04	**2.19**	3.74	34.05	116.88	**8.84**	29.41	37.84	**3.63**	205.32
49	10.68	7.50	**5.64**	39.65	162.40	**12.26**	30.57	24.62	**11.82**	385.25
50	**10.87**	19.33	15.63	84.34	153.54	**25.25**	37.73	34.81	**23.16**	– *
51	**8.76**	14.98	9.43	28.98	13.71	**7.73**	20.08	30.12	**3.18**	34.30
52	**3.21**	12.35	5.10	44.52	32.20	**26.22**	24.80	24.90	**6.72**	41.51
53	18.01	17.15	**9.84**	53.79	10.86	**10.24**	10.82	7.64	**7.48**	38.49
54	**7.10**	29.22	11.47	70.08	67.12	**23.75**	33.32	47.24	**7.83**	42.79
∅	**9.35**	11.05	9.46	46.93	58.85	**16.73**	27.45	28.10	**9.51**	80.72

*The index fund's capital was not sufficient to cover the total transaction costs transaction costs are USD 10 per trade. Whenever a cash change occurs, all of the 2'000 stocks must be traded, which causes fixed transaction costs of USD 20'000 and some proportional transaction costs. After few cash changes, the index fund's capital may not be large enough to cover the total transaction costs. This happened for the largest instances with cash changes, i.e., instances 34, 35, and 50. We therefore marked these instances in Table 1.9 by a dash.

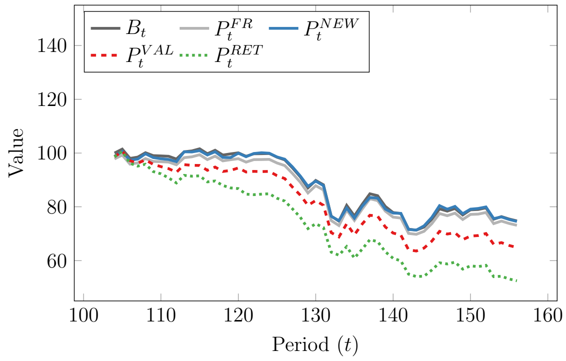
- A comparison between the MADs of the self-financing full-replication strategy and the MADs of the MILP formulations in the investment setting *NR-SF* demonstrates that the benefits of using optimization compared to a naïve approach increase with an increasing number of index constituents and when cash changes occur.

- The MADs for the instances 24–27 and 51–54 show that for instances that contain periods of market stresses, the tracking accuracy of all formulations is comparable to the tracking accuracy achieved for the other instances. For the instances 24–27, Figure 1.4 depicts the value developments over the investment horizon of the benchmark portfolio, the full-replication strategy, and the index funds by using the different formulations in the proposed *NR-SF* investment setting.

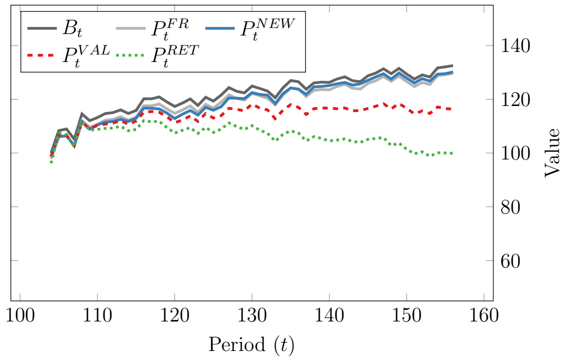
Next, we investigate the risk-return characteristics of the index funds. Table 1.10 shows the following risk-return characteristics as averages over the problem instances with and without cash changes: the root-mean squared error (RMSE) between the returns of the index fund and the returns of the index according to (1.44), the mean return of the index fund and the index (MEAN), the volatility of the returns of the index fund and the index (VOL), and the skewness (SKEW) and kurtosis (KURT) of the distributions of the returns of the index fund and the index. From Table 1.10, we can draw the following main conclusions:

- Similar to the results based on the MAD (cf. Tables 1.8 and 1.9), the proposed formulation *NEW* achieved the highest tracking accuracy in terms of the RMSE in the investment settings *R-SF* and *NR-SF* with self-financing tracking portfolios. In the investment setting, *R-EF*, the return-based formulation achieved the best RMSE.

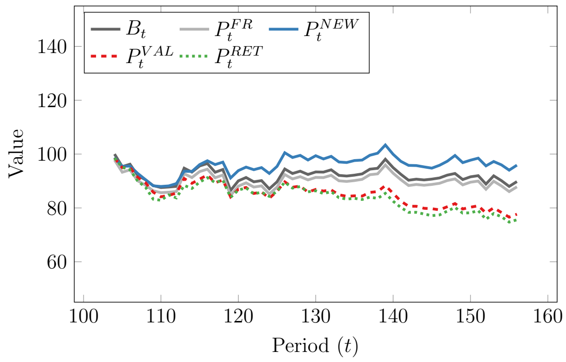
- The lower tracking accuracy in terms of RMSE of the existing formulations compared to the proposed formulation in the investment settings with self-financing tracking portfolios can be, at least partially, attributed to the mean returns that are considerably below the mean index return and the mean return obtained by using the new formulation.



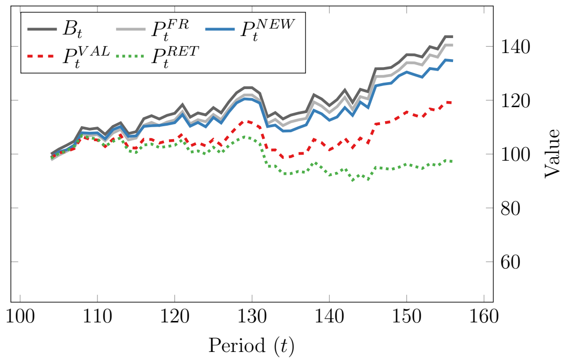
(a) Instance 24; down-down market trend



(b) Instance 25; down-up market trend



(c) Instance 26; up-down market trend



(d) Instance 27; up-up market trend

Figure 1.4: Value developments over the investment horizon of the benchmark portfolio (B_t), the full-replication approach (P_t^{FR}), and the index fund by applying the different formulations *NEW*, *VAL*, and *RET* (P_t^{NEW}, P_t^{VAL}, and P_t^{RET}, respectively) in the investment setting *NR-SF* for the problem instances introduced by Guastaroba et al. (2009).

Table 1.10: Root mean squared error (RMSE) between the returns of the index fund and the index according to (1.44), volatility of the returns of the index fund and the index (VOL), mean return of the index fund and the index (MEAN), and skewness (SKEW) and kurtosis (KURT) of the distributions of the returns of the index fund and the index; averages over the problem instances with and without cash changes; all calculations based on continuously compounded returns.

		Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*			
		VAL	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*Index*
Inst. 1–27	RMSE [%]	0.67	0.61	0.65	0.89	1.62	0.80	0.57	0.94	0.56	–
	MEAN [%]	0.14	0.09	0.08	−0.38	−1.12	−0.04	−0.30	−0.60	0.05	0.09
	VOL [%]	2.23	2.11	2.27	2.26	2.36	2.28	2.23	2.24	2.24	2.23
	SKEW	−0.20	−0.14	−0.26	−0.25	−0.26	−0.33	−0.32	−0.29	−0.31	−0.25
	KURT	1.44	1.60	1.60	1.51	1.70	1.31	1.57	1.60	1.49	1.49
Inst. 28–57	RMSE [%]	0.65	0.62	0.66	0.92	1.00	0.75	0.49	0.56	0.42	–
	MEAN [%]	0.11	0.07	0.11	−0.38	−0.54	−0.05	−0.20	−0.22	0.01	0.09
	VOL [%]	2.25	2.09	2.26	2.25	2.15	2.23	2.22	2.22	2.21	2.23
	SKEW	−0.29	−0.17	−0.31	−0.25	−0.17	−0.30	−0.27	−0.27	−0.30	−0.25
	KURT	1.69	1.63	1.68	1.52	1.66	1.18	1.48	1.47	1.52	1.52

- The risk measures volatility, kurtosis, and skewness obtained by using all three formulations in all three investment settings are similar to the values of the respective risk measures of the index. This shows that all formulations are capable of replicating the risk characteristics of the index.

Figure 1.5 provides additional information regarding the risk-return characteristics. The three figures in the left column show a scatter plot for each investment setting in which each point represents an instance. The y-axis represents the difference between the mean return of the index funds and the index and the x-axis represents the difference between the volatility of the returns of the index funds and the index. The risk-return characteristics of the index are perfectly replicated if both the difference in the mean return and the difference in the volatility is zero. The three figures in the right column show a box plot for each investment setting of the RMSEs obtained by using the different formulations. From Figure 1.5 and Table 1.10, we can gain the following main insights:

- In the investment settings with self-financing portfolios, the proposed formulation achieves the smallest median RMSE and the RMSEs for the different problem instances tend to be more similar compared to the existing formulations as can be seen from the range between the first-quartile RMSE and the third-quartile RMSE in Figures 1.5d and 1.5f.

- The formulation *RET* tends to lead to index funds that are less riskier in terms of volatility, especially in the investment setting *R-EF*, than the index (see Figure 1.5a). This can also be seen in Table 1.10.

- The mean returns obtained by using the existing formulations tend to be considerably below the mean returns obtained by the proposed formulation and the mean returns of the index (see Figures 1.5c and 1.5e and Table 1.10).

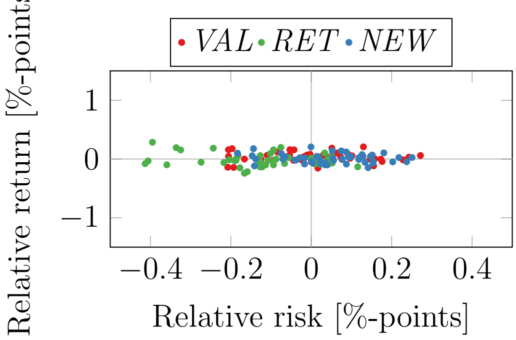
Further, we investigate the characteristics of the tracking portfolios that were determined by the different formulations under the different investment settings. Table 1.11 shows the average transaction costs (TC), the average portfolio cardinality (PC), and the average excess cash (EC). The averages were computed over all periods of the investment horizon once for all problem instances without cash changes and once for all problem instances with cash changes. From Table 1.11, we can draw the following conclusions:

- The proposed formulation *NEW* did not generate excess cash. The index fund's capital was either invested in the tracking portfolio or spent for transaction costs. This is a desirable property because a large amount of excess cash may deteriorate the tracking accuracy when the index returns differ from the interest rate on excess cash (cf. Chiam et al., 2013).

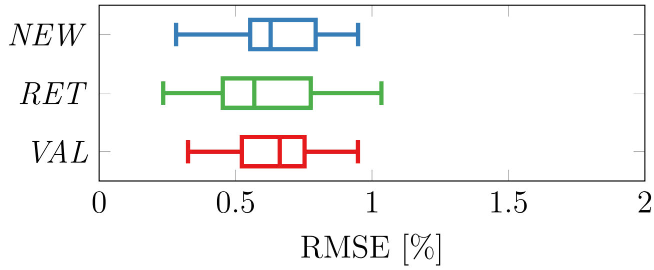
- In terms of portfolio cardinality and transaction costs, all formulations determined similar tracking portfolios in the investment setting *R-EF*, in which transaction costs are paid out of a separate account. This similarity can be explained by the restrictions such as the maximum portfolio cardinality and the tight ranges for the investments in each stock that are imposed in the restrictive investment setting *R-EF*.

- In the investment settings *R-SF* and *NR-SF*, in which tracking portfolios are self-financing, the tracking portfolios obtained by the formulation *VAL* were still similar to those obtained with the formulation *RET*. In contrast, the tracking portfolios obtained by the formulation *NEW* had a lower cardinality, particularly in the non-restrictive setting, and caused less transaction costs. When combining these findings with the previous findings regarding the MADs and the RMSEs, we can conclude that the inferior tracking accuracy of formulations *VAL* and *RET* may be attributed in part to the higher transaction costs that were caused by a higher portfolio cardinality or by rebalancing the tracking portfolio more often and to a larger extent.

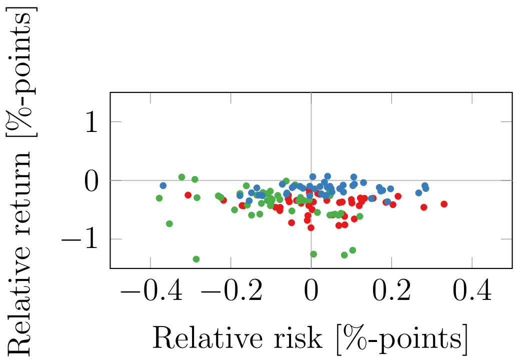
Further, we investigate the computational efficiency of the different formulations. Table 1.12 shows the average CPU time in seconds (CPU) and the average relative MIP



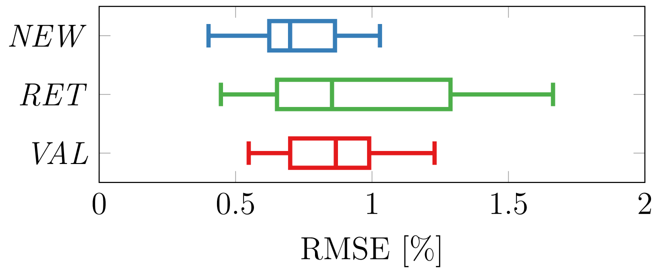
(a) *R-EF* investment setting



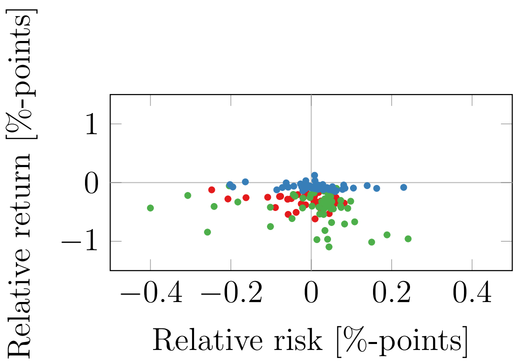
(b) *R-EF* investment setting



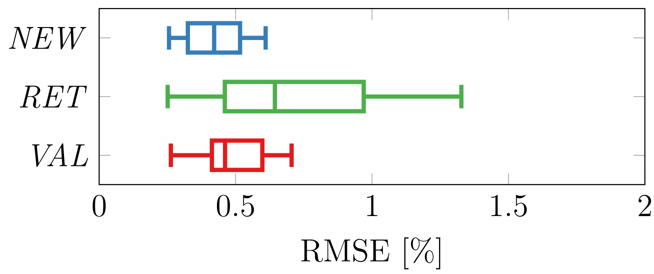
(c) *R-SF* investment setting



(d) *R-SF* investment setting



(e) *NR-SF* investment setting



(f) *NR-SF* investment setting

Figure 1.5: Risk-return characteristics relative to the index for all 54 instances; relative risk: difference between the volatility of the returns of the index fund and the index; relative return: difference between the mean return of the index fund and the index; RMSE: root-mean squared error according to (1.44); outliers regarding the RMSE were removed in all charts.

Table 1.11: Average transaction costs (TC), portfolio cardinality (PC), and excess cash (EC) over all rebalancing decisions and problem instances with cash changes (Instances 1–27) and without cash changes (Instances 28–54).

		Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*		
		VAL	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*
1–27	TC	0.7	0.6	0.9	0.5	0.7	0.2	0.4	0.6	0.1
	PC	22.8	23.5	22.9	23.0	23.0	15.6	104.5	111.0	32.4
	EC	0.5	10.5	0.0	0.5	9.2	0.0	0.0	0.4	0.0
28–54	TC	4.1	2.9	4.8	2.3	2.7	0.8	1.5	1.3	0.4
	PC	22.6	23.2	22.6	23.3	23.0	22.0	114.0	110.8	65.6
	EC	3.4	59.8	0.0	3.0	54.7	0.0	0.1	3.3	0.0

Table 1.12: Average CPU time in seconds (CPU) and relative MIP gap in % (GAP) over all rebalancing decisions and problem instances with cash changes (Instances 1–27) and without cash changes (Instances 28–54).

		Setting: *R-EF*			Setting: *R-SF*			Setting: *NR-SF*		
		VAL	*RET*	*NEW*	*VAL*	*RET*	*NEW*	*VAL*	*RET*	*NEW*
1–27	CPU	35.4	57.3	**34.1**	34.1	55.8	**9.3**	67.4	67.9	**8.6**
	GAP	30.1	35.5	**29.9**	26.7	34.4	**0.3**	20.9	20.5	**0.3**
28–54	CPU	47.0	67.3	**46.1**	45.6	67.4	**20.3**	64.0	61.1	**31.8**
	GAP	33.7	40.0	**33.2**	31.3	40.2	**1.9**	22.9	19.6	**4.6**

gap in % (GAP), again computed over all periods of the investment horizon once for all problem instances without cash changes and once for all problem instances with cash changes. Bold values denote the shortest CPU time and the lowest relative MIP gap for each investment setting. The results in Table 1.12 show that the formulation *NEW* has the shortest CPU times and the lowest relative MIP gaps compared to the other formulations. Hence, the proposed formulation is computationally the most efficient one. This higher efficiency can be explained in part by the lower integrality gaps obtained when using the formulation *NEW* rather than the two existing formulations (cf. Table 1.13). These integrality gaps are computed based on the objective function value of the best feasible solution after the CPU time limit ($BEST_{int}$) and the objective function value obtained by solving the linear programming relaxation (OPT_{frac}) as $\frac{BEST_{int}-OPT_{frac}}{BEST_{int}}$.

Finally, we investigate the benefit of using the valid inequality (1.26) in the investment setting *NR-SF*. For this purpose, we compare the optimal objective function value of the linear programming relaxation of the three considered formulations for the initial portfo-

Table 1.13: Average integrality gap in [%] computed based on the objective function value of the best feasible solution after the CPU time limit ($BEST_{int}$) and the objective function value obtained by solving the linear programming relaxation (OPT_{frac}): $\frac{BEST_{int}-OPT_{frac}}{BEST_{int}}$.

		VAL	*RET*	*NEW*
	R-EF	71.08	77.67	68.68
Instances 1–27	*R-SF*	64.90	76.79	21.69
	NR-SF	51.51	41.47	15.39
	R-EF	78.67	79.08	77.87
Instances 28–54	*R-SF*	73.20	78.84	27.93
	NR-SF	51.56	40.21	22.71

lio construction at time S when using the valid inequality (OFV_{VI}) with the respective optimal objective function value when the valid inequality is ignored (OFV). The relative increase in the objective function value computed as $\frac{\text{OFV}_{\text{VI}}-\text{OFV}}{\text{OFV}}$ ranges from 0.01% to 0.23% for the formulations *VAL* and *NEW*; even though this is a small relative increase, it demonstrates that the valid inequality leads to a tighter linear programming relaxation. For the formulation *RET*, the valid inequality had no influence on the linear programming relaxation.

1.8 Conclusions

In this paper, we considered the index-tracking problem, which is the problem of replicating the performance of a financial index as accurately as possible. We presented a new MILP formulation for the index-tracking problem that leads to a high similarity in terms of the normalized historical value developments between the tracking portfolio and the index, and to low rebalancing costs. In a computational experiment based on a set of real-world problem instances, we demonstrated that the proposed formulation is superior to existing formulations in the literature in terms of tracking accuracy and computational efficiency. We also compared the MILP formulations to the naïve full-replication strategy, which indicated that the benefits that can be obtained in practice by using optimization increase with an increasing number of index constituents and increasing cash changes.

In future research, it could be promising to employ the methods presented in this paper in other fields of portfolio optimization, such as enhanced-index tracking (cf., e.g., Guastaroba et al., 2016). For this purpose, the index values can be multiplied by a factor that represents the target excess return. Then, this modified index can be tracked by using the method proposed in this paper. Tracking this modified index corresponds to outperforming the original index.

Appendix

1.A Proofs

Proof of Proposition 1. We prove both directions of implication.

($\Rightarrow$): Let us assume:

$$\frac{Q_t(\mathbf{x})}{Q_{t-1}(\mathbf{x})} - 1 = \frac{I_t}{I_{t-1}} - 1 \ (t \in \{2, \ldots, T\}) \tag{1.46}$$

Thus, we have $\frac{Q_{t-1}(\mathbf{x})}{I_{t-1}} = \frac{Q_t(\mathbf{x})}{I_t}$, $t \in \{2, \ldots, T\}$, and hence, $\frac{Q_1(\mathbf{x})}{I_1} = \frac{Q_2(\mathbf{x})}{I_2} = \ldots = \frac{Q_{T-1}(\mathbf{x})}{I_{T-1}} = \frac{Q_T(\mathbf{x})}{I_T}$. Therefore, based on (1.46), we obtain

$$Q_t(\mathbf{x}) = I_t \frac{Q_T(\mathbf{x})}{I_T} \ (t \in \{1, \ldots, T\}) \tag{1.47}$$

Hence, we can replace $Q_t(\mathbf{x})$ based on (1.47) in the distance between the normalized historical trajectories of the tracking portfolio and the index, which proves the forward direction:

$$\sum_{t=1}^{T} \left| \frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})} - \frac{I_t}{I_T} \right| = \sum_{t=1}^{T} \left| \frac{I_t}{I_T} - \frac{I_t}{I_T} \right| = 0 \tag{1.48}$$

($\Leftarrow$): Let us assume that $\sum_{t=1}^{T} \left| \frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})} - \frac{I_t}{I_T} \right| = 0$, and therefore:

$$Q_t(\mathbf{x}) - I_t \frac{Q_T(\mathbf{x})}{I_T} = 0 \ (t \in \{1, \ldots, T\})$$

$$\Rightarrow \frac{Q_t(\mathbf{x})}{Q_T(\mathbf{x})} = \frac{I_t}{I_T} \ (t \in \{1, \ldots, T\}) \tag{1.49}$$

$$\Rightarrow \frac{Q_T(\mathbf{x})}{Q_{t-1}(\mathbf{x})} = \frac{I_T}{I_{t-1}} \ (t \in \{2, \ldots, T \ + 1\}) \tag{1.50}$$

By multiplying for all $t \in \{2, \ldots, T\}$ the left-hand side and the right-hand side of (1.49) by the left-hand side and the right-hand side of (1.50), respectively, we obtain $\frac{Q_t(\mathbf{x})}{Q_{t-1}(\mathbf{x})} = \frac{I_t}{I_{t-1}}$, $t \in \{2, \ldots, T\}$, which proves the reverse direction. $\square$

Proof of Proposition 2. To see why Proposition 2 holds, we add $\frac{C_T - Q_T(\mathbf{x}^*)}{I_T} \sum_{t=1}^{T} I_t$ to both sides of the above inequality:

$$\begin{aligned}
&\sum_{t=1}^{T}\left|Q_t(\mathbf{x}^*) - I_t\frac{Q_T(\mathbf{x}^*)}{I_T}\right| < \frac{Q_T(\mathbf{x}^*)}{I_T}\sum_{t=1}^{T} I_t \\
&\Rightarrow \sum_{t=1}^{T}\left|Q_t(\mathbf{x}^*) - I_t\frac{Q_T(\mathbf{x}^*)}{I_T}\right| + \frac{C_T - Q_T(\mathbf{x}^*)}{I_T}\sum_{t=1}^{T} I_t < \\
&\quad \frac{Q_T(\mathbf{x}^*)}{I_T}\sum_{t=1}^{T} I_t + \frac{C_T - Q_T(\mathbf{x}^*)}{I_T}\sum_{t=1}^{T} I_t = \frac{C_T}{I_T}\sum_{t=1}^{T} I_t \\
&\Rightarrow f(\mathbf{x}^*) < f(\mathbf{0})
\end{aligned}$$

□

Proof of Proposition 3. We first derive two different upper bounds on the value of the tracking portfolio $\sum_{j\in J} q_{jT}X_{jT}$.

The first upper bound is based on the constraints (1.3) that provide a maximum purchasing value for each stock of $v^b_{jT} \leq \eta_{jT}C_T w^b_{jT}$. Based on the constraints (1.2), we can replace v^b_{jT} in the constraints (1.3) by $q_{jT}(X_{jT} - Y_{jT}) + v^s_{jT}$ to obtain the following upper bound on the investment in each stock:

$$q_{jT}X_{jT} \leq q_{jT}Y_{jT} + \eta_{jT}C_T w^b_{jT} - v^s_{jT} \leq q_{jT}Y_{jT} + \eta C_T w^b_{jT}$$

Note that v^s_{jT} can be dropped because it cannot be negative and η_{jT} can be replaced by η because $\eta \geq \eta_{jT}$, $j \in J$, by definition. Hence, we obtain a first upper bound on the value of the tracking portfolio as follows:

$$\sum_{j\in J} q_{jT}X_{jT} \leq \sum_{j\in J} q_{jT}Y_{jT} + \eta C_T \sum_{j\in J} w^b_{jT} \tag{1.51}$$

The second upper bound is based on the budget constraint (1.1). Based on the constraints (1.6), we can replace G_{jT} by $c^b_j v^b_{jT} + c^s_j v^s_{jT} + c^f_j(w^b_{jT} + w^s_{jT})$ in the budget constraint, and rearrange to obtain the following upper bound on the value of the tracking portfolio:

$$\begin{aligned}
\sum_{j\in J} q_{jT}X_{jT} &\leq C_T - \sum_{j\in J}\left(c^b_j v^b_{jT} + c^s_j v^s_{jT} + c^f_j(w^b_{jT} + w^s_{jT})\right) \\
&\leq C_T - \sum_{j\in J}\left(c^b v^b_{jT} + c^s v^s_{jT} + c^f(w^b_{jT} + w^s_{jT})\right) \\
&\leq C_T - c^b\sum_{j\in J} v^b_{jT} - c^f\sum_{j\in J} w^b_{jT}
\end{aligned}$$

Again, v^s_{jT} and w^s_{jT} could be dropped because they cannot be negative. Also, c^f_j, c^b_j, and c^s_j could be replaced by c^f, c^b, and c^s because $c^f \leq c^f_j$, $c^b \leq c^b_j$, and $c^s \leq c^s_j$, respectively, for all $j \in J$ by definition.

As above, we can again replace v^b_{jT} based on (1.2), and remove v^s_{jT} to obtain:

$$\begin{aligned}\sum_{j\in J} q_{jT}X_{jT} &\leq C_T - c^b \sum_{j\in J} \left(q_{jT}(X_{jT} - Y_{jT}) + v^s_{jT}\right) - c^f \sum_{j\in J} w^b_{jT} \\ &\leq C_T - c^b \sum_{j\in J} q_{jT}X_{jT} + c^b \sum_{j\in J} q_{jT}Y_{jT} - c^f \sum_{j\in J} w^b_{jT}\end{aligned}$$

By rearranging, we obtain the second upper bound on the value of the tracking portfolio:

$$\sum_{j\in J} q_{jT}X_{jT} \leq \frac{C_T + c^b \sum_{j\in J} q_{jT}Y_{jT} - c^f \sum_{j\in J} w^b_{jT}}{1 + c^b}$$

Note that by definition, $C_T = \kappa_T + \sum_{j\in J} Y_{jT}q_{jT} + c_{T-1}(1 + r_T)$. Hence, we obtain:

$$\sum_{j\in J} q_{jT}X_{jT} \leq \sum_{j\in J} q_{jT}Y_{jT} + \frac{\kappa_T + c_{T-1}(1 + r_T) - c^f \sum_{j\in J} w^b_{jT}}{1 + c^b} \tag{1.52}$$

Now, both upper bounds (1.51) and (1.52) depend on the number of stocks that are purchased $b = \sum_{j\in J} w^b_{jT}$. Then, we can write the two upper bounds (1.51) and (1.52) as follows:

$$\sum_{j\in J} q_{jT}X_{jT} \leq \sum_{j\in J} q_{jT}Y_{jT} + \eta C_T b \tag{1.53}$$

$$\sum_{j\in J} q_{jT}X_{jT} \leq \sum_{j\in J} q_{jT}Y_{jT} + \frac{\kappa_T + c_{T-1}(1 + r_T) - c^f b}{1 + c^b} \tag{1.54}$$

Since the right-hand side of (1.53) and (1.54) is monotonically increasing and decreasing in b, respectively, the highest value for $\sum_{j\in J} q_{jT}X_{jT}$ can be obtained when both right-hand sides take the same value, i.e.,

$$\eta C_T b = \frac{\kappa_T + c_{T-1}(1 + r_T) - c^f b}{1 + c^b} \iff b = \frac{\kappa_T + c_{T-1}(1 + r_T)}{(1 + c^b)\eta C_T + c^f}$$

We know that the number of purchased stocks b must be integer, i.e., either $\lceil b \rceil$ or $\lfloor b \rfloor$ stocks are purchased, and that any tracking portfolio must satisfy both constraints (1.53) and (1.54). Hence, if $\lceil b \rceil$ stocks are purchased, the value of the tracking portfolio cannot exceed the following term:

$$\min \left\{ \sum_{j \in J} q_{jT} Y_{jT} + \eta C_T \lceil b \rceil , \sum_{j \in J} q_{jT} Y_{jT} + \frac{\kappa_T + c_{T-1}(1 + r_T) - c^f \lceil b \rceil}{1 + c^b} \right\}$$

Analogously, for a purchase of $\lfloor b \rfloor$ stocks, we obtain the following term:

$$\min \left\{ \sum_{j \in J} q_{jT} Y_{jT} + \eta C_T \lfloor b \rfloor , \sum_{j \in J} q_{jT} Y_{jT} + \frac{\kappa_T + c_{T-1}(1 + r_T) - c^f \lfloor b \rfloor}{1 + c^b} \right\}$$

Hence, in any case, the value of the tracking portfolio cannot exceed the maximum of both terms above, which proves (1.26). □

Bibliography

Andriosopoulos, K., Nomikos, N., 2014. Performance replication of the Spot Energy Index with optimal equity portfolio selection: Evidence from the UK, US and Brazilian markets. European Journal of Operational Research 234 (2), 571–582.

Beasley, J. E., 1990. OR-Library: Distributing test problems by electronic mail. Journal of the Operational Research Society 41 (11), 1069–1072.

Beasley, J. E., Meade, N., Chang, T.-J., 2003. An evolutionary heuristic for the index tracking problem. European Journal of Operational Research 148 (3), 621–643.

Busse, J. A., Goyal, A., Wahal, S., 2010. Performance and persistence in institutional investment management. The Journal of Finance 65 (2), 765–790.

Canakgoz, N. A., Beasley, J. E., 2009. Mixed-integer programming approaches for index tracking and enhanced indexation. European Journal of Operational Research 196 (1), 384–399.

Chiam, S. C., Tan, K. C., Al Mamun, A., 2013. Dynamic index tracking via multi-objective evolutionary algorithm. Applied Soft Computing 13 (7), 3392–3408.

Filippi, C., Guastaroba, G., Speranza, M., 2016. A heuristic framework for the bi-objective enhanced index tracking problem. Omega 65, 122–137.

Gaivoronski, A. A., Krylov, S., Van der Wijst, N., 2005. Optimal portfolio selection and dynamic benchmark tracking. European Journal of Operational Research 163 (1), 115–131.

Guastaroba, G., Mansini, R., Ogryczak, W., Speranza, M. G., 2016. Linear programming models based on Omega ratio for the enhanced index tracking problem. European Journal of Operational Research 251 (3), 938–956.

Guastaroba, G., Mansini, R., Speranza, M. G., 2009. On the effectiveness of scenario generation techniques in single-period portfolio optimization. European Journal of Operational Research 192 (2), 500–511.

Guastaroba, G., Speranza, M. G., 2012. Kernel Search: An application to the index tracking problem. European Journal of Operational Research 217 (1), 54–68.

Investment Company Institue, 2016. 2016 Investment Company Fact Book. Investment Company Institute.

Konno, H., Wijayanayake, A., 2001. Minimal cost index tracking under nonlinear transaction costs and minimal transaction unit constraints. International Journal of Theoretical and Applied Finance 4 (6), 939–957.

Lwin, K. T., Qu, R., MacCarthy, B. L., 2017. Mean-VaR portfolio optimization: A nonparametric approach. European Journal of Operational Research 260 (2), 751–766.

Malkiel, B. G., 1995. Returns from investing in equity mutual funds 1971 to 1991. The Journal of Finance 50 (2), 549–572.

Maringer, D., Oyewumi, O., 2007. Index tracking with constrained portfolios. Intelligent Systems in Accounting, Finance and Management 15 (1–2), 57–71.

Montfort, K. v., Visser, E., van Draat, L. F., 2008. Index tracking by means of optimized sampling. The Journal of Portfolio Management 34 (2), 143–152.

Rudolf, M., Wolter, H.-J., Zimmermann, H., 1999. A linear model for tracking error minimization. Journal of Banking & Finance 23 (1), 85–103.

Sant'Anna, L. R., Filomena, T. P., Guedes, P. C., Borenstein, D., 2017. Index tracking with controlled number of assets using a hybrid heuristic combining genetic algorithm and non-linear programming. Annals of Operations Research 258 (2), 849–867.

Takeda, A., Niranjan, M., Gotoh, J., Kawahara, Y., 2013. Simultaneous pursuit of out-of-sample performance and sparsity in index tracking portfolios. Computational Management Science 10 (1), 21–49.

Valle, C., Meade, N., Beasley, J., 2014. Absolute return portfolios. Omega 45, 20–41.

Valle, C., Meade, N., Beasley, J., 2015. Factor neutral portfolios. OR Spectrum 37 (4), 843–867.

Paper II

A two-stage approach to the UCITS-constrained index-tracking problem[3]

Oliver Strub Norbert Trautmann

Department of Business Administration
University of Bern

Contents

[3]Paper under review, submitted to Computers & Operations Research

Abstract

Undertakings for Collective Investments in Transferable Securities (UCITS) are investment funds that are regulated by the European Union. UCITS have become increasingly popular, resulting in a total corresponding amount of assets under management of 8.5€ trillion by the end of 2016. We present a two-stage approach to the problem of how to construct a portfolio of assets for a UCITS that aims to replicate the returns of a financial index subject to the constraints imposed by the UCITS regulations. In the first stage, we apply a genetic algorithm that treats subsets of the index constituents as individuals to construct a good feasible solution in a short CPU time. In this genetic algorithm, we use a new representation of subsets, which is the first to exhibit all of the following four desirable properties: feasibility, efficiency, locality, and heritability. In the second stage, we apply local branching based on a new mixed-integer quadratic programming formulation to improve the best solution obtained in the first stage. In a numerical experiment on real-world data, the approach yields very good feasible solutions in a short CPU time.

2.1 Introduction

An investment fund is a pool of capital collected from different investors. Professional asset managers invest the collected capital on behalf of the investors in a portfolio of assets such as stocks or bonds. Investment funds that aim to replicate or track the returns of a particular financial index are known as index funds. Index funds are very popular because, compared with investment funds that aim to achieve an excess return over an index, they are less expensive to manage, which translates into lower fees for the investors, and they often yield higher returns (cf., e.g., Busse et al., 2010; Malkiel, 1995; Montfort et al., 2008). To achieve a small tracking error when replicating index returns, the most intuitive approach is full replication, which requires an investment in all constituents of an index in accordance with the index composition. One drawback of full replication is the high management and transaction costs that arise for indices with many constituents (cf., e.g., Guastaroba and Speranza, 2012; Sharma et al., 2017). By investing in only a small subset of the index constituents, these costs can be reduced substantially.

Undertakings for Collective Investments in Transferable Securities (UCITS) are investment funds that are regulated by the European Union (EU). UCITS have become economically important in recent years, and over 8.5 € trillion in net assets were managed through such funds at the end of 2016 (cf. European Fund and Asset Management Association (EFAMA), 2017); this is comparable to the US $16 trillion scale of the US mutual fund industry (cf. Investment Company Institue, 2017). UCITS are subject to regulatory constraints imposed by the UCITS directive of the European Parliament. As noted by Kolm et al. (2014), such regulatory constraints may present a challenge for asset managers when constructing their portfolios.

We consider the UCITS-constrained index-tracking problem (UCITP) introduced by Krink et al. (2009), which is the problem of how to construct a portfolio for a UCITS index fund, i.e., an index fund regulated by the EU. The objective of the UCITP is to minimize the mean-squared error (MSE) between the returns of the portfolio and the index over a set of historical in-sample periods. The MSE is one of the most widely used measures of tracking error in practice (cf. Corielli and Marcellino, 2006) and in the literature (cf., e.g., Andriosopoulos and Nomikos, 2014; Beasley et al., 2003; Chiam et al., 2013; Maringer and Oyewumi, 2007; Montfort et al., 2008; Sant'Anna et al., 2017a). The underlying assumption motivating the minimization of the MSE for historical in-sample periods is that a small in-sample MSE will also tend to lead to a small MSE in future out-of-sample periods. The UCITP comprises the following constraints. A lower and an upper bound on the number of different assets that can be included in the portfolio are prescribed. In addition, a lower bound on the relative weight of each asset selected for inclusion in the portfolio is prescribed. Finally, the constraints of the UCITS directive must be satisfied. These include a short-selling prohibition and the 5/10/40 concentration rule, which states that the weight of each selected asset must not exceed a lower threshold of 5%, except that the weights of some assets may be increased up to a middle threshold of 10%, provided that the sum of the weights exceeding the lower threshold does not exceed an upper threshold of 40%.

In the literature, two approaches to the UCITP have been proposed: a mixed-integer quadratic programming (MIQP) approach (cf. Scozzari et al., 2013) and an approach based on differential evolution and combinatorial search (cf. Krink et al., 2009). Both approaches yield good feasible solutions to small- and medium-scale instances of the UCITP but fail to do so for large-scale instances. The reason for this is the substantial amount of CPU time required for fine-tuning the portfolio weights by applying differential evolution and for solving the quadratic-programming relaxations. For other optimization problems in finance, genetic algorithms have previously been successfully applied (cf. Gilli and Schumann, 2012). Specifically, several genetic algorithms have been proposed for solving the index-tracking problem without the UCITS regulatory constraints (cf., e.g., Andriosopoulos and Nomikos, 2014; Beasley et al., 2003; Chiam et al., 2013; Ruiz-Torrubiano and Suárez, 2009). According to Gottlieb et al. (2001), the most important element in the design of such genetic algorithms is the representation, i.e. the mapping between the data structure of a solution, referred to as the genotype, and the decoded solution, referred to as the phenotype. To enable the design of an efficient and effective genetic algorithm, the representation should exhibit the following four properties (cf. Gottlieb et al., 2001): efficiency, meaning that a genotype can be rapidly decoded into its corresponding phenotype; locality, meaning that small changes in a genotype lead to small changes in the corresponding phenotype; heritability, meaning that combining parent genotypes using crossover operators produces child genotypes whose corresponding phenotypes exhibit combined features of the parent phenotypes; and a fourth property that is called feasibility hereafter. The feasibility property is satisfied if all feasible and no infeasible phenotypes are represented in the set of all possible genotypes. To the best of our knowledge, there is no representation of subsets in the literature that exhibits the feasibility property with respect to a constraint on the subset's minimum and maximum cardinality. Moreover, the existing genetic algorithms lack the ability to handle the regulatory constraints for UCITS.

The contributions of this paper are threefold. First, we present a new representation of subsets that exhibits the four desired properties of efficiency, locality, heritability, and feasibility. The feasibility property enables the use of fast and simple conventional evolutionary operators without requiring any time-consuming repair operators or penalty functions to handle infeasible phenotypes. The proposed representation of subsets should be of general interest because it can be used in genetic algorithms for solving any optimization problem that involves the selection of a subset, such as the feature-selection problem in machine learning. Second, we present a new MIQP formulation of the UCITP that requires fewer constraints than the existing formulation of Scozzari et al. (2013). Third, we present a new two-stage approach to the UCITP that is able to devise very good feasible solutions for UCITP instances of arbitrary size in a short CPU time. In the first stage of the proposed approach, we simplify the UCITP by considering only equally weighted portfolios, which results in a pure combinatorial asset-selection problem. To solve this asset-selection problem, we apply a genetic algorithm based on the proposed subset representation. The purpose of the first stage is to obtain a good feasible solution in a short CPU time. In the second stage, we improve the best solution obtained in the first stage by applying a local search method based on local branching that was introduced by Fischetti and Lodi (2003) and the new MIQP formulation of the UCITP. Compared with other local search methods, the advantage of local branching is that it is exact in nature, which allows provably optimal solutions to be determined starting from any feasible initial solution for small problem instances, but can also be applied heuristically, which allows very good feasible solutions to be determined for larger instances.

We report a computational experiment performed using 45 UCITP instances based on real-world stock-market data. The three main results of this experiment are as follows: 1) in comparison with two existing subset representations, using the new representation instead in the genetic algorithm leads to faster evolution and better results in terms of the objective function value; 2) when a pure MIQP approach is applied subject to a limit on the CPU time, the new MIQP formulation leads to better results in terms of the objective function value than the existing MIQP formulation; and 3) the two-stage approach leads to better results than the pure MIQP approach based on the new formulation within a limited CPU time. Further analysis demonstrates that the two-stage approach results in the lowest out-of-sample tracking error on average, and that the UCITS regulations reduce the portfolio risk in terms of the out-of-sample tracking error.

The remainder of this paper is organized as follows. In Section 2.2, we review the related literature. In Section 2.3, we present the proposed two-stage approach. In Section 2.4, we report the results of our computational experiment. In Section 2.5, we offer some concluding remarks and an outlook on future research.

2.2 Related literature

In this section, we review the related literature. In Subsection 2.2.1, we present an overview of existing approaches to the index-tracking problem without regulatory constraints. In Subsection 2.2.2, we present the MIQP formulation of the UCITP introduced by Scozzari et al. (2013). In Subsection 2.2.3, we discuss the existing subset representations used in genetic algorithms.

2.2.1 Index tracking

Sharma et al. (2017) categorize index-tracking approaches into two broad groups. Approaches in the first group use factor models to construct a portfolio (cf., e.g., Canakgoz and Beasley, 2009; Corielli and Marcellino, 2006; Rudd, 1980). Approaches in the second group minimize some measure of the tracking error, often subject to a cardinality constraint, i.e., a constraint on the number of assets that can be included in the tracking portfolio. Here, we focus on the second group of approaches, specifically on the different tracking-error measures that have been used. These measures can themselves be divided into two groups: value-based and return-based tracking-error measures (cf. Gaivoronski et al., 2005; Strub and Baumann, 2018).

Return-based tracking errors are calculated based on the returns of the portfolio and the index. Roll (1992) minimizes the tracking-error variance (TEV), i.e., the variance of the differences between the portfolio returns and the index returns. Kwiatkowski (1992) minimize the TEV subject to a cardinality constraint. Mutunge and Haugland (2018) show that the TEV minimization subject to a cardinality constraint is NP-hard, and present a greedy heuristic to tackle the problem. The TEV is commonly used in both practical and theoretical work (cf. Corielli and Marcellino, 2006). Nevertheless, Beasley et al. (2003) argue against the use of the TEV because with this measure, a portfolio can have a tracking error of zero even if its returns are constantly below those of the index. Because of this drawback of the TEV, the mean-squared error (MSE) of the return differences has often been used instead in the literature (cf., e.g., Andriosopoulos and Nomikos, 2014; Beasley et al., 2003; Chiam et al., 2013; Maringer and Oyewumi, 2007; Sant'Anna et al., 2017b; Takeda et al., 2013). According to Rudolf et al. (1999), the use of quadratic tracking-error measures such as the TEV and MSE is common in financial practice because they reveal a number of desirable statistical properties. However, Rudolf et al. (1999) argue that quadratic tracking-error measures are difficult for practitioners to interpret, and they propose four different tracking-error measures based on the absolute differences between the returns of the portfolio and the index. One advantage of these four measures is that they can be formulated as linear objective functions. Chen and Kwon (2012) maximize the correlation between the portfolio returns and the index returns, which is also formulated as a linear objective function.

Value-based tracking errors are calculated based on the value developments of the portfolio and the index. Konno and Wijayanayake (2001) and Guastaroba and Speranza (2012) use the mean absolute deviation (MAD) between the value developments of the portfolio and the index as a measure of the tracking error. The MAD can also be formulated as a linear objective function. Strub and Baumann (2018) propose a value-based tracking error that also exhibits properties of return-based tracking errors; specifically, the proposed tracking error is zero if and only if the historical returns of the tracking portfolio and the index coincide.

Following Krink et al. (2009), who introduced the UCITP, we chose to use the MSE in this paper because it is very commonly applied in practice and in the literature. However, the approach presented in this paper could be used without structural adjustments for any tracking-error measure that can be formulated as a linear or convex quadratic function.

2.2.2 Existing MIQP formulation of the UCITP

The MIQP formulation (M-STPK) of the UCITP presented by Scozzari et al. (2013) is given below. Table 2.1 defines the nomenclature used in this MIQP formulation.

$$
\text{(M-STPK)} \begin{cases}
\text{Min. } \dfrac{1}{|T|} \displaystyle\sum_{t \in T} \left(\sum_{i \in I} r_t^i w_i - r_t \right)^2 & & (2.1) \\
\text{s.t. } \displaystyle\sum_{i \in I} w_i = 1 & & (2.2) \\
l \leq \displaystyle\sum_{i \in I} y_i \leq k & & (2.3) \\
\varepsilon y_i \leq w_i \leq \delta y_i & (i \in I) & (2.4) \\
\displaystyle\sum_{i \in I} v_i \leq \eta & & (2.5) \\
\zeta u_i \leq w_i \leq \zeta + u_i & (i \in I) & (2.6) \\
w_i + u_i - 1 \leq v_i \leq w_i & (i \in I) & (2.7) \\
v_i \leq u_i & (i \in I) & (2.8) \\
w_i \geq 0,\ v_i \geq 0,\ y_i \in \{0,1\},\ u_i \in \{0,1\} & (i \in I) & (2.9)
\end{cases}
$$

The objective function given in (2.1) captures the MSE between the returns of the index-tracking portfolio ($\sum_{i \in I} r_t^i w_i$) and the corresponding index (r_t) over all historical periods $t \in T$. Constraint (2.2) is the budget constraint and ensures that the portfolio weights sum to one. The cardinality constraint (2.3) defines a feasible range between l and k for the number of assets to be included in the tracking portfolio. The cardinality constraint employs binary variables y_i, where y_i is equal to one if asset $i \in I$ is included in the tracking portfolio and zero otherwise. The constraints defined in (2.4) impose a lower bound (ε) and an upper bound that corresponds to the middle UCITS threshold (δ) on the weight of each asset included in the tracking portfolio and simultaneously ensures that the binary variables y_i are assigned the appropriate values. Constraint (2.5) limits the sum of the portfolio weights that exceed the lower UCITS threshold (ζ) to the upper UCITS threshold (η). In this constraint, continuous decision variables v_i are used, where v_i is equal to the weight of asset i if its weight exceeds the lower UCITS threshold and zero otherwise. To determine appropriate values of these continuous decision variables v_i, binary decision variables u_i are introduced, where u_i is equal to one if the weight of asset i exceeds the UCITS lower threshold and zero otherwise. Appropriate values are assigned to these binary decision variables based on the constraints defined in (2.6). Based on the values of the binary decision variables u_i, the constraints defined in (2.7) assign

Table 2.1: Nomenclature for the MIQP formulation.

Parameters and sets:	
n	Number of index constituents
I	Set of identity tags of the index constituents ($I = \{1, \dots, n\}$)
T	Set of historical in-sample time periods
l/k	Minimum/maximum portfolio cardinality
ε	Minimum weight of each asset if selected
$\zeta/\delta/\eta$	Lower/middle/upper UCITS concentration-rule thresholds
r_t/r_t^i	Return of index/asset $i \in I$ in period $t \in T$
Decision variables:	
w_i	Weight of asset $i \in I$ in the portfolio
y_i	$= 1$, if $w_i > 0$; $= 0$, otherwise ($i \in I$)
v_i	$= w_i$, if $w_i > \zeta$; $= 0$, otherwise ($i \in I$)
u_i	$= 1$, if $w_i > \zeta$; $= 0$, otherwise ($i \in I$)

appropriate values to the continuous decision variables v_i. The constraints given in (2.8) ensure that each variable v_i is set to zero if the weight of asset i does not exceed the lower UCITS threshold ($u_i = 0$). The domains of the decision variables are specified by (2.9).

2.2.3 Existing subset representations

In this subsection, we discuss the existing representations of subsets that have previously been used in genetic algorithms for problems that involve the selection of a subset of a set of identity tags $I = \{1, \dots, n\}$ subject to a feasible range for the subset's cardinality. Examples of such problems are the UCITP considered in this paper, the index-tracking problem without regulatory constraints, and the problem of selecting the best features for linear regression or machine learning (cf., e.g., Bertolazzi et al., 2016; Bertsimas and King, 2015; Bertsimas et al., 2016). The representations that have previously been used in genetic algorithms for solving these problems can be divided into two classes: pure subset representations and mixed representations. Phenotypes of the first class represent subsets only. By contrast, phenotypes of the second class also represent additional decisions related to the elements to be included in the subset, such as the portfolio weights.

These two classes of representations can each be further divided into two subclasses based on the genotypes used: pure subset representations comprise binary and integer representations, whereas mixed representations comprise real-valued and hybrid representations. In the following, we describe the four subclasses.

In binary representations, a vector $\{0, 1\}^n$ is used as a genotype (cf. Brill et al., 1992;

Kuncheva and Jain, 1999; Moral-Escudero et al., 2006; Oh et al., 2004; Ruiz-Torrubiano and Suárez, 2007; Siedlecki and Sklansky, 1989). The binary digits correspond to the decisions regarding whether each element is included in the subset. For example, if the i^{th} digit in the vector is equal to one, then the identity tag i is included in the subset.

In integer representations, the genotypes are based on integers that correspond to the identity tags of the selected elements. In Strub and Trautmann (2016), a vector of distinct integers from the set I is used as a genotype. Moral-Escudero et al. (2006) and Ruiz-Torrubiano and Suárez (2009, 2010) directly use subsets of the set I as genotypes.

In real-valued representations, a vector $\mathbb{R}^n$ is employed as a genotype (cf. Andriosopoulos and Nomikos, 2014; Diosan, 2005; Streichert et al., 2004). A corresponding phenotype is constructed by including in the subset all identity tags i such that the value of the i^{th} element in the real-valued vector is non-zero. If identity tag i is included in the subset, then the i^{th} value in the real-valued vector is used as the value of the associated continuous decision variable.

Hybrid representations are combinations of either binary or integer representations with real-valued representations. Chiam et al. (2013), Raymer et al. (2000), Skolpadungket et al. (2007), and Streichert et al. (2004) use a binary vector $\{0, 1\}^n$ and a real-valued vector $\mathbb{R}^n$ as a genotype. The value of the i^{th} element in the real-valued vector is multiplied by the value of the i^{th} element in the binary vector. If the resulting value is non-zero, then the identity tag i is included in the subset, and the resulting value is assigned to the associated continuous decision variable. Hence, the binary vector can be interpreted as a masking vector (cf. Raymer et al., 2000). Chiam et al. (2008) consider the mean-variance portfolio-optimization problem and use a permutation of the vector $[1, 2, \ldots, n]$ combined with a real-valued vector $\mathbb{R}^n$ as a genotype. The portfolio is constructed by selecting the assets with the identity tags defined by the order of the permuted vector. The assets are included in the subset, with weights assigned in accordance with the values in the real-valued vector, until the sum of the weights of the assets included in the portfolio exceeds one. Then, all weights are normalized such that their sum is equal to one.

Table 2.2 presents an illustrative example of how the discussed representations are used to decode a genotype into the corresponding phenotype. For all representations except the integer representation, the table shows a possible genotype that is decoded into a subset with an infeasible cardinality. Moreover, the integer representation and the second hybrid representation listed in the table require special evolutionary operators that maintain the properties of the genotypes, i.e., the uniqueness of the integers in each genotype. Hence, none of the discussed representations exhibits the feasibility property, which means that either simple and fast conventional evolutionary operators cannot be applied or penalty

Table 2.2: Illustrative example of the subset representations with $n = 5$, a feasible subset cardinality of three or four, and associated continuous variables that correspond to the portfolio weights w_i, $i = 1, \ldots, n$.

Representation	Possible genotype	Decoded phenotype	Feasible
Binary	$[0, 1, 0, 0, 1]$	$\{2, 5\}$	✗
Integer	$[2, 3, 5]$	$\{2, 3, 5\}$	✓
Real	$[0, 0.75, 0, 0, 0.25]$	$\{2, 5\}$, $w_2 = 0.75$, $w_5 = 0.25$	✗
Hybrid with binary	$[0, 1, 0, 0, 1]$, $[0.5, 0.75, 0.8, 0.6, 0.25]$	$\{2, 5\}$, $w_2 = 0.75$, $w_5 = 0.25$	✗
Hybrid with integer	$[2, 5, 4, 3, 1]$, $[0.35, 0.9, 0.8, 0.25, 0.3]$	$\{2, 5\}$, $w_2 = 0.75$, $w_5 = 0.25$	✗

functions or repair operators must be applied to handle infeasible phenotypes.

2.3 Solution approach

In Strub and Trautmann (2017), we presented a preliminary version of the solution approach proposed in this paper. In this preliminary version, we used a hybrid genetic algorithm similar to that of Moral-Escudero et al. (2006), in which the fitness of each individual is determined by applying an exact solution method such as mixed-integer programming. To reduce the CPU time for the fitness evaluations, we first estimated the fitness of the individuals in an efficient way and then evaluated the fitness of promising individuals only. In the present paper, we propose a new way of combining a genetic algorithm with another solution method in a sequential manner. Specifically, we present a two-stage approach in which a genetic algorithm is used in the first stage to determine a good feasible equally weighted portfolio and an MIQP-based local-branching method is used in the second stage to improve the solution from the first stage. The new two-stage approach produces superior results compared with the approach presented in Strub and Trautmann (2017) and even allows provably optimal solutions to be determined for small-scale instances.

In Subsection 2.3.1, we present the new MIQP formulation that we use in the local-branching method. In Subsection 2.3.2, we present the new subset representation that we use in the genetic algorithm. In Subsection 2.3.3, we present the two-stage approach in detail. Table 2.1 defines the notation used.

2.3.1 New MIQP formulation of the UCITP

For the new MIQP formulation of the UCITP, we replace the continuous and binary decision variables v_i and u_i, respectively, that are used in Scozzari et al. (2013) with the

Table 2.1: Nomenclature for the two-stage approach.

s	Size of population (number of individuals)
P	Population (set of individuals)
M	Mating pool (set of individuals)
$\underline{d}/\overline{d}$	Minimum/maximum dimension of genotype vectors, with $\underline{d} > 0$ and $\overline{d} \leq n$
$\mathbf{g}^i \in \{1, \ldots, n\}^{d_i}$	Genotype vector $\left[g_1^i, \ldots, g_{d_i}^i\right]$ of individual i with $d_i \in \{\underline{d}, \ldots, \overline{d}\}$
$f(\mathbf{g}^i)$	Fitness of individual i
$\mathbf{b}$	Genotype vector of the individual with the best known fitness
$random$	Uniformly distributed random number from the half-closed interval [0,1)
p_c	Probability of crossover
$p_e/p_a/p_r$	Probability of an exchange/addition/removal of an element in/to/from a genotype vector during mutation
n_g	Maximum number of generations for the genetic algorithm
n_s	Maximum number of stocks that are considered during local branching ($k \leq n_s \leq n$)

following two kinds of decision variables x_i and z_i:

$$x_i \geq w_i - \zeta, \text{ if } w_i > \zeta; \; \geq 0, \text{ otherwise} \qquad (i \in I)$$

$$z_i = 1, \text{ if } w_i > \zeta; \; \geq 0, \text{ otherwise} \qquad (i \in I)$$

The new MIQP formulation of the UCITP reads as follows:

$$\text{(M-ST)} \begin{cases} \text{Min.} & (2.1) \\ \text{s.t.} & (2.2)\text{–}(2.4) \\ & w_i - \zeta \leq x_i \qquad (i \in I) \qquad (2.10) \\ & x_i \leq (\delta - \zeta) z_i \qquad (i \in I) \qquad (2.11) \\ & \sum_{i \in I} (x_i + \zeta z_i) \leq \eta \qquad (2.12) \\ & w_i \geq 0, \; x_i \geq 0, \; y_i \in \{0,1\}, \; z_i \in \{0,1\} \qquad (i \in I) \qquad (2.13) \end{cases}$$

We use the objective function defined in (2.1) and adopt constraints (2.2)–(2.4) from the formulation (M-STPK) of Scozzari et al. (2013) to model the budget constraint, the cardinality constraint, and the lower and upper bounds on the portfolio weights of the selected assets. The remaining constraints defined in (2.10)–(2.13) differ from those used in (M-STPK). In the following, we describe these new constraints.

The constraints defined in (2.10) determine the values of the non-negative continuous variables x_i: if the weight of asset i does not exceed the lower UCITS threshold, then the variable x_i can take a value of zero; otherwise, x_i must be assigned a value that is at least equal to the difference between the weight of asset i and the lower UCITS threshold. The constraints defined in (2.11) determine the values of the variables z_i: z_i must be equal to one if the weight of asset i exceeds the lower UCITS threshold and can be equal to either zero or one otherwise. In addition, the constraints given in (2.11) set upper bounds on the variables x_i because the weight of any asset cannot exceed the lower UCITS threshold by more than $\delta - \zeta$. The constraints given in (2.12) ensure that the sum of the weights exceeding the lower UCITS threshold does not exceed the upper UCITS threshold. It is important to note that the left-hand side corresponds only to an upper bound on the sum of the weights of the assets whose weights exceed the lower UCITS threshold. However, this upper bound is sufficient to guarantee that the 5/10/40 UCITS concentration rule is satisfied.

The proposed MIQP formulation (M-ST) requires fewer constraints than the existing formulation (M-STPK). The proposed formulation requires only $4n + 4$ constraints (ignoring those that define the domains of the decision variables), whereas the existing formulation contains $7n + 4$ constraints.

2.3.2 New subset representation

A representation has three components (cf. Gottlieb et al., 2001): the phenotypes, the genotypes, and the decoding procedure that maps the genotypes to the phenotypes.

In the proposed representation of subsets, the phenotypes correspond to subsets of the set of identity tags $I = \{1, \ldots, n\}$. As genotype, we use a d-dimensional vector of integers $\mathbf{g} \in \{1, \ldots, n\}^d$ with d between $\underline{d}$ and $\overline{d}$. Here, we assume that the values of the parameters $\underline{d}$ and $\overline{d}$ can be chosen such that the resulting phenotypes are always feasible. The question of how to choose these values for the UCITP is discussed in Section 2.4.1. The decoding procedure (cf. Algorithm 2.1) maps a genotype vector to a phenotype S as follows. Each element g_i of the genotype vector is included in the phenotype S if $g_i \notin S$. If $g_i \in S$, then g_i is modified until $g_i \notin S$, and this modified integer g_i is inserted into S. Hence, all phenotypes S correspond to subsets of the set I with a cardinality equal to the dimension d of the corresponding genotype vector. As an example with $n = 6$ and $d = 4$, the vector $[2, 2, 6, 6]$ denotes a possible genotype, which is decoded into the phenotype $\{2, 3, 6, 1\}$.

In the worst case, i.e., if the genotype contains $\overline{d}$ integers that are all identical, then Algorithm 2.1 requires $\frac{(\overline{d}-1)\overline{d}}{2}$ modifications of the genotype's elements and $\overline{d}$ insertions of the genotype's modified elements into the phenotype S. In the best case, i.e., if all elements in the genotype vector are distinct, no element needs to be modified, and Algorithm 2.1

Algorithm 2.1 $\mathcal{O}(\overline{d}^2)$ Decoding

procedure DEC($\mathbf{g} \in \{1, \ldots, n\}^d$)
$S := \emptyset$
for $i := 1$ to d **do**
while $g_i \in S$ **do**
$g_i := (g_i + 1) \bmod n$
end while
$S := S \cup \{g_i\}$
end for
return S
end procedure

performs only $\overline{d}$ insertions of the genotype's elements into the phenotype. Hence, the best-case and worst-case time complexities of the decoding procedure are $\mathcal{O}(\overline{d})$ and $\mathcal{O}(\overline{d}^2)$, respectively.

Based on a sorting algorithm that sorts the genotype vector in $\mathcal{O}(\overline{d}\log\overline{d})$ iterations in the worst case (cf., e.g., Quicksort and Mergesort described in Cormen et al., 2001), we design a new decoding procedure (cf. Algorithm 2.2) that has a better worst-case time complexity than the $\mathcal{O}(\overline{d}^2)$ decoding procedure. The new procedure works as follows. First, the integers in the genotype vector are sorted in a non-decreasing order. Then, all duplicate integers in the genotype are increased by applying the REMOVEDUPLICATES procedure (cf. Algorithm 2.3) such that there are no more duplicate integers in the genotype vector. Since the for-loop is executed $\overline{d} - 1$ times, the worst-case time complexity of the REMOVEDUPLICATES procedure is $\mathcal{O}(\overline{d})$. The sorted and modified genotype vector is then adjusted such that no integer is larger than n. This is done using the ADJUST procedure (cf. Algorithm 2.4). The while-loop in the ADJUST procedure is executed $\overline{d}$ times, and thus, the procedure has a worst-case time complexity of $\mathcal{O}(\overline{d})$. In total, the worst-case time complexity of Algorithm 2.2 is therefore $\mathcal{O}(\overline{d}\log\overline{d})$.

Algorithm 2.2 $\mathcal{O}(\overline{d}\log\overline{d})$ Decoding

procedure FASTDEC($\mathbf{g} \in \{1, \ldots, n\}^d$)
Sort $\mathbf{g}$ in non-decreasing order
$\mathbf{g} :=$ REMOVEDUPLICATES($\mathbf{g}$)
$S :=$ ADJUST($\mathbf{g}$)
return S
end procedure

In the following, we illustrate the $\mathcal{O}(\overline{d}\log\overline{d})$ decoding procedure by means of a small illustrative example. Suppose for this example that $n = 10$ and that the genotype vector is

Algorithm 2.3 $\mathcal{O}(\overline{d}\log\overline{d})$ Decoding – RemoveDuplicates procedure

```
procedure RemoveDuplicates(g ∈ {1, ..., n}^d)
    for i := 2 to d do
        if g_{i−1} ≥ g_i then
            g_i := g_{i−1} + 1
        end if
    end for
    return g
end procedure
```

Algorithm 2.4 $\mathcal{O}(\overline{d}\log\overline{d})$ Decoding – Adjust procedure

```
procedure Adjust(g ∈ {1, ..., n}^d)
    S := ∅; i := 1; m := 1; j := d
    while j ≥ i do
        if g_j > n then
            if g_i = m then
                S := S ∪ {g_i}; i := i + 1; m := m + 1
            else
                S := S ∪ {m}; j := j − 1; m := m + 1
            end if
        else
            S := S ∪ {g_j}; j := j − 1
        end if
    end while
    return S
end procedure
```

$[8, 8, 1, 10, 10, 9]$. Then, the decoding procedure is as follows. First, the genotype vector is sorted in a non-decreasing order, which leads to the genotype vector $[1, 8, 8, 9, 10, 10]$. This sorted vector is then modified using the RemoveDuplicates procedure such that it does not contain any duplicate integers. The result is the genotype vector $[1, 8, 9, 10, 11, 12]$. Finally, using the Adjust procedure, the genotype vector is adjusted such that it does not contain any integers larger than n. The resulting decoded phenotype is $\{1, 2, 3, 10, 9, 8\}$.

Two special features of the proposed representation are that the number of all possible genotypes exceeds the number of all possible phenotypes and that not all phenotypes are represented by the same number of genotypes. Hence, the representation exhibits a biased redundancy (cf. Rothlauf, 2011). Phenotypes with consecutive identity tags are represented by the most genotypes because the decoding procedure inserts consecutive identity tags into the phenotypes in place of duplicate integers in the genotypes. This knowledge can be exploited in a very simple way. For example, in mean-variance portfolio-optimization problems, assets with low correlation are likely to be included in an optimal solution. Hence, pairs of weakly correlated assets could be assigned consecutive identity tags.

The properties of feasibility, efficiency, locality, and heritability are investigated in Subsection 2.4.3.1.

2.3.3 Two-stage approach

The two-stage approach proceeds as follows. In the first stage, a genetic algorithm is applied based on the proposed subset representation to obtain a good feasible solution. Then, the local-branching method presented by Fischetti and Lodi (2003) is applied based on the new MIQP formulation (M-ST) of the UCITP to improve the solution found in the first stage. In the following, we describe the two stages.

2.3.3.1 Stage one: genetic algorithm

The genetic algorithm (cf. Algorithm 2.7 in the appendix) is designed similarly to the simple genetic algorithm (SGA) described by Rothlauf (2011). First, an initial population is generated at random. Then, the evolutionary process begins and is repeated until a given number of generations n_g is reached. During the evolutionary process, a process of binary tournament selection with replacement (cf. Rothlauf, 2011) is applied to determine the mating pool M. The individuals in the mating pool are then either combined with probability p_c using a crossover operator or left unchanged. The resulting individuals are inserted into the population P'. Finally, a mutation operator is applied to the individuals in P', and the old population P is replaced with the new population P'.

In the genetic algorithm, the fitness of each individual is calculated as follows. The genotype $\mathbf{g}$ is decoded, and the resulting set of assets is used to define the assets to be included in the tracking portfolio, each with an equal weight of $\frac{1}{|\textsc{Dec}(\mathbf{g})|}$. Hence, the fitness is calculated using the following function:

$$f(\mathbf{g}) = \frac{1}{|T|} \sum_{t \in T} \left(\sum_{i \in \textsc{Dec}(\mathbf{g})} r_t^i \frac{1}{|\textsc{Dec}(\mathbf{g})|} - r_t \right)^2 \tag{2.14}$$

In the following, we briefly describe the crossover and mutation operators (cf. Algorithms 2.6 and 2.8 in the appendix) that are very similar to standard operators from the literature.

In the mutation operator, a randomly chosen element of the genotype vector is exchanged with a randomly chosen integer from the set $\{1, \ldots, n\}$ with probability p_e. In addition to this standard mutation operator, we also allow a randomly chosen element to be added to or removed from the genotype vector with probability p_a or p_r, respectively. Because the dimension of the genotype vector can change during mutation, we must check whether the dimension of the mutated genotype vector is feasible, i.e., whether it is in the range $\left[\underline{d}, \overline{d}\right]$. If the mutated genotype vector has a feasible dimension, it is returned; otherwise, the genotype before the mutation is returned.

The crossover operator is very similar to a conventional m-point crossover operator. First, the dimensions of the child genotype vectors $\mathbf{g}^3$ and $\mathbf{g}^4$ are set equal to the dimensions of the parent genotype vectors $\mathbf{g}^1$ and $\mathbf{g}^2$, respectively. Then, the crossover point m is randomly chosen. The elements on the left side of m from parent genotype vectors $\mathbf{g}^1$ and $\mathbf{g}^2$ are assigned to the child genotype vectors $\mathbf{g}^4$ and $\mathbf{g}^3$, respectively. Furthermore, the elements on the right side of m (including m) from parent genotype vectors $\mathbf{g}^1$ and $\mathbf{g}^2$ are assigned to the child genotype vectors $\mathbf{g}^3$ and $\mathbf{g}^4$, respectively. With two parent genotype vectors $\mathbf{g}^1$ and $\mathbf{g}^2$ that have the same dimension, the operator is identical to the m-point crossover operator. As in the case of the mutation operator, if the dimensions of the input genotype vectors are feasible, then the returned genotype vectors will also have feasible dimensions.

2.3.3.2 Stage two: local branching

To improve the solution obtained in the first stage, we apply the local-branching method described in Algorithm 2.5. The algorithm takes as input the genotype vector $\mathbf{b}$ that represents the best individual from stage 1. Based on this individual, the MIQP formulation

(M-ST-A) is solved to determine an optimal portfolio, i.e., the optimal portfolio weights for the assets selected in the solution from stage 1. Then, the set J, which represents the assets selected in the current solution, is initialized. The parameters a and b are also initialized. These parameters are used in the local-branching constraint that is explained below. Then, local branching starts; it is conducted either exactly or heuristically depending on the value used for the parameter n_s. In the following, we describe the exact and heuristic behaviors of the local-branching method.

If n_s is not smaller than the number of index constituents n, then the method operates exactly. In this case, the original set I is used as the set of assets considered during local branching. Then, the MIQP formulation (M-ST-B) is solved. We do not impose a separate time limit for this MIQP in addition to the overall time limit for the two-stage approach. The MIQP formulation (M-ST-B) corresponds to the formulation (M-ST) with the additional local-branching constraint given in (2.22). This local-branching constraint ensures that at least a and at most b of the binary variables y_i change in value with respect to a previous solution. For this purpose, the first sum in the local-branching constraint counts the number of binary variables y_i, $i \in J$, that had a value of one in the previous solution and a value of zero in the current solution. The second sum counts the number of binary variables y_i, $i \in I \setminus J$, that had a value of zero in the previous solution and a value of one in the current solution. If a better solution is found, then the parameters a and b are reset to one and two, respectively. Otherwise, the parameter a is increased to $b + 1$, and b is increased to the new value of a plus 1. Since we do not impose a time limit for solving the MIQP formulation (M-ST-B), we need not consider solutions with a smaller a and b because we know that there is no better solution for a smaller a and b. Hence, if $n_s \geq n$ and the overall time limit for the two-stage approach is sufficiently long, then the local-branching method will eventually find a solution with proven optimality.

If we choose $n_s < n$, then the method proceeds heuristically. In this case, only a subset of the set of all index constituents is considered in each iteration of the local-branching method. Specifically, I is set to J, and some randomly selected elements from the set $\{1, \ldots, n\}$ are added to I such that the cardinality of I is equal to n_s. Furthermore, we do not adjust a, and we adjust b only if no better solution could be found by solving the MIQP formulation (M-ST-B). The reason for this is that with $n_s < n$, a better solution with $a = 1$ could exist.

The procedure is repeated until a specified time limit is reached or a provably optimal solution is found.

$$
\text{(M-ST-A)}\begin{cases}
\text{Min. } \dfrac{1}{|T|}\displaystyle\sum_{t\in T}\left(\sum_{i\in\textsc{Dec}(\mathbf{b})} r_t^i w_i - r_t\right)^2 & & (2.15)\\
\text{s.t. } \displaystyle\sum_{i\in\textsc{Dec}(\mathbf{b})} w_i = 1 & & (2.16)\\
\varepsilon \le w_i \le \delta & (i \in \textsc{Dec}(\mathbf{b})) & (2.17)\\
w_i - \zeta \le x_i & (i \in \textsc{Dec}(\mathbf{b})) & (2.18)\\
x_i \le (\delta - \zeta) z_i & (i \in \textsc{Dec}(\mathbf{b})) & (2.19)\\
\displaystyle\sum_{i\in\textsc{Dec}(\mathbf{b})} (x_i + \zeta z_i) \le \eta & & (2.20)\\
w_i \ge 0,\ x_i \ge 0,\ z_i \in \{0,1\} & (i \in \textsc{Dec}(\mathbf{b})) & (2.21)
\end{cases}
$$

$$
\text{(M-ST-B)}\begin{cases}
\text{Min. } (2.1) & \\
\text{s.t. } (2.2)\text{–}(2.4), (2.10)\text{–}(2.13) & \\
a \le \displaystyle\sum_{i\in J}(1 - y_i) + \sum_{i\in I\setminus J} y_i \le b & (2.22)
\end{cases}
$$

Algorithm 2.5 Local Branching – Stage 2

procedure $\textsc{LocalBranching}(\mathbf{b})$
    $J := \textsc{Dec}(\mathbf{b})$; $a := 1$; $b := 2$; Solve (M-ST-A)
    while time limit not reached **do**
        if $n_s \ge n$ **then**
            $I := \{1, \dots, n\}$
        else
            $I := J$; Add random elements from $\{1, \dots, n\}$ to I until $|I| = n_s$
        end if
        Solve (M-ST-B)
        if better solution found **then**
            $a := 1$;$b := 2$
            $J := \{i \in I :$ asset i is selected in the solution to (M-ST-B)$\}$
        else
            if $n_s \ge n$ **then** $a := b + 1$; $b := a + 1$ **else** $b := b + 1$ **end if**
            if $a > n$ **then return** optimal solution **end if**
        end if
    end while
end procedure

2.4 Numerical experiment

In this section, we report the results of our computational experiment. In this experiment, we investigated the performance of different solution approaches to the UCITP in terms of the best objective function value and the running time. Specifically, we investigated and compared the performance of the following approaches:

- The genetic algorithm based on the new subset representation (with the $\mathcal{O}(\overline{d}^2)$ and $\mathcal{O}(\overline{d}\log\overline{d})$ decoding procedures with Quicksort as the sorting algorithm) was compared with a binary and an integer subset representation from the literature. For the binary representation, we used the implementation from the genetic algorithm utility library (GAUL; cf. Adcock, 2017) with the so-called death penalty for handling infeasible solutions (cf. Moral-Escudero et al., 2006), a bit-exchange operator as the mutation operator, and the m-point crossover operator. For the integer representation, we used the direct subset representation with the Random Respectful Recombination (R3) crossover operator and a bit-exchange mutation operator as described in Moral-Escudero et al. (2006).
- A pure MIQP approach based on the formulation (M-ST) was compared with a similar approach based on the formulation (M-STPK).
- The proposed two-stage approach was compared with the pure MIQP approach based on the formulation (M-ST) and with the genetic algorithm based on the proposed subset representation; here, we also evaluated the portfolios in an out-of-sample period.

In Subsection 2.4.1, we explain the test settings used in the experiment. In Subsection 2.4.2, we present the test instances. In Subsection 2.4.3, we report the results.

2.4.1 Test design

For the experiments, we used the parameter values given in Table 2.1. These parameters were partially dictated by the problem instances. Some of them, however, could be chosen and would affect the performance of the approaches. For these parameters, we used standard values. By using a parameter-tuning approach such as that presented in López-Ibánez et al. (2011), the results of the approaches might be improved. To ensure that the cardinality of the phenotypes in the genetic algorithm permitted the construction of a feasible solution, i.e., an equally weighted portfolio satisfying all constraints of the UCITP, we must ensure that $\overline{d} = k$ and $\underline{d} = 20$. Note that we assume here that $\frac{1}{k} \geq \varepsilon$.

Table 2.1: Values of parameters and sets used in the experiment.

Parameters/sets	Values
T	$\{1, \ldots, 104\}$
$n/k/r_t/r_t^i$	different depending on the test instance (cf. Subsection 2.4.2)
$\varepsilon/\zeta/\delta/\eta$	0.01/0.05/0.1/0.4
$l/s/\underline{d}/\overline{d}/n_g/n_s$	16/10n/20/k/500/100
$p_c/p_e/p_a/p_r$	0.5/0.25/0.25/0.25

To see why $\underline{d} = 20$, note that because of the UCITS concentration rule, all weights in an equally weighted portfolio must not exceed a value of 5%, because otherwise the portfolio would be infeasible. Hence, a selection of at least 20 and at most k assets enables the construction of a feasible equally weighted portfolio. Note that, however, when we consider portfolios that are not equally weighted, it is also possible to construct a portfolio that satisfies the UCITS concentration rule with 16 assets, i.e., with weights of 10% assigned to four stocks and weights of 5% assigned to twelve stocks.

For the comparison of the subset representations, slightly different parameter values from those in Table 2.1 were chosen. We set $l = k = \underline{d} = \overline{d} = 20$ because the subset representation of Moral-Escudero et al. (2006) is applicable only to a fixed portfolio cardinality. With a fixed portfolio cardinality, no addition or removal of elements to or from the genotype is possible during mutation. Therefore, we set the probabilities p_r and p_a to zero.

All approaches were implemented in C, and Gurobi 7.0 was used as the solver for the MIQP problems. All calculations were performed on an HP Z820 workstation with two 3.1 GHz Intel Xeon CPUs and 128 GB of RAM. As the running-time limit, we used two different values for each approach: 120 seconds and 1000 seconds per instance. All approaches were run five times with different random seeds. For the comparison of the subset representations, we imposed no running-time limit and ran the genetic algorithm for 500 generations.

2.4.2 Test instances

We considered 45 problem instances, all derived from real-world data. Each instance comprises the closing prices of n stocks and the index values for 156 periods. The first 104 periods were used as the in-sample data for the optimization, and the following 52 periods were used for out-of-sample evaluations of the portfolios. For the parameter k, we used the values 20 and 40 except for the Swiss Market Index (SMI) instance and the

Table 2.2: Problem instances.

Instance no.	Index	n	k	Time horizon
1/24	Hang Seng	31	20/31	1992–1995
2/25	DAX100	85	20/40	1992–1995
3/26	FTSE100	89	20/40	1992–1995
4/27	S&P100	98	20/40	1992–1995
5/28	Nikkei225	225	20/40	1992–1995
6/29	S&P500	457	20/40	NA
7/30	Russell2000	1,319	20/40	NA
8/31	Russell3000	2,152	20/40	NA
9	SMI	20	20	2012–2015
10/32	Hang Seng	49	20/40	2012–2015
11/33	EUROSTOXX50	50	20/40	2012–2015
12/34	FTSE100	96	20/40	2012–2015
13/35	S&P100	99	20/40	2012–2015
14/36	NASDAQ100	101	20/40	2012–2015
15/37	DAX100	102	20/40	2012–2015
16/38	SPI	198	20/40	2012–2015
17/39	Nikkei225	220	20/40	2012–2015
18/40	S&PASX300	254	20/40	2012–2015
19/41	S&P500	489	20/40	2012–2015
20/42	FTSE All Share	567	20/40	2012–2015
21/43	STOXXEURO600	575	20/40	2012–2015
22/44	S&P1200	1,179	20/40	2012–2015
23/45	NASDAQ Composite	2,140	20/40	2012–2015

Hang Seng Index instance, which include only 20 and 31 stocks, respectively.

Instances 1–8 and 24–31 were introduced by Beasley et al. (2003) and Canakgoz and Beasley (2009) and can be downloaded from the OR-Library (cf. Beasley, 1990). Instances 9–23 and 32–45 were introduced in Strub and Baumann (2018). For each instance, Table 2.2 lists the name of the index, the number of stocks n, the value(s) of the parameter k, and on which time horizon the data were collected (if known).

2.4.3 Results

In this subsection, we first report the results of the comparison of the subset representations. Then, we report the results of the comparison of the MIQP formulations. Then, we report the results of the comparison of the two-stage approach with the pure MIQP approach and the pure genetic algorithm. Finally, we investigate the impact of the UCITS concentration rule on the objective function values and the out-of-sample root-mean-squared errors.

2.4.3.1 Subset representations

The results in this subsection are reported as averages over all runs and over problem instances 1–23 with the values $l = k = \underline{d} = \overline{d} = 20$, as mentioned above.

First, we investigated the feasibility property of the different representations. The results are summarized in Table 2.3. The binary representation does not exhibit the feasibility property, as seen from the fact that approximately one quarter of all genotypes encountered during the 500 generations represented an infeasible phenotype. For the other representations, all genotypes represented feasible phenotypes. However, this does not guarantee that the integer representation exhibits the feasibility property, and in fact, it does not because special mutation and crossover operators must be applied and only subsets with a fixed cardinality can be represented. For the proposed representation, however, the feasibility property is satisfied.

Table 2.3: Feasibility: frequencies in [%] of genotypes representing a feasible or an infeasible phenotype.

	Binary	Integer	New $\mathcal{O}(\overline{d}^2)$	New $\mathcal{O}(\overline{d}\log\overline{d})$
Feasible	76.97	100.00	100.00	100.00
Infeasible	23.03	0.00	0.00	0.00

Next, we investigated the representations' locality. The locality depends on the mutation operator and on the metric used to measure the distance between phenotypes. We use the mutation operator presented in Subsection 2.3.3.1. For the distance $d^P(S_1, S_2)$ between two phenotypes S_1 and S_2, we used the following metric:

$$d^P(S_1, S_2) = \frac{|S_1 \setminus S_2| + |S_2 \setminus S_1| + \big|\, |S_1| - |S_2| \,\big|}{2} \tag{2.23}$$

This distance metric counts the number of exchanges, removals, or additions necessary to transform one phenotype into the other and can thus be regarded as an edit distance. As an example, for the phenotypes $S_1 = \{1, 2, 3\}$ and $S_2 = \{1, 2, 4\}$, we obtain a distance of $d^P(S_1, S_2) = 1$. Table 2.4 shows the numbers of mutations for which the distance between the phenotypes before and after mutation was zero, one, and larger than one. From this table, we can gain the following insights:

- Since there is no case with a distance larger than one, all representations exhibit the locality property.

- The integer and binary representations lead to a smaller fraction of cases with a distance of one. The reason is that the binary and integer representations use a bit-exchange mutation operator that is likely to select two bits with the same value, resulting in a distance of zero.

Table 2.4: Locality: frequencies in [%] of certain distances $d^P(S, S')$ between the phenotypes before mutation (S) and after mutation (S').

$d^P(S, S')$	Binary	Integer	New $\mathcal{O}(\overline{d}^2)$	New $\mathcal{O}(\overline{d}\log\overline{d})$
0	95.14	93.01	75.12	75.12
1	4.86	6.99	24.88	24.88
> 1	0.00	0.00	0.00	0.00

Next, we investigated the heritability property. The heritability property is satisfied if after a crossover, the distance between the mother and father phenotypes is no smaller than any of the distances between parent and child phenotypes, i.e., between the phenotypes of mother and daughter, mother and son, father and daughter, and father and son. For each representation, Table 2.5 lists the frequency of crossovers in which all parent-child distances were no larger than the distance between the parents and the frequency of crossovers in which any parent-child distance was larger than the distance between the parents. From this table, we can conclude that the binary and integer representations perfectly exhibit the heritability property. The proposed representation also exhibits the heritability property, as can be seen from the very small number of cases in which any child-parent distance was larger than the distance between the parents.

Table 2.5: Heritability: distance between parent and child phenotypes vs. distance between parent phenotypes. A difference smaller than or equal to zero means that the distance between the parent phenotypes was not smaller than each of the four distances between one of the two parents and one of the two children; a positive difference means that at least one of the four distances between one of the two parents and one of the two children was larger than the distance between the parents. Frequencies are expressed in [%].

	Binary	Integer	New $\mathcal{O}(\overline{d}^2)$	New $\mathcal{O}(\overline{d}\log\overline{d})$
≤ 0	100.00	100.00	99.89	99.89
> 0	0.00	0.00	0.11	0.11

Next, we investigated the efficiency property. From Subsection 2.3.2, we know that the worst-case time complexity of the decoding procedure for the proposed subset representation is either $\mathcal{O}(\overline{d}^2)$ or $\mathcal{O}(\overline{d}\log\overline{d})$, depending on the algorithm used. These complexities can be regarded as efficient. To see how the efficiency of the proposed representation compares with that of other subset representations, we list in Table 2.6 the running times necessary to complete 500 generations of the genetic algorithm based on the different representations. These running times encompass not only the decoding process but also the mutation and crossover operators. From this table, we can see that the proposed representation is the fastest on average. Moreover, the $\mathcal{O}(\overline{d}^2)$ decoding procedure is slightly faster than the $\mathcal{O}(\overline{d}\log\overline{d})$ decoding procedure, which can be attributed to the best-case time complexity of $\mathcal{O}(\overline{d})$ for the $\mathcal{O}(\overline{d}^2)$ decoding procedure that is achieved when there are no duplicate integers in the genotype vectors. This is often the case because $\overline{d}$ is much smaller than n for most of the instances, which reduces the probability of duplicate integers. Furthermore, Table 2.6 shows the averages of the best objective function values after 500 generations. The proposed representation also yields the best results in terms of the objective function value. The superiority of these results compared with those of the binary representation can, at least in part, be attributed to the fact that only feasible phenotypes are investigated when using the proposed representation. Meanwhile, a possible explanation for the superior results compared with those of the integer representation may be that the higher frequency of actual mutations performed with the proposed representation (cf. Table 2.4) was beneficial.

Table 2.6: Objective function value (OFV) of the best solution after 500 generations and the corresponding running time necessary.

	Binary	Integer	New $\mathcal{O}(\overline{d}^2)$	New $\mathcal{O}(\overline{d}\log\overline{d})$
Time [s]	52.6	33.8	22.5	23.1
OFV	145.3	92.1	88.4	88.4

From this first set of experiments, we can conclude that the genetic algorithm based on the novel subset representation yields the best results in terms of running time and objective function value. In addition, the proposed subset representation exhibits the properties of feasibility, locality, heritability, and efficiency.

2.4.3.2 MIQP formulations

Next, we compared the two formulations (M-ST) and (M-STPK). For all instances, Table 2.7 lists the best objective function value (column UB) and the best lower bound on the objective function value (column LB) obtained by the Gurobi solver after 120 seconds based on the two formulations. Bold values indicate the better formulation for each instance based on the objective function value. As seen from this table, both formulations lead to similar lower bounds. However, the new formulation yields better objective function values on average. We also performed two non-parametric statistical tests, specifically two Wilcoxon signed rank tests, to compare the results. These two tests indicated that the median lower bounds are not statistically different at a standard significance level (p-value: 0.4531) but that the median of the best objective function values obtained when using (M-ST) is significantly lower than that obtained with (M-STPK) (p-value: 0.0028).

2.4.3.3 Two-stage approach

In columns four to nine of Table 2.8, we compare the best objective function values obtained using the two-stage approach with the best objective function values obtained using the pure MIQP approach based on the formulation (M-ST) and using the genetic algorithm based on the proposed subset representation with the $\mathcal{O}(\overline{d}^2)$ decoding procedure and with no limit on the number of generations. The results indicate that the proposed two-stage approach yields the best results on average for both a short running-time limit of 120 seconds and a longer running-time limit of 1000 seconds.

We also evaluated the portfolios constructed using the three approaches over an out-of-sample period of 52 weeks. Specifically, we calculated the root-mean-squared error (RMSE) between the returns of the tracking portfolio and the index for the entire out-of-sample period. The results are shown in columns 10 to 15 of Table 2.8. These results indicate that the portfolios' in-sample performances (objective function values) are consistent with the out-of-sample performances (RMSEs) in terms of the ranking among the three approaches; i.e., the two-stage approach also leads to the best out-of-sample results. However, the longer running-time limit had no marked influence on the out-of-sample results despite improving the in-sample results.

We also analyzed the differences in the objective function value and the RMSE using the non-parametric statistical tests implemented in the software package MULTIPLETEST (cf. `http://sci2s.ugr.es/sicidm` and Garcia and Herrera, 2008). Tables 2.9 and 2.10 report the main results obtained. Table 2.9 reports the Friedman ranks; a lower rank indicates better performance. According to these ranks, the proposed two-stage approach performed best in terms of both the objective function value and the RMSE. Table 2.10 reports the p-values obtained using different statistical procedures with respect to the null hypothesis that the performance does not differ between the two approaches represented in each row. For the in-sample period, the two-stage approach performed significantly better than both other approaches according to almost all tests. In addition,

Table 2.7: Best objective function values (UB) and best lower bounds on the objective function value (LB) after 120 seconds expressed as averages over all runs with different random seeds (scaled by 10^6). M-STPK: MIQP approach based on the formulation (M-STPK) of Scozzari et al. (2013); M-ST: MIQP approach based on the new formulation (M-ST); bold values indicate the best objective function value for each instance; * denotes that the solution's optimality can be proven.

Instance			M-STPK		M-ST	
n	k	No.	UB	LB	UB	LB
20	20	9	**90.9***	90.9	**90.9***	90.9
31	20	1	**56.1***	56.1	**56.1***	56.1
31	31	24	**47.3***	47.3	**47.3***	47.3
49	20	10	**27.8***	27.8	**27.8***	27.8
49	40	32	**13.3***	13.3	**13.3***	13.3
50	20	11	24.0	14.1	**23.6**	14.2
50	40	33	**5.9***	5.9	**5.9***	5.9
85	20	2	**24.6**	6.2	**24.6**	6.4
85	40	25	**3.7***	3.7	**3.7***	3.7
89	20	3	61.8	18.5	**58.6**	18.9
89	40	26	**12.7**	10.5	12.8	10.5
96	20	12	**22.3**	6.3	23.3	6.0
96	40	34	**3.9**	3.1	4.0	3.1
98	20	4	**36.8**	8.7	37.8	8.5
98	40	27	**6.1**	4.8	6.2	4.9
99	20	13	22.4	3.2	**21.0**	3.2
99	40	35	3.9	1.3	**3.7**	1.3
101	20	14	48.7	25.5	**46.7**	24.8
101	40	36	**16.6**	16.4	**16.6**	16.3
102	20	15	28.4	13.9	**27.7**	13.4
102	40	37	**9.9***	9.9	**9.9***	9.9
198	20	16	35.0	24.6	**34.6**	24.6
198	40	38	**22.0***	22.0	**22.0***	22.0
220	20	17	**45.7**	0.0	49.5	0.0
220	40	39	**7.5**	0.0	**7.5**	0.0
225	20	5	79.8	0.0	**73.5**	0.0
225	40	28	**9.7**	0.0	9.8	0.0
254	20	18	36.8	0.0	**34.1**	0.0
254	40	40	5.7	0.0	**4.9**	0.0
457	20	6	155.0	0.0	**118.7**	0.0
457	40	29	63.0	0.0	**20.5**	0.0
489	20	19	112.4	0.0	**45.3**	0.0
489	40	41	27.8	0.0	**9.9**	0.0
567	20	20	80.9	0.0	**41.3**	0.0
567	40	42	**56.0**	0.0	201.9	0.0
575	20	21	164.9	0.0	**70.9**	0.0
575	40	43	61.0	0.0	**27.5**	0.0
1179	20	22	251.8	0.0	**87.7**	0.0
1179	40	44	60.7	0.0	**27.6**	0.0
1319	20	7	**401.2**	0.0	447.8	0.0
1319	40	30	246.1	0.0	**163.7**	0.0
2140	20	23	533.5	0.0	**250.1**	0.0
2140	40	45	732.9	0.0	**119.4**	0.0
2152	20	8	**204.5**	0.0	240.7	0.0
2152	40	31	70.3	0.0	**58.7**	0.0
		Average	89.6	9.6	**60.6**	9.6

Table 2.8: Best in-sample objective function values (scaled by 10^6) and corresponding out-of-sample root-mean-squared errors (RMSE) expressed as averages over all runs with different random seeds. M-ST: MIQP approach based on the formulation (M-ST); GA: Algorithm 2.7; TSA: two-stage approach; bold values indicate the best approach in terms of the objective function value or RMSE for each instance.

Instance			in-sample objective function value						out-of-sample RMSE in [%]					
			120 seconds			1000 seconds			120 seconds			1000 seconds		
n	k	No.	M-ST	GA	TSA	M-ST	GA	TSA	M-ST	GA	TSA	M-ST	GA	TSA
20	20	9	**90.9**	311.3	**90.9**	**90.9**	311.3	**90.9**	**2.4**	4.3	**2.4**	**2.4**	4.3	**2.4**
31	20	1	**56.1**	194.2	**56.1**	**56.1**	194.2	**56.1**	**3.3**	5.0	**3.3**	**3.3**	5.0	**3.3**
31	31	24	**47.3**	194.2	**47.3**	**47.3**	194.2	**47.3**	**3.3**	5.0	**3.3**	**3.3**	5.0	**3.3**
49	20	10	**27.8**	61.1	**27.8**	**27.8**	61.1	**27.8**	**2.3**	2.9	**2.3**	**2.3**	2.9	**2.3**
49	40	32	**13.3**	60.8	**13.3**	**13.3**	60.8	**13.3**	**1.6**	2.9	**1.6**	**1.6**	2.9	**1.6**
50	20	11	**23.6**	48.8	24.4	**23.6**	48.0	24.0	**1.9**	1.9	2.0	**1.9**	2.0	2.0
50	40	33	**5.9**	25.0	**5.9**	**5.9**	24.7	**5.9**	**1.0**	1.4	**1.0**	**1.0**	1.4	**1.0**
85	20	2	24.6	52.2	**24.1**	23.9	52.1	**23.9**	2.5	2.6	**2.4**	2.6	2.5	**2.3**
85	40	25	**3.7**	46.5	3.8	**3.7**	45.8	**3.7**	**1.6**	2.5	1.6	**1.6**	2.6	**1.6**
89	20	3	**58.6**	98.8	68.7	**57.8**	98.2	65.7	**2.7**	2.8	3.1	**2.7**	2.8	3.2
89	40	26	**12.8**	43.2	13.1	**12.4**	41.9	**12.4**	1.9	2.0	**1.7**	**1.9**	2.1	**1.9**
96	20	12	**23.3**	47.3	23.3	**21.9**	46.0	23.1	2.2	2.4	**2.2**	2.2	2.4	**2.2**
96	40	34	4.0	32.1	**3.9**	**3.9**	30.3	3.9	1.4	2.1	**1.3**	1.4	2.2	**1.3**
98	20	4	**37.8**	74.5	43.7	35.3	73.1	**35.3**	**2.9**	3.0	2.9	3.0	3.0	**3.0**
98	40	27	**6.2**	38.7	7.1	**5.9**	34.5	**5.9**	2.0	2.3	**1.7**	**2.1**	2.1	**2.1**
99	20	13	**21.0**	38.2	21.3	**19.8**	38.2	21.3	2.9	2.8	**2.5**	2.9	2.7	**2.5**
99	40	35	**3.7**	17.2	**3.7**	3.6	16.4	**3.6**	**1.8**	2.0	1.9	**1.8**	2.0	1.9
101	20	14	**46.7**	127.1	47.0	**45.8**	127.1	46.7	3.4	4.4	**3.4**	3.7	4.4	**3.5**
101	40	36	**16.6**	126.1	16.6	**16.6**	125.4	**16.6**	**2.5**	4.2	**2.5**	**2.5**	4.3	**2.5**
102	20	15	27.7	61.6	**27.7**	**27.6**	61.6	27.7	**2.1**	2.4	2.1	**2.1**	2.4	2.1
102	40	37	**9.9**	54.8	**9.9**	**9.9**	54.1	**9.9**	**1.4**	2.3	**1.4**	**1.4**	2.3	**1.4**
198	20	16	**34.6**	114.3	34.9	**34.4**	113.6	**34.4**	2.2	3.5	**2.2**	**2.2**	3.1	**2.2**
198	40	38	**22.0**	113.3	22.0	**22.0**	112.1	**22.0**	2.6	3.1	**2.4**	**2.6**	3.0	**2.6**
220	20	17	49.5	78.6	**39.2**	**33.7**	76.2	38.6	3.8	**3.7**	3.9	**3.7**	3.8	3.9
220	40	39	7.5	35.5	**4.8**	5.1	33.9	**4.5**	3.1	**3.0**	3.3	3.1	**2.9**	3.0
225	20	5	73.5	84.0	**48.9**	**44.9**	80.4	45.0	**3.4**	3.5	3.9	**3.3**	3.6	3.9
225	40	28	9.8	23.0	**6.4**	**5.7**	22.3	6.1	2.7	**2.4**	2.6	2.5	**2.5**	2.6
254	20	18	34.1	69.8	**21.5**	**19.5**	69.8	20.1	**2.2**	3.2	2.4	**2.3**	3.2	2.4
254	40	40	4.9	55.7	**3.8**	**2.8**	48.8	3.1	**1.6**	3.1	1.9	1.7	3.4	**1.7**
457	20	6	118.7	95.5	**58.9**	82.1	88.9	**54.6**	6.4	6.2	**5.9**	6.5	**6.0**	6.1
457	40	29	20.5	28.0	**6.6**	12.3	22.4	**6.4**	4.9	5.0	**4.8**	**4.8**	4.8	4.8
489	20	19	45.3	33.8	**22.8**	30.2	32.1	**21.4**	**3.3**	3.6	3.5	**3.2**	3.4	3.5
489	40	41	9.9	10.0	**2.9**	5.4	8.1	**2.3**	2.5	2.6	**2.5**	2.5	2.6	**2.4**
567	20	20	41.3	43.6	**22.6**	26.5	43.4	**20.8**	**2.5**	3.0	2.6	2.6	3.0	**2.5**
567	40	42	201.9	15.2	**2.8**	4.7	13.2	**2.3**	3.0	2.2	**2.0**	**1.9**	2.2	2.0
575	20	21	70.9	39.5	**24.5**	43.0	39.3	**22.9**	**3.4**	3.7	3.5	**3.4**	3.6	3.5
575	40	43	27.5	15.4	**3.1**	6.9	11.0	**3.0**	**2.4**	2.5	2.6	2.7	**2.4**	2.6
1179	20	22	87.7	35.0	**23.9**	59.3	34.9	**23.3**	3.9	3.4	**3.4**	3.5	3.4	**3.4**
1179	40	44	27.6	16.1	**6.7**	12.1	8.3	**2.0**	3.5	**2.8**	2.9	3.3	**2.9**	3.1
1319	20	7	447.8	156.7	**115.2**	309.0	156.7	**113.6**	13.8	11.2	**10.5**	14.5	11.2	**10.6**
1319	40	30	163.7	**64.6**	**64.6**	68.6	41.9	**11.1**	10.6	**9.0**	**9.0**	12.1	**9.1**	9.3
2140	20	23	250.1	65.6	**49.0**	249.8	65.6	**45.8**	6.6	4.3	**4.1**	6.6	**4.3**	4.4
2140	40	45	119.4	**62.0**	**62.0**	119.4	27.6	**5.1**	6.0	**4.9**	**4.9**	6.0	4.6	**4.2**
2152	20	8	240.7	78.6	**58.3**	233.8	78.6	**53.6**	8.3	**6.8**	6.9	8.4	**6.8**	6.9
2152	40	31	**58.7**	63.6	63.6	55.2	25.8	**5.5**	8.4	**4.8**	**4.8**	8.3	**4.9**	5.4
		Average	60.6	70.0	**30.0**	45.3	66.5	**25.2**	3.5	3.6	**3.2**	3.5	3.6	**3.2**

Table 2.9: Friedman ranks.

	in-sample		out-of-sample	
	120 s	1000 s	120 s	1000 s
M-ST	1.88	1.79	1.88	1.90
GA	2.70	2.80	2.46	2.36
TSA	1.42	1.41	1.67	1.74

Table 2.10: p-values of multiple comparisons between all algorithms. Cases in which the null hypothesis can be rejected at a significance level of $\alpha = 0.1$ are marked in bold. Post hoc procedures: Nemenyi, Holm, Shaffer, Berg.

		in-sample				out-of-sample			
	Hypothesis	Neme	Holm	Shaf	Berg	Neme	Holm	Shaf	Berg
120 s	GA vs. TSA	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	M-ST vs. GA	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.01**	**0.01**	**0.01**
	M-ST vs. TSA	**0.09**	**0.03**	**0.03**	**0.03**	0.95	0.32	0.32	0.32
1000 s	GA vs. TSA	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.01**	**0.01**	**0.01**
	M-ST vs. GA	**0.00**	**0.00**	**0.00**	**0.00**	**0.09**	**0.06**	**0.03**	**0.03**
	M-ST vs. TSA	0.22	**0.07**	**0.07**	**0.07**	1.00	0.46	0.46	0.46

the MIQP approach performed significantly better than the pure genetic algorithm. For the out-of-sample period, even though the difference between the MIQP approach and the two-stage approach is not statistically significant, the results are similar to those obtained for the in-sample period.

2.4.3.4 Impact of UCITS concentration rule

Table 2.11 shows for all instances the results of the two-stage approach when the UCITS concentration rule is ignored. Columns two to four show the results for the instances with $k = 20$. The results of the remaining instances are shown in columns five to seven. From Table 2.11, we can gain the following main insights for the instances with $k = 20$:

- When the concentration rule is ignored, the determined portfolios do not satisfy the concentration rule.
- Ignoring the concentration rule allows to construct portfolios with a smaller objective function value.

- The out-of-sample root-mean-squared errors increase when the concentration rule is ignored. Hence, we can argue that the concentration rule reduces the portfolio risk in terms of out-of-sample root-mean-squared error.

For the instances with a larger value of k, the impact of the concentration rule on the results can be neglected, because the average portfolio weights decrease with larger values of k, and portfolio weights larger than the UCITS lower concentration-rule threshold occur less often.

Table 2.11: Impact of the UCITS concentration rule; $\sum_{i \in I: w_i > 0.05} w_i$: sum of weights in the best portfolio that exceed the UCITS lower threshold; DIFF OBJ and DIFF RMSE: difference of the best objective function values (scaled by 10^6) and the best out-of-sample root-mean-squared errors (in percentage points), respectively, between the two-stage approach when the concentration rule is ignored and the two-stage approach when the concentration rule is considered; values expressed as averages over all runs with different random seeds; negative values indicate lower values for the case when the concentration rule is ignored; time limit: 120 seconds.

	$k = 20$			$k > 20$		
n	$\sum_{i \in I: w_i > 0.05} w_i$	DIFF OBJ	DIFF RMSE	$\sum_{i \in I: w_i > 0.05} w_i$	DIFF OBJ	DIFF RMSE
20	0.69	−4.25	0.33	–	–	–
31	0.69	−8.38	0.47	0.65	−2.91	0.16
49	0.64	−1.63	0.06	0.31	0.00	0.00
50	0.62	−0.76	0.00	0.19	0.00	0.00
85	0.64	−2.81	0.09	0.45	−0.17	−0.05
89	0.60	−3.59	−0.01	0.16	0.00	0.00
96	0.58	−0.55	0.05	0.36	0.00	0.00
98	0.57	−2.47	0.16	0.21	0.00	0.00
99	0.60	−1.23	0.16	0.11	0.00	−0.01
101	0.58	−1.33	0.59	0.26	0.00	0.00
102	0.66	−2.15	−0.08	0.30	0.00	0.00
198	0.58	−1.06	0.01	0.33	0.00	0.00
220	0.57	0.97	0.18	0.22	−0.10	−0.06
225	0.57	0.67	−0.03	0.05	−0.12	−0.03
254	0.65	−1.61	0.14	0.39	−0.18	−0.04
457	0.56	−1.51	0.21	0.12	0.00	0.00
489	0.58	−2.17	0.02	0.04	0.00	0.00
567	0.61	−2.57	0.11	0.18	0.00	0.00
575	0.62	1.65	0.26	0.10	0.00	0.00
1179	0.59	−2.30	−0.03	0.01	−0.62	0.04
1319	0.62	−6.73	−0.09	0.00	0.00	0.00
2140	0.67	−2.68	0.47	0.00	0.00	0.00
2152	0.61	−1.71	0.02	0.00	0.00	0.00
∅	0.61	−2.10	0.13	0.20	−0.19	0.00

2.5 Conclusion

We presented a hybrid two-stage approach to the UCITS-constrained index-tracking problem based on a genetic algorithm and local branching. For the genetic algorithm, we presented a new representation of subsets, and for the local-branching method, we presented a novel MIQP formulation of the UCITP. We tested the proposed two-stage approach in a computational experiment based on real-world data. The results demonstrate that the proposed two-stage approach yields significantly better results than either a pure genetic algorithm or a pure MIQP approach within a limited running time.

Future research should investigate whether the two-stage approach's performance can be improved by exploiting the biased redundancy of the new subset representation. Furthermore, additional practical portfolio constraints could be considered, such as those presented by Filippi et al. (2016), Guastaroba and Speranza (2012), and Strub and Baumann (2018). A further promising direction for future research would be to investigate the performance of genetic algorithms based on the new subset representation for other optimization problems that involve the selection of a subset, such as the feature-selection problem in machine learning.

Appendix

2.A Further algorithms

Algorithm 2.6 Mutation

procedure MUTATE($\mathbf{g}^1 \in \{1, \ldots, n\}^{d_1}$)
    $\mathbf{g}^2 := \mathbf{g}^1$; $d_2 := d_1$
    if $random < p_e$ **then**
        Randomly choose $i \in \{1, \ldots, d_2\}$, $j \in \{1, \ldots, n\}$; $g_i^2 := j$
    end if
    if $random < p_r$ **then**
        Randomly choose $i \in \{1, \ldots, d_2\}$; Remove element g_i^2 from $\mathbf{g}^2$; $d_2 := d_2 - 1$
    end if
    if $random < p_a$ **then**
        Randomly choose $j \in \{1, \ldots, n\}$; Add j to $\mathbf{g}^2$; $d_2 := d_2 + 1$
    end if
    if $\underline{d} \leq d_2 \leq \overline{d}$ **then return** $\mathbf{g}^2$ **else return** $\mathbf{g}^1$ **end if**
end procedure

Algorithm 2.7 Genetic Algorithm (GA) – Stage 1

```
procedure GA
    P := ∅
    for all i ∈ {1, ..., s} do
        Randomly choose d_i ∈ {d̲, ..., d̄}
        Randomly choose g^i ∈ {1, ..., n}^{d_i}
        if i = 1 ∨ f(g^i) < f(b) then b := g^i end if
        P := P ∪ {g^i}
    end for
    while Number of generations < n_g do
        M := ∅
        while |M| < s do
            Randomly select individuals g^1, g^2 ∈ P
            if f(g^1) ≤ f(g^2) then
                M := M ∪ {g^1}
            else
                M := M ∪ {g^2}
            end if
        end while
        P' := ∅, i := 1
        while i ≤ s do
            if random < p_c then
                (g^i, g^{i+1}) := CrossOver(g^i ∈ M, g^{i+1} ∈ M)
            end if
            P' := P' ∪ {g^i} ∪ {g^{i+1}}; i := i + 2
        end while
        for all g^i ∈ P' do
            g^i := Mutate(g^i)
            if f(g^i) < f(b) then b := g^i end if
        end for
        P := P'
    end while
    return b
end procedure
```

Algorithm 2.8 Crossover

procedure CrossOver($\mathbf{g}^1 \in \{1, \ldots, n\}^{d_1}$, $\mathbf{g}^2 \in \{1, \ldots, n\}^{d_2}$)
    $d_3 := d_1$; $d_4 := d_2$
    Initialize $\mathbf{g}^3 \in \{1, \ldots, n\}^{d_3}$, $\mathbf{g}^4 \in \{1, \ldots, n\}^{d_4}$
    Randomly choose $m \in \{1, \ldots, \min\{d_1, d_2\} + 1\}$
    for all $i \in \{1, \ldots, d_3\}$ **do**
        if $i < m$ **then** $g_i^3 := g_i^2$ **else** $g_i^3 := g_i^1$ **end if**
    end for
    for all $i \in \{1, \ldots, d_4\}$ **do**
        if $i < m$ **then** $g_i^4 := g_i^1$ **else** $g_i^4 := g_i^2$ **end if**
    end for
    return $\mathbf{g}^3$ and $\mathbf{g}^4$
end procedure

Bibliography

Adcock, S. A., 2017. GAUL: The genetic algorithm utility library. URL http://gaul.sourceforge.net/gaul_reference_guide.html

Andriosopoulos, K., Nomikos, N., 2014. Performance replication of the Spot Energy Index with optimal equity portfolio selection: Evidence from the UK, US and Brazilian markets. European Journal of Operational Research 234 (2), 571–582.

Beasley, J. E., 1990. OR-Library: Distributing test problems by electronic mail. Journal of the Operational Research Society 41 (11), 1069–1072.

Beasley, J. E., Meade, N., Chang, T.-J., 2003. An evolutionary heuristic for the index tracking problem. European Journal of Operational Research 148 (3), 621–643.

Bertolazzi, P., Felici, G., Festa, P., Fiscon, G., Weitschek, E., 2016. Integer programming models for feature selection: New extensions and a randomized solution algorithm. European Journal of Operational Research 250 (2), 389–399.

Bertsimas, D., King, A., 2015. OR forum–an algorithmic approach to linear regression. Operations Research 64 (1), 2–16.

Bertsimas, D., King, A., Mazumder, R., 2016. Best subset selection via a modern optimization lens. The Annals of Statistics 44 (2), 813–852.

Brill, F. Z., Brown, D. E., Martin, W. N., 1992. Fast generic selection of features for neural network classifiers. IEEE Transactions on Neural Networks 3 (2), 324–328.

Busse, J. A., Goyal, A., Wahal, S., 2010. Performance and persistence in institutional investment management. The Journal of Finance 65 (2), 765–790.

Canakgoz, N. A., Beasley, J. E., 2009. Mixed-integer programming approaches for index tracking and enhanced indexation. European Journal of Operational Research 196 (1), 384–399.

Chen, C., Kwon, R. H., 2012. Robust portfolio selection for index tracking. Computers & Operations Research 39 (4), 829–837.

Chiam, S. C., Tan, K. C., Al Mamun, A., 2008. Evolutionary multi-objective portfolio optimization in practical context. International Journal of Automation and Computing 5 (1), 67–80.

Chiam, S. C., Tan, K. C., Al Mamun, A., 2013. Dynamic index tracking via multi-objective evolutionary algorithm. Applied Soft Computing 13 (7), 3392–3408.

Corielli, F., Marcellino, M., 2006. Factor based index tracking. Journal of Banking & Finance 30 (8), 2215–2233.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., 2001. Introduction to algorithms. Vol. 6. MIT press, Cambridge.

Diosan, L., 2005. A multi-objective evolutionary approach to the portfolio optimization problem. In: International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06). Vol. 2. IEEE, pp. 183–187.

European Fund and Asset Management Association (EFAMA), 2017. 2017 EFAMA Annual Report. European Fund and Asset Management Association (EFAMA).

Filippi, C., Guastaroba, G., Speranza, M., 2016. A heuristic framework for the bi-objective enhanced index tracking problem. Omega 65, 122–137.

Fischetti, M., Lodi, A., 2003. Local branching. Mathematical Programming 98 (1–3), 23–47.

Gaivoronski, A. A., Krylov, S., Van der Wijst, N., 2005. Optimal portfolio selection and dynamic benchmark tracking. European Journal of Operational Research 163 (1), 115–131.

Garcia, S., Herrera, F., 2008. An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research 9 (Dec), 2677–2694.

Gilli, M., Schumann, E., 2012. Heuristic optimisation in financial modelling. Annals of Operations Research 193 (1), 129–158.

Gottlieb, J., Julstrom, B. A., Raidl, G. R., Rothlauf, F., 2001. Prüfer numbers: A poor representation of spanning trees for evolutionary search. In: Spector, L., et al. (Eds.), Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation. GECCO'01. Morgan Kaufmann Publishers Inc., San Francisco, pp. 343–350.

Guastaroba, G., Speranza, M. G., 2012. Kernel Search: An application to the index tracking problem. European Journal of Operational Research 217 (1), 54–68.

Investment Company Institue, 2017. 2017 Fact Book. Investment Company Institute.

Kolm, P. N., Tütüncü, R., Fabozzi, F. J., 2014. 60 years of portfolio optimization: Practical challenges and current trends. European Journal of Operational Research 234 (2), 356–371.

Konno, H., Wijayanayake, A., 2001. Minimal cost index tracking under nonlinear transaction costs and minimal transaction unit constraints. International Journal of Theoretical and Applied Finance 4 (6), 939–957.

Krink, T., Mittnik, S., Paterlini, S., 2009. Differential evolution and combinatorial search for constrained index-tracking. Annals of Operations Research 172 (1), 153–176.

Kuncheva, L. I., Jain, L. C., 1999. Nearest neighbor classifier: simultaneous editing and feature selection. Pattern Recognition Letters 20 (11), 1149–1156.

Kwiatkowski, J. W., 1992. Algorithms for index tracking. IMA Journal of Management Mathematics 4 (3), 279–299.

López-Ibánez, M., Dubois-Lacoste, J., Stützle, T., Birattari, M., 2011. The irace package, iterated race for automatic algorithm configuration. Tech. rep., TR/IRIDIA/2011-004, IRIDIA, Université Libre de Bruxelles, Belgium.

Malkiel, B. G., 1995. Returns from investing in equity mutual funds 1971 to 1991. The Journal of Finance 50 (2), 549–572.

Maringer, D., Oyewumi, O., 2007. Index tracking with constrained portfolios. Intelligent Systems in Accounting, Finance and Management 15 (1–2), 57–71.

Montfort, K. v., Visser, E., van Draat, L. F., 2008. Index tracking by means of optimized sampling. The Journal of Portfolio Management 34 (2), 143–152.

Moral-Escudero, R., Ruiz-Torrubiano, R., Suárez, A., 2006. Selection of optimal investment portfolios with cardinality constraints. In: 2006 IEEE International Conference on Evolutionary Computation. IEEE, pp. 2382–2388.

Mutunge, P., Haugland, D., 2018. Minimizing the tracking error of cardinality constrained portfolios. Computers & Operations Research 90, 33–41.

Oh, I.-S., Lee, J.-S., Moon, B.-R., 2004. Hybrid genetic algorithms for feature selection. IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (11), 1424–1437.

Raymer, M. L., Punch, W. F., Goodman, E. D., Kuhn, L. A., Jain, A. K., 2000. Dimensionality reduction using genetic algorithms. IEEE Transactions on Evolutionary Computation 4 (2), 164–171.

Roll, R., 1992. A mean/variance analysis of tracking error. The Journal of Portfolio Management 18 (4), 13–22.

Rothlauf, F., 2011. Design of modern heuristics: principles and application. Springer, Berlin, Heidelberg.

Rudd, A., 1980. Optimal selection of passive portfolios. Financial Management, 57–66.

Rudolf, M., Wolter, H.-J., Zimmermann, H., 1999. A linear model for tracking error minimization. Journal of Banking & Finance 23 (1), 85–103.

Ruiz-Torrubiano, R., Suárez, A., 2007. Use of heuristic rules in evolutionary methods for the selection of optimal investment portfolios. In: 2007 IEEE Congress on Evolutionary Computation. IEEE, pp. 212–219.

Ruiz-Torrubiano, R., Suárez, A., 2009. A hybrid optimization approach to index tracking. Annals of Operations Research 166 (1), 57–71.

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

Sant'Anna, L. R., Filomena, T. P., Caldeira, J. F., 2017a. Index tracking and enhanced indexing using cointegration and correlation with endogenous portfolio selection. The Quarterly Review of Economics and Finance 65, 146–157.

Sant'Anna, L. R., Filomena, T. P., Guedes, P. C., Borenstein, D., 2017b. Index tracking with controlled number of assets using a hybrid heuristic combining genetic algorithm and non-linear programming. Annals of Operations Research 258 (2), 849–867.

Scozzari, A., Tardella, F., Paterlini, S., Krink, T., 2013. Exact and heuristic approaches for the index tracking problem with UCITS constraints. Annals of Operations Research 205 (1), 235–250.

Sharma, A., Agrawal, S., Mehra, A., 2017. Enhanced indexing for risk averse investors using relaxed second order stochastic dominance. Optimization and Engineering 18, 407–442.

Siedlecki, W., Sklansky, J., 1989. A note on genetic algorithms for large-scale feature selection. Pattern Recognition Letters 10 (5), 335–347.

Skolpadungket, P., Dahal, K., Harnpornchai, N., 2007. Portfolio optimization using multi-objective genetic algorithms. In: 2007 IEEE Congress on Evolutionary Computation. IEEE, pp. 516–523.

Streichert, F., Ulmer, H., Zell, A., 2004. Comparing discrete and continuous genotypes on the constrained portfolio selection problem. In: Deb, K. (Ed.), Genetic and Evolutionary Computation – GECCO 2004. GECCO 2004. Lecture Notes in Computer Science, vol 3103. Springer, Berlin, Heidelberg, pp. 1239–1250.

Strub, O., Baumann, P., 2018. Optimal construction and rebalancing of index-tracking portfolios. European Journal of Operational Research 264 (1), 370–387.

Strub, O., Trautmann, N., 2016. An application of Microsoft Excel's evolutionary solver based on a novel chromosome encoding scheme to the 1/N portfolio tracking problem. In: Suryadi, K., et al. (Eds.), Industrial Engineering and Engineering Management (IEEM), 2016 IEEE International Conference on. IEEE, Bali, pp. 745–749.

Strub, O., Trautmann, N., 2017. A genetic algorithm for the UCITS-constrained index-tracking problem. In: Evolutionary Computation (CEC), 2017 IEEE Congress on. IEEE, pp. 822–829.

Takeda, A., Niranjan, M., Gotoh, J., Kawahara, Y., 2013. Simultaneous pursuit of out-of-sample performance and sparsity in index tracking portfolios. Computational Management Science 10 (1), 21–49.

Paper III

Tracking and outperforming large stock-market indices[4]

Mario Gnägi Oliver Strub

Department of Business Administration
University of Bern

Contents

[4]Paper submitted to Omega

Abstract

Enhanced index-tracking funds aim to track the returns of a given financial benchmark index as closely as possible while outperforming that index by a small positive excess return. These funds are attractive to investors, especially when the index is large and thus well diversified. We consider the problem of determining a portfolio for an enhanced index-tracking fund that is benchmarked against a large stock-market index subject to real-life constraints that may be imposed by investors, stock exchanges, or investment guidelines. Existing approaches to enhanced index tracking exhibit one of the following shortcomings: they may not exploit information about the weights of the stocks in the index, they may neglect real-life constraints such as the minimum trading values imposed by stock exchanges, or they may not devise good feasible portfolios within a reasonable computational time when the index is large. To overcome these shortcomings, we present two matheuristic approaches based on a novel mixed-integer quadratic programming formulation. We tested both matheuristics on a novel set of problem instances based on large stock-market indices with up to more than 9,000 *constituents. Our computational results indicate that within a limited computational time, both matheuristics yield better feasible portfolios than benchmark approaches in terms of the objective function value and out-of-sample risk-return characteristics.*

3.1 Introduction

A stock-market index reflects the overall development of the stocks that constitute that index. Examples of such indices include the Standard & Poor's 500 index, the EURO STOXX 50 index, and the Thomson Reuters Global index, which reflect the development of national, regional, and global stock markets, respectively. Stock-market indices serve as benchmarks for evaluating the performance of professional managers of both active and passive investment funds. A passive fund, also known as an index-tracking fund, aims to replicate the return of an index, whereas an active fund aims to achieve an excess return over its benchmark index. Passive funds tend to be less risky and incur lower management costs than active funds (cf. Beasley et al., 2003). However, active funds have a higher potential return. Recently, a new type of investment fund has emerged, so-called enhanced index-tracking funds, which are based on the idea of combining the advantages of both active and passive funds by aiming at a small target excess return with minimum additional risk relative to the index, i.e., a minimum tracking error (cf. Filippi et al., 2016). Note that we regard index-tracking funds as a special type of enhanced index-tracking funds with a target excess return of zero. Enhanced index-tracking funds are attractive to investors, especially when such a fund is benchmarked against an index that has a large number of constituents and thus is well diversified.

We consider the enhanced index-tracking problem (EITP) faced by the portfolio manager of an enhanced index-tracking fund that is benchmarked against a large stock-market index. In the EITP, the portfolio manager is given the current composition of the index and the current composition of the portfolio, which can consist of stocks from the index and cash. The portfolio manager can receive cash deposits and cash withdrawal requests. The available investment budget consists of the net cash flow from deposits and withdrawals plus the value of the current portfolio. Furthermore, the portfolio manager is given the following data from the past, i.e., the in-sample period: the values of the index, the prices of the stocks that currently constitute the index, and the interest rates on cash. The portfolio manager needs to decide how to revise (rebalance) the current portfolio such that the rebalanced portfolio will exhibit a small tracking error and achieve a given target excess return in the future, i.e., the out-of-sample period. Because future outcomes are not known in advance, the portfolio manager aims to minimize the expected tracking error subject to a constraint that prescribes some minimum expected excess return. When rebalancing the portfolio, the manager must consider a budget constraint that ensures that the investment in the stocks plus the total transaction costs spent for rebalancing do not exceed the investment budget. Furthermore, the portfolio manager must also consider various real-life constraints that may be imposed by investment guidelines, the investors, or stock exchanges. Specifically, we consider the following real-life constraints, which are common both in the literature and in practice (cf., e.g., Filippi et al., 2016; Guastaroba and Speranza, 2012; Strub and Baumann, 2018). The number of stocks included in the portfolio, i.e., the portfolio cardinality, must not exceed a given upper bound because investing in all constituents of a large index would be impractical due to the consequent prohibitive management costs. The trading value of each traded stock and the weight of each stock in the portfolio must be within given ranges. The total proportional and fixed transaction costs spent for rebalancing must not exceed a given fraction of the investment budget. Finally, the short selling of stocks is prohibited, and it is assumed that fractional units of stocks can be traded. Note that the EITP also includes the construction of a new portfolio as a special case when the portfolio before rebalancing consists only of cash.

In the literature, various mathematical programming formulations have been proposed for the problem of determining an enhanced index-tracking portfolio. These formulations differ with respect to the real-life constraints considered, the way the expected tracking error is attempted to be minimized, and whether and how the expected excess return is integrated. With respect to the real-life constraints, some authors have determined enhanced index-tracking portfolios without considering real-life constraints (cf., e.g., Roll, 1992), whereas others have considered all real-life constraints as defined in the EITP (cf., e.g., Strub and Baumann, 2018). With respect to the expected tracking error, the earliest studies attempted to directly minimize the expected tracking error as defined in the EITP, which is a quadratic function of the expected covariances between the returns of the stocks, the weights of the stocks in the portfolio, and the weights of the stocks in the index (cf., e.g., Roll, 1992). By contrast, later studies attempted to minimize the expected tracking error indirectly by using as the objective function either a quadratic (cf., e.g., Beasley et al., 2003) or a linear (cf., e.g., Filippi et al., 2016) dissimilarity function that captured the total deviations between the historical value developments or the historical returns of the portfolio and the index. In the most recent study, the goal of minimizing the expected tracking error directly was revisited (cf. Mutunge and Haugland, 2018). With respect to the expected excess return, some studies have focused on the problem of determining the portfolio for an index-tracking fund without considering the expected excess return (cf., e.g., Strub and Baumann, 2018). In other studies, the expected excess return has been considered by using a bi-objective approach with the maximization of the expected excess return as a second competing objective (cf., e.g., Filippi et al., 2016), by introducing into the objective function a second term that captures the expected excess return (cf., e.g., Beasley et al., 2003), or by introducing a constraint that prescribes a minimum expected excess return (cf., e.g., Roll, 1992). From an optimization point of view, these various means of integrating the expected excess return are very similar because all functions used for the expected excess return are linear. Various exact approaches, such as mixed-integer programming, and metaheuristic approaches, such as population-based heuristics or local-search heuristics, have all been proposed as solution approaches for the problem of determining an enhanced index-tracking portfolio.

We have identified four gaps in the literature on enhanced index tracking. Gap 1: it remains an open question whether it is preferable in terms of the out-of-sample tracking error and the out-of-sample excess return to use a linear dissimilarity function as the objective function or whether it is better to use the expected tracking error itself, which, together with the real-life constraints, constitutes a cardinality-constrained quadratic optimization problem that is known to be very challenging to solve (cf. Bertsimas and Shioda, 2009; Wu et al., 2017). Gap 2: the existing mathematical programming formulations for the problem of determining an enhanced index-tracking portfolio that consider transaction costs implicitly allow the holding of cash because the budget constraint is modeled as an inequality or because the modeled transaction costs correspond to merely an upper bound on the true transaction costs. Consequently, these cash holdings are not considered in the formulation of the expected tracking error and expected excess return. Gap 3: the EITP as defined above has not previously been considered because the problems studied in the literature either may not involve the minimization of the expected tracking error itself or may neglect some of the real-life constraints of the EITP. The existing solution approaches for the related problems studied in the literature may not be appropriate for the EITP. The existing exact approaches would require the solution of a series of quadratic programming relaxations, which may become computationally very expensive when large indices are considered. The existing metaheuristic approaches would require adaptation to the real-life constraints of the EITP, which may reduce their effectiveness because they are tailored for other specific problems that are less constrained. Gap 4: there are no available instances of the EITP based on large stock-market indices; the existing instances of related problems either are based on small indices or do not provide information about the index composition.

The contributions of this paper are fourfold. First, to address gap 1, we present new arguments that using the expected tracking error itself instead of a linear dissimilarity function as the objective function may lead to superior out-of-sample tracking errors and out-of-sample excess returns, especially when the index is large, because dissimilarity functions fail to exploit the known index composition. Second, to address gap 2, we present a novel mixed-integer quadratic programming (MIQP) formulation and two novel benchmark mixed-integer linear programming (MILP) formulations for the problem of determining an enhanced index-tracking portfolio. In the MIQP formulation, we use the expected tracking error as the objective function. In the two benchmark MILP formulations, we use the two dissimilarity functions proposed by Guastaroba and Speranza (2012) and Filippi et al. (2016) as the objective functions. For all three formulations, we present a novel modeling of the considered real-life constraints in which cash holdings are explicitly considered. We also strengthen the three formulations by removing redundant variables and constraints. Third, to address gap 3, we present two matheuristics based on our novel MIQP formulation that are able to determine good feasible portfolios, i.e., portfolios that satisfy all considered real-life constraints, within a reasonable computational time for instances of the EITP based on large indices. Matheuristics are particularly suitable for the EITP because they combine the flexibility of mathematical programming to easily incorporate complex constraints such as the considered real-life constraints with the ability of heuristics to find good feasible solutions quickly. The two matheuristics are initialized with the same construction heuristic but differ with respect to the applied improvement heuristics. The construction heuristic, which is used to find an initial feasible portfolio, is based on linearizing the expected tracking error by using the identity matrix as a simplified covariance matrix and by considering absolute instead of squared deviations in the terms of the resulting function. The first improvement heuristic is based on the concept of local branching, which has been successfully applied to various combinatorial optimization problems (cf. Fischetti and Lodi, 2003). In local branching, starting from the initial feasible solution, the solution space to be searched is iteratively defined with an upper bound on the number of binary variables whose values flip. The novelty of this improvement heuristic is that we consider a subset of promising stocks that differs in each iteration to reduce the required computational time. The second improvement heuristic is based on the concept of iterated greedy heuristics (cf., e.g., Ruiz and Stützle, 2007). In iterated greedy heuristics, a current feasible solution is iteratively deconstructed and subsequently reconstructed in a greedy manner to form a new feasible solution. The novelties of this improvement heuristic are that we also consider a different subset of promising stocks in each iteration and that, in contrast to existing iterated greedy heuristics (cf., e.g., Strub and Trautmann, 2016), we apply mixed-integer quadratic programming for the reconstruction. Finally, to address gap 4, we generated a set of 18 instances of the EITP based on large regional and global real-world stock-market indices maintained by Thomson Reuters. The largest of these indices has more than 9,000 constituents. Based on these instances, we conducted a computational experiment, which yielded the following two main results: 1) the proposed matheuristics are able to determine, within a limited computational time, considerably better feasible portfolios in terms of the objective function value than those determined by using an exact approach based on the MIQP formulation, and 2) in terms of the out-of-sample tracking error and the out-of-sample excess return, the feasible portfolios determined by using the two matheuristics are superior to the feasible portfolios determined by using an exact approach based on either of the two benchmark MILP formulations.

The remainder of this paper is organized as follows. In Section 3.2, we review the existing solution approaches in the literature for determining the portfolio for an enhanced index-tracking fund. In Section 3.3, we present the MIQP formulation and the two benchmark MILP formulations. In Section 3.4, we present the two matheuristics. In Section 3.5, we report the results of our computational experiment. In Section 3.6, we offer some concluding remarks and an outlook on future research.

3.2 Related literature

Various papers in the literature have studied problems that are related to the EITP. Table 3.1 lists, for each of these papers, whether it considers the real-life constraints of the EITP mentioned above and whether the objective is index tracking (IT) or enhanced index tracking (EIT). We categorize the papers into two groups based on whether the objective function used is non-linear or linear. In the following, we describe the proposed solution approaches of both groups.

The first group of problems consists of those that involve the optimization of a non-linear objective function. In some papers, only indices with a small number of constituents are considered, such that exact approaches are applicable (cf. Gaivoronski et al., 2005; Jansen and Van Dijk, 2002; Rudd, 1980). In other papers, the real-life constraints are neglected, which allows closed-form solutions to be devised (cf. Jorion, 2003; Roll, 1992). In the remaining papers, metaheuristics such as evolutionary algorithms (cf. Andriosopoulos and Nomikos, 2014; Chiam et al., 2013; Krink et al., 2009; Maringer and Oyewumi, 2007; Sant'Anna et al., 2017a,b; Scozzari et al., 2013) or local-search heuristics (cf. Kwiatkowski, 1992; Mutunge and Haugland, 2018; Takeda et al., 2013) are proposed. The majority of the papers in this first group neglect most of the real-life constraints of the EITP. An exception is the paper by Beasley et al. (2003), in which the goal is to optimize the trade-off between a non-linear dissimilarity function and the expected excess return subject to a cardinality constraint, minimum and maximum weights for the stocks included in the portfolio, and a budget for proportional transaction costs. An evolutionary algorithm is presented that uses cross-over and mutation operators to combine and modify, respectively, individuals that represent feasible and infeasible solutions. The presented algorithm includes a customized procedure for determining portfolio weights, a repair operator, and a penalty term in the objective function to handle infeasible solutions.

The second group of problems consists of those that involve the optimization of a linear objective function. For these problems, exact approaches such as linear programming and MILP approaches are able to devise good feasible solutions within a reasonable computational time, even when real-life constraints and large indices are considered (cf. Bruni et al., 2015; Filippi et al., 2016; Guastaroba et al., 2016; Guastaroba and Speranza, 2012; Rudolf et al., 1999; Strub and Baumann, 2018). Among all these problems, those studied in the following papers are most similar to the EITP in terms of the real-life constraints considered. Strub and Baumann (2018) introduce a MILP formulation for determining the portfolio for an index-tracking fund in which a linear dissimilarity function is minimized subject to all real-life constraints of the EITP. Guastaroba and Speranza (2012) minimize the mean absolute deviation (MAD) between the historical values of the index and the portfolio, which is modeled as a linear dissimilarity function, subject to a budget for fixed and proportional transaction costs, minimum and maximum portfolio weights, and a cardinality constraint. They also present a heuristic called Kernel Search, which is a matheuristic that can easily handle various real-life constraints. In this heuristic, the information from the solution to the linear programming relaxation is exploited to construct different sub-problems that can be solved quickly. They also show that their heuristic can be applied for enhanced index tracking by tracking an artificial index that represents the index return plus the target excess return. Filippi et al. (2016) aim to maximize a linear excess-return function and minimize the same linear dissimilarity function subject to the same real-life constraints as those of Guastaroba and Speranza (2012). They modify the Kernel Search heuristic such that it can be applied to the considered problem. In the MILP formulation presented by Strub and Baumann (2018), implicit cash holdings can occur because the budget constraint is modeled as an inequality, which is necessary because the total transaction costs spent for rebalancing plus the value of the portfolio may not exactly match the investment budget. In the MILP formulations proposed by Guastaroba and Speranza (2012) and Filippi et al. (2016), implicit cash holdings can occur because the modeled transaction costs correspond merely to an upper bound on the true transaction costs. A drawback of these implicit cash holdings is that they are not considered in the calculation of the historical portfolio values and thus are also ignored in the dissimilarity and excess return functions.

The existing solution approaches presented in the literature may not be appropriate for the EITP. The existing exact approaches and the Kernel Search heuristic would first require the solution of the continuous relaxation of the MIQP formulation of the EITP, which is a quadratic program that becomes computationally very expensive to solve when large indices are considered. The existing metaheuristics would require adaptation to the real-life constraints of the EITP, which may reduce their effectiveness because they are tailored for other specific problems that do not include all of the real-life constraints of the EITP. A further drawback of metaheuristics is that they may investigate many infeasible solutions and thus be ineffective.

Table 3.1: Problems related to the EITP considered in the literature.

	Paper	Real-life constraints				Objective	
		Cardinality	Min./max. weights	Transaction costs	Min./max. trades	IT	EIT
Non-linear objective function	Jorion (2003)					✓	✓
	Roll (1992)					✓	✓
	Rudd (1980)		✓			✓	✓
	Konno and Wijayanayake (2001)			✓		✓	
	Chiam et al. (2013)			✓		✓	
	Gaivoronski et al. (2005)	✓		✓		✓	
	Takeda et al. (2013)	✓				✓	
	Kwiatkowski (1992)	✓				✓	
	Maringer and Oyewumi (2007)	✓				✓	
	Mutunge and Haugland (2018)	✓				✓	
	Sant'Anna et al. (2017a)	✓				✓	✓
	Sant'Anna et al. (2017b)	✓				✓	
	Andriosopoulos and Nomikos (2014)	✓	✓			✓	✓
	Jansen and Van Dijk (2002)	✓	✓			✓	
	Scozzari et al. (2013)	✓	✓			✓	
	Krink et al. (2009)	✓	✓			✓	
	Beasley et al. (2003)	✓	✓	✓		✓	
Linear objective function	Rudolf et al. (1999)					✓	
	Bruni et al. (2015)	✓	✓			✓	✓
	Guastaroba et al. (2016)	✓	✓			✓	✓
	Guastaroba and Speranza (2012)	✓	✓	✓		✓	✓
	Filippi et al. (2016)	✓	✓	✓		✓	✓
	Strub and Baumann (2018)	✓	✓	✓	✓	✓	

3.3 Mixed-integer linear and quadratic programming formulations

In this section, we present the novel MIQP formulation and the novel benchmark MILP formulations for determining enhanced index-tracking portfolios. In Subsection 3.3.1, we first present the objective functions and the constraints on the expected excess return that are used in the three mixed-integer programming (MIP) formulations. In Subsection 3.3.2, we present new arguments that using the expected tracking error instead of a dissimilarity function as the objective function may lead to superior portfolios in terms of the out-of-sample tracking error and out-of-sample excess return when large indices are considered. In Subsection 3.3.3, we introduce the formulation of the real-life constraints. In Subsection 3.3.4, we strengthen the formulation of the real-life constraints and present the complete MIP formulations.

Table 3.1 shows the nomenclature used in the MIP formulations. The set of available assets consists of the set of index constituents $U = \{1, \ldots, n\}$ and an asset $n + 1$ that represents the explicitly modeled cash holdings. Note that in Table 3.1, the decision variables are defined only for a subset $I \subseteq U$ of the index constituents. We define the MIP formulations in this general form for the set of considered stocks I because we can then use the same formulations with only minor modifications for the heuristic solution approaches presented in Section 3.4. Note that the set I must contain all stocks that are included in the portfolio before rebalancing and cannot be sold off completely because of the minimum and maximum trading values.

3.3.1 Objective functions and constraints on the expected excess return

The two competing objectives in enhanced index tracking are the minimization of the expected tracking error and the maximization of the expected excess return. In this subsection, we present the functions used to model these objectives in the proposed MIP formulations. For the MIQP formulation, we use the functions presented by Roll (1992), and for the two MILP formulations, we use the functions presented by Guastaroba and Speranza (2012) and Filippi et al. (2016). We adjust all functions to account for the set of considered stocks I and the explicitly modeled cash holdings.

The expected tracking error, which is used in the MIQP formulation, is a function of the expected covariances σ_{ij} between the returns of assets $i \in U \cup \{n + 1\}$ and $j \in U \cup \{n + 1\}$, the weights $\frac{P_{iT} X_i}{C}$ of the assets $i \in U \cup \{n + 1\}$ in the portfolio, and the

Table 3.1: Nomenclature for the MIP formulations.

Sets and parameters:	
T	Point in time at which the EITP must be solved (today)
n	Number of stocks in the index
U	Set of index constituents ($U = \{1, \dots, n\}$)
I_s	Set of stocks that must be included in the portfolio after rebalancing due to the minimum and maximum trading values ($I_s = \{i \in U : 0 < P_{iT}Y_i < \zeta_i C \vee P_{iT}Y_i > \eta_i C\}$)
I	Set of considered stocks ($I_s \subseteq I \subseteq U$)
k	Maximum portfolio cardinality
κ	Net cash flow from deposits and withdrawals
i_t	Continuously compounded interest rate on cash for the period starting at $t-1$ and ending at t, $t \in \{2, \dots, T\}$
I_t/P_{it}	Historical value/price of index/stock $i \in U$ at $t \in \{1, \dots, T\}$
$P_{n+1,t}$	Historical value of the asset that represents cash, with $P_{n+1,T} = 100$ and $P_{n+1,t} = P_{n+1,T} \exp(-\sum_{s=t}^{T-1} i_s)$, $t \in \{1, \dots, T-1\}$
Y_i	Number of units of asset $i \in U \cup \{n+1\}$ in the portfolio before rebalancing
C	Investment budget ($C = \kappa + \sum_{i \in U \cup \{n+1\}} Y_i P_{iT}$)
ζ_i/η_i	Minimum/maximum trading value of stock $i \in U$ if traded, expressed as a percentage of C
ε_i/δ_i	Minimum/maximum weight of stock $i \in U$ if included in the portfolio after rebalancing
c_i^f	Fixed transaction cost for trading stock $i \in U$
c_i^b/c_i^s	Proportional transaction cost for buying/selling stock $i \in U$ as a percentage of the trading value
γ	Maximum total transaction costs, expressed as a percentage of C
w_i^I	Weight of asset $i \in U \cup \{n+1\}$ in the index, with $w_{n+1}^I = 0$
$\overline{r}_i$	Expected return of asset $i \in U \cup \{n+1\}$ ($\overline{r}_i = \frac{1}{T-1} \sum_{t \in \{2,\dots,T\}} \frac{P_{it} - P_{i,t-1}}{P_{i,t-1}}$)
α	Prescribed minimum expected excess return
σ_{ij}	Expected covariance between the discrete returns of asset $i \in U \cup \{n+1\}$ and asset $j \in U \cup \{n+1\}$ calculated by using the estimator of Ledoit and Wolf (2004b)
Continuous non-negative decision variables:	
X_i	Number of units of asset $i \in I \cup \{n+1\}$ in the portfolio after rebalancing
G_i	Total transaction costs associated with stock $i \in I$
v_i^b/v_i^s	Value bought/sold of stock $i \in I$
Binary decision variables:	
z_i	$= 1$, if $X_i > 0$; $= 0$, otherwise ($i \in I$)
z_i^b	$= 1$, if $X_i > Y_i$; $= 0$, otherwise ($i \in I$)
z_i^s	$= 1$, if $X_i < Y_i$; $= 0$, otherwise ($i \in I$)

weights w_i^I of the assets $i \in U \cup \{n+1\}$ in the index, with $w_{n+1}^I = 0$. Any stock that is not included in I will have a portfolio weight of zero. Thus, based on the set of decision variables X_i for $i \in I \cup \{n+1\}$, the following function represents the expected tracking error:

$$\sum_{i,j \in I \cup \{n+1\}} \sigma_{ij} \left(\frac{P_{iT} X_i}{C} - w_i^I \right) \left(\frac{P_{jT} X_j}{C} - w_j^I \right) - \\ 2 \sum_{i \in I \cup \{n+1\}} \sum_{j \in U \setminus I} \sigma_{ij} \left(\frac{P_{iT} X_i}{C} w_j^I - w_i^I w_j^I \right) + \sum_{i,j \in U \setminus I} \sigma_{ij} w_i^I w_j^I \tag{3.1}$$

In this paper, we use the estimator of Ledoit and Wolf (2004b) for the expected covariances. As Ledoit and Wolf (2004a) note, using this estimator ensures that the matrix of the expected covariances is positive definite, and thus, the expected tracking error is a convex quadratic function of the weights of the stocks in the portfolio, which allows commercial MIP solvers such as CPLEX and Gurobi to be applied.

The expected excess return is the difference between the expected return of the portfolio and the expected return of the index:

$$\sum_{i \in I \cup \{n+1\}} \frac{P_{iT} X_i}{C} \overline{r}_i - \sum_{i \in U \cup \{n+1\}} w_i^I \overline{r}_i \tag{3.2}$$

In the MIQP formulation, we minimize the expected tracking error subject to a constraint that prescribes a minimum expected excess return of α, as follows:

$$\begin{cases} \text{Min. } (3.1) & (3.3) \\ \text{s.t.} \quad \sum_{i \in I \cup \{n+1\}} \frac{P_{iT} X_i}{C} \overline{r}_i - \sum_{i \in U \cup \{n+1\}} w_i^I \overline{r}_i \geq \alpha & (3.4) \end{cases}$$

The first MILP formulation is based on a tracking target whose historical in-sample values are equal to the in-sample values of the index scaled to the investment budget at time point T and multiplied by $(1 + \alpha^*)$, where α^* is a decision variable that represents the portfolio's expected excess return. The dissimilarity function then captures the MAD between the values $\sum_{i \in I \cup \{n+1\}} P_{it} X_i$ of the portfolio and the values $(1 + \alpha^*) I_t \frac{C}{I_T}$ of the tracking target over all in-sample time points $t \in \{1, \ldots, T\}$. With the introduction of the non-negative decision variables u_t and d_t for $t \in \{1, \ldots, T\}$, the MAD can be minimized subject to the constraint on the expected excess return as follows:

$$\begin{cases} \text{Min. } \dfrac{1}{T} \displaystyle\sum_{t \in \{1,\dots,T\}} (u_t + d_t) & (3.5) \\ \text{s.t. } \quad u_t - d_t = \displaystyle\sum_{i \in I \cup \{n+1\}} P_{it} X_i - (1+\alpha^*) I_t \frac{C}{I_T} \qquad (t \in \{1,\dots,T\}) & (3.6) \\ \qquad \alpha^* \geq \alpha & (3.7) \end{cases}$$

In the second MILP formulation, the dissimilarity function captures the MAD between the in-sample values of the scaled index and the portfolio. For the expected excess return, the average excess return over the entire in-sample period is used. By again using the non-negative decision variables u_t and d_t, the MAD can be minimized subject to the constraint on the expected excess return as follows:

$$\begin{cases} \text{Min. } \dfrac{1}{T} \displaystyle\sum_{t \in \{1,\dots,T\}} (u_t + d_t) & (3.8) \\ \text{s.t. } \quad u_t - d_t = \displaystyle\sum_{i \in I \cup \{n+1\}} P_{it} X_i - I_t \frac{C}{I_T} \qquad (t \in \{1,\dots,T\}) & (3.9) \\ \qquad \dfrac{1}{(T-1)} \displaystyle\sum_{t \in \{2,\dots,T\}} \left(\sum_{i \in I \cup \{n+1\}} \frac{P_{iT} X_i}{C} \left(\frac{P_{it}}{P_{i,t-1}} - 1 \right) - \left(\frac{I_t}{I_{t-1}} - 1 \right) \right) \geq \alpha & (3.10) \end{cases}$$

3.3.2 Expected tracking error: a comparison with dissimilarity functions

In this subsection, we compare the expected tracking error with the two dissimilarity functions presented in Subsection 3.3.1. For this purpose, we consider all index constituents, i.e., $I = U$, and we consider only a budget constraint that ensures that the entire investment budget is invested in the assets. We assume that the number of available assets is much larger than the number of in-sample time points, i.e., $|U \cup \{n+1\}| \gg T$, and that the matrix consisting of the in-sample prices of each stock, where each stock corresponds to a column, has full row rank. Both assumptions are usually satisfied when the index is large. We further assume that the matrix of the expected covariances is positive definite. This assumption is always satisfied when an appropriate estimator is used for the expected covariances, such as that of Ledoit and Wolf (2004b).

When the expected tracking error is to be minimized with a positive-definite matrix of expected covariances, the only solution with zero expected tracking error is the portfolio that has the same composition as the index. To see this, let $\mathbf{x}$ be the column vector of the differences between the weights of the stocks in the portfolio and the weights of the stocks in the index, and let $\mathbf{A}$ be the matrix of the expected covariances. Then, the expected tracking error is computed as $\mathbf{x}^\top \mathbf{A}\mathbf{x}$, which is always positive for $\mathbf{x} \neq \mathbf{0}$, where $\mathbf{0}$ is the zero vector with the appropriate dimensionality. Hence, the only portfolio with zero tracking error is $\mathbf{x} = \mathbf{0}$, which is the portfolio whose composition matches the index composition.

By contrast, when a dissimilarity function is used, i.e., when the known index composition is ignored, infinitely many different portfolios can exist that achieve a dissimilarity of zero with respect to the index. To see this, note that finding a portfolio with zero dissimilarity is equivalent to solving a system of T linear equations with $n + 1$ unknowns, where these T equations state that the portfolio value at each time point $t \in \{1, \ldots, T\}$ must match the scaled index value at that time point. Note that the equation for time point T also ensures that the budget constraint is satisfied because of the scaling of the index values. Under the assumption of a full row rank matrix of stock prices, infinitely many solutions to this linear system exist, which means that infinitely many portfolios with zero dissimilarity exist.

Based on the arguments above, one drawback of minimizing a dissimilarity function is that, in contrast to the case of directly minimizing the expected tracking error, many different portfolios can exist that each have an objective function value of zero but a composition that strongly differs from that of the index. These portfolios may have very high out-of-sample tracking errors and very low out-of-sample excess returns. Hence, for our computational experiment reported in Section 3.5, we expect that over all considered problem instances, the worst-case tracking error and the worst-case excess return for the out-of-sample period will be much worse when a dissimilarity function is minimized instead of the expected tracking error.

3.3.3 Real-life constraints

Next, we model the real-life constraints. The constraints expressed in (3.11) assign at least the absolute value bought or sold of each stock $i \in I$ to the non-negative decision variable v_i^b or v_i^s, respectively. These decision variables are used to model the transaction costs and the minimum and maximum trading values.

$$v_i^b - v_i^s = P_{iT}(X_i - Y_i) \qquad (i \in I) \tag{3.11}$$

The purpose of constraints (3.12) and (3.13) is twofold. First, the binary variables z_i^b and z_i^s are assigned a value of one if the variables v_i^b and v_i^s, respectively, take a positive value and a value of zero otherwise. Second, the constraints prescribe minimum and maximum values of $\zeta_i C$ and $\eta_i C$, respectively, for v_i^b and v_i^s.

$$\zeta_i C z_i^b \leq v_i^b \leq \eta_i C z_i^b \qquad (i \in I) \tag{3.12}$$

$$\zeta_i C z_i^s \leq v_i^s \leq \eta_i C z_i^s \qquad (i \in I) \tag{3.13}$$

The constraints defined in (3.14) ensure that for each stock $i \in I$, at most one of the binary variables z_i^b and z_i^s can be set to one.

$$z_i^b + z_i^s \leq 1 \qquad (i \in I) \tag{3.14}$$

Together, constraints (3.12), (3.13), and (3.14) ensure that for each stock $i \in I$, either v_i^b or v_i^s must be set to zero. Because it is not possible for both variables v_i^b and v_i^s to take positive values simultaneously for a given stock $i \in I$, the constraints defined in (3.11) assign the actual values bought or sold of each stock $i \in I$ to the variables v_i^b or v_i^s, respectively. These actual values are necessary to model the minimum and maximum trading values $\zeta_i C$ and $\eta_i C$ using constraints (3.12) and (3.13).

Based on the variables v_i^b, v_i^s, z_i^b, and z_i^s, the transaction costs G_i for each stock $i \in I$ are calculated using the constraints defined in (3.15). Note that the variables G_i take values equal to the actual transaction costs associated with each stock $i \in I$, because we ensure that the variables v_i^b and v_i^s take the actual values bought and sold of each stock, and that at most one of the binary variables z_i^b and z_i^s can be set to one if stock $i \in I$ is traded, whereas both variables z_i^b and z_i^s are set to zero otherwise.

$$G_i = c_i^b v_i^b + c_i^s v_i^s + c_i^f (z_i^b + z_i^s) \qquad (i \in I) \tag{3.15}$$

To model the budget constraint (3.16), we introduce an additional cash asset $n+1$, with $X_{n+1} \geq 0$. The budget constraint then states that the available investment budget C must be either held in cash, invested in the stocks that constitute the index, or spent for transaction costs. Note the possibility that some stocks were included in the portfolio before rebalancing but are not included in the set of considered stocks I. Hence, the shares of these stocks must be sold, incurring total transaction costs of $\sum_{i \in U \setminus I : Y_i > 0} \left(c_i^s Y_i P_{iT} + c_i^f \right)$. Since the variables G_i take values equal to the actual transaction costs associated with each stock $i \in I$, constraint (3.16) ensures that the variable X_{n+1} corresponds exactly to the part of the investment budget that is not invested in stocks or spent for transaction costs. Hence, we can explicitly account for these cash holdings when formulating the expected tracking error and the expected excess return.

$$\sum_{i \in I \cup \{n+1\}} P_{iT} X_i + \sum_{i \in I} G_i + \sum_{i \in U \setminus I : Y_i > 0} \left(c_i^s Y_i P_{iT} + c_i^f \right) = C \tag{3.16}$$

Constraint (3.17) prescribes a budget of γC for the total transaction costs.

$$\sum_{i \in I} G_i + \sum_{i \in U \setminus I : Y_i > 0} \left(c_i^s Y_i P_{iT} + c_i^f \right) \leq \gamma C \tag{3.17}$$

The constraints (3.18) ensure that each binary variable z_i takes a value of one if stock $i \in I$ is included in the portfolio after rebalancing and a value of zero otherwise. Furthermore, these constraints define minimum and maximum values of ε_i and δ_i, respectively, for the weight of each stock $i \in I$ in the portfolio.

$$\varepsilon_i z_i \leq \frac{P_{iT} X_i}{C} \leq \delta_i z_i \qquad (i \in I) \tag{3.18}$$

Based on the binary variables z_i, the cardinality constraint (3.19) is formulated as follows.

$$\sum_{i \in I} z_i \leq k \tag{3.19}$$

The domains of the decision variables are specified by (3.20) and (3.21).

$$X_i \geq 0 \qquad (i \in I \cup \{n+1\}) \tag{3.20}$$

$$z_i^b, z_i^s, z_i \in \{0, 1\}; \; v_i^b, v_i^s, G_i \geq 0 \qquad (i \in I) \tag{3.21}$$

3.3.4 Strengthened mixed-integer programming formulations

The formulation of the real-life constraints presented in Subsection 3.3.3 can be strengthened by removing some redundant constraints and variables. This strengthening is based on the following three insights. We note that some stocks must always be included in the portfolio because selling all units of these stocks is impossible due to the specified minimum or maximum trading values. To simplify the notation, we introduce the set $I_s \subseteq I$, which comprises all these stocks that are included in the portfolio before rebalancing and cannot be sold off completely (i.e., $I_s = \{i \in U : 0 < P_{iT} Y_i < \zeta_i C \vee P_{iT} Y_i > \eta_i C\}$). Filippi et al. (2016) note that if a stock is not included in the portfolio before rebalancing and is traded, then this stock will always be included in the portfolio after rebalancing. Strub and Baumann (2018) note that stocks that are not included in the portfolio before rebalancing cannot be sold because short selling is not allowed. By combining all three insights, we obtain the following restrictions on stocks based on their values in the portfolio before rebalancing. For each stock i that is not included in the portfolio before rebalancing, i.e., $Y_i = 0$, selling stock i is not possible, and trading stock i means that it will be included in the portfolio after rebalancing. For each stock i that has a value in the portfolio before rebalancing that is positive but smaller than the minimum trading value, i.e., $0 < P_{iT}Y_i < \zeta_i C$, selling stock i is not possible, and thus, stock i must be included in the portfolio after rebalancing. For each stock i that has a value in the portfolio before rebalancing that is larger than the maximum trading value, i.e., $\eta_i C < P_{iT}Y_i$, selling all units of stock i is not possible, and thus, stock i must be included in the portfolio after rebalancing.

Based on the restrictions above, we can eliminate certain variables. For each stock i that cannot be sold, the binary variable z_i^s and the continuous variable v_i^s must both be zero and thus can be removed. Additionally, the continuous variable v_i^b can be replaced with $P_{iT}(X_i - Y_i)$. Furthermore, for each stock i that is not included in the portfolio before rebalancing, i.e., $Y_i = 0$, we can replace the binary variable z_i^b with the binary variable z_i because selling is not possible and buying stock i means that it will be included in the portfolio after rebalancing. For each stock i that must be included in the portfolio after rebalancing, we can set the binary variable z_i equal to one.

The novel MIQP formulations (M-Q) and the two novel benchmark MILP formulations (M-L1) and (M-L2) below include the strengthened versions of the real-life constraints (without redundant variables and constraints) along with their different objective functions and constraints on the expected excess return.

$$
(\text{M-Q})\left\{
\begin{array}{lll}
\text{Min.} & (3.1) & \\
\text{s.t.} & (3.4), (3.16), (3.17) & \\
& v_i^b - v_i^s = P_{iT}(X_i - Y_i) & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.22) \\
& \zeta_i C z_i \leq X_i P_{iT} \leq \eta_i C z_i & (i \in I : Y_i = 0) \quad (3.23) \\
& \zeta_i C z_i^b \leq (X_i - Y_i) P_{iT} \leq \eta_i C z_i^b & (i \in I : 0 < P_{iT} Y_i < \zeta_i C) \quad (3.24) \\
& \zeta_i C z_i^b \leq v_i^b \leq \eta_i C z_i^b & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.25) \\
& \zeta_i C z_i^s \leq v_i^s \leq \eta_i C z_i^s & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.26) \\
& z_i^b + z_i^s \leq 1 & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.27) \\
& G_i = c_i^b P_{iT} X_i + c_i^f z_i & (i \in I : Y_i = 0) \quad (3.28) \\
& G_i = c_i^b P_{iT}(X_i - Y_i) + c_i^f z_i^b & (i \in I : 0 < P_{iT} Y_i < \zeta_i C) \quad (3.29) \\
& G_i = c_i^b v_i^b + c_i^s v_i^s + c_i^f (z_i^b + z_i^s) & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.30) \\
& \varepsilon_i \leq \dfrac{P_{iT} X_i}{C} \leq \delta_i & (i \in I_s) \quad (3.31) \\
& \varepsilon_i z_i \leq \dfrac{P_{iT} X_i}{C} \leq \delta_i z_i & (i \in I \setminus I_s) \quad (3.32) \\
& \displaystyle\sum_{i \in I \setminus I_s} z_i \leq k - |I_s| & (3.33) \\
& X_i \geq 0 & (i \in I \cup \{n+1\}) \quad (3.34) \\
& G_i \geq 0 & (i \in I) \quad (3.35) \\
& z_i \in \{0, 1\} & (i \in I \setminus I_s) \quad (3.36) \\
& v_i^b, v_i^s \geq 0, \;\; z_i^s \in \{0, 1\} & (i \in I : P_{iT} Y_i \geq \zeta_i C) \quad (3.37) \\
& z_i^b \in \{0, 1\} & (i \in I : Y_i > 0) \quad (3.38)
\end{array}
\right.
$$

$$
(\text{M-L1})\left\{
\begin{array}{lll}
\text{Min.} & (3.5) & \\
\text{s.t.} & (3.6), (3.7), (3.16), (3.17), (3.22), (3.23), & \\
& (3.24), (3.25), (3.26), (3.27), (3.28), (3.29), & \\
& (3.30), (3.31), (3.32), (3.33), (3.34), (3.35), & \\
& (3.36), (3.37), (3.38) & \\
& \alpha^* \in \mathbb{R} & (3.39) \\
& u_t, d_t \geq 0 & (t \in \{1, \ldots, T\}) \quad (3.40)
\end{array}
\right.
$$

$$
\text{(M-L2)} \begin{cases} \text{Min.} & (3.8) \\ \text{s.t.} & (3.9), (3.10), (3.16), (3.17), (3.22), (3.23), \\ & (3.24), (3.25), (3.26), (3.27), (3.28), (3.29), \\ & (3.30), (3.31), (3.32), (3.33), (3.34), (3.35), \\ & (3.36), (3.37), (3.38) \\ & u_t, d_t \geq 0 \qquad\qquad (t \in \{1, \ldots, T\}) \qquad (3.41) \end{cases}
$$

3.4 Heuristic solution approaches

In this section, we present the two novel matheuristics for the EITP. In Subsection 3.4.1, we present a construction heuristic for determining an initial feasible portfolio for the EITP. In Subsections 3.4.2 and 3.4.3, we present the two improvement heuristics based on local branching and the concept of iterated greedy heuristics, respectively, for improving this initial feasible portfolio. Table 3.1 defines the additional nomenclature used in formulating the two matheuristics.

3.4.1 Construction heuristic

Constructing a feasible portfolio is not straightforward. It is possible that no feasible portfolio exists, e.g., when the prescribed minimum excess return is set too high or when a current portfolio must be rebalanced so heavily to ensure the prescribed minimum and maximum weights that the prescribed budget for transaction costs is too low. Even if a feasible portfolio exists, when selecting the stocks that should be included in the portfolio

Table 3.1: Additional nomenclature for the heuristic solution approaches.

Parameters:	
q	Parameter that defines the number of considered stocks
Δ	Number of stocks that can be added, removed, or exchanged
$\overline{\nu}$	Maximum number of iterations without improvement
d	Maximum number of stocks to be removed
Continuous non-negative decision variables:	
u_i/d_i	Absolute upward/downward deviation between the portfolio weight and index weight of asset $i \in I$

by applying, e.g., a random or greedy algorithm, it might not be possible to find weights for these selected stocks such that the portfolio is feasible with respect to all constraints.

Therefore, we propose a MILP-based construction heuristic that is able to find a feasible portfolio quickly and can also prove the nonexistence of a feasible portfolio. To simplify the search for a feasible portfolio for the MIQP formulation (M-Q), we use the identity matrix as a simplified covariance matrix. Thus, the objective function (3.1) reduces to the following terms:

$$\sum_{i \in I \cup \{n+1\}} \left(\frac{P_{iT} X_i}{C} - w_i^I \right)^2 + \sum_{i \in U \setminus I} (w_i^I)^2 \tag{3.42}$$

Furthermore, we consider the sum of the absolute deviations (instead of the squared deviations) between the weights of the assets in the portfolio and the weights of the assets in the index, and we ignore the second sum because it is constant. The resulting objective function can be optimized subject to the constraints of the EITP using the following MILP formulation:

$$\text{(M-C)} \begin{cases} \text{Min.} \quad \displaystyle\sum_{i \in I \cup \{n+1\}} (u_i + d_i) & (3.43) \\ \text{s.t.} \quad u_i - d_i = \dfrac{P_{iT} X_i}{C} - w_i^I \qquad (i \in I \cup \{n+1\}) & (3.44) \\ \quad (3.4), (3.16), (3.17), (3.22), (3.23), (3.24), & \\ \quad (3.25), (3.26), (3.27), (3.28), (3.29), (3.30), & \\ \quad (3.31), (3.32), (3.33), (3.34), (3.35), (3.36), & \\ \quad (3.37), (3.38) & \\ \quad u_i, d_i \geq 0 \qquad (i \in I \cup \{n+1\}) & (3.45) \end{cases}$$

To further reduce the difficulty of finding a feasible portfolio, we consider only a limited set of promising stocks I. If no feasible portfolio can be found based on a given set I, we increase the cardinality of the set I. This preselection is crucial for finding good feasible portfolios for the MILP formulation (M-C). In general, the set I should contain the stocks with the highest weights in the index to allow a small objective function value to be achieved. Moreover, the set I should contain the stocks that are in the current portfolio, because not including these stocks in the set I would mean that we were required to sell all units of these stocks, which would incur high transaction costs.

Algorithm 3.1 describes the construction heuristic. First, we include in the set I all stocks that are held in the portfolio before rebalancing, and we initialize ν. Then, we gradually expand the set I by including the k stocks that have the highest weights in the index and are not yet included in the set I. Thereafter, based on the expanded set I, the MILP formulation (M-C) is solved. This process is repeated until a feasible portfolio is found or until $I = U$. If no feasible portfolio is found with $I = U$, this proves that no feasible portfolio exists.

Algorithm 3.1 Construction heuristic

procedure ConstructionHeuristic()
    $\nu \leftarrow 0$; $I \leftarrow \{i \in U : Y_i > 0\}$;
    while true **do**
        while $|I| < \min\{n,\ |\{i \in U : Y_i > 0\}| + k(1+\nu)\}$ **do**
            $I \leftarrow I \cup \{\min\{i \in U \setminus I : w_i^I = \max_{j \in U \setminus I} w_j^I\}\}$;
        end while
        Solve (M-C);
        if feasible portfolio found **then**
            return set of stocks included in the feasible portfolio;
        else if $I = U$ **then**
            return no feasible portfolio exists;
        end if
        $\nu \leftarrow \nu + 1$;
    end while
end procedure

3.4.2 Local branching heuristic

Local branching refers to a local-search framework for MIP formulations that is based on so-called local-branching cuts. Given a feasible solution, these local-branching cuts iteratively define the neighborhood to be searched by placing an upper bound on the number of binary variables whose values can be flipped, either from one to zero or from zero to one. Based on this framework, we extend the MIQP formulation (M-Q) by incorporating constraints (3.46) and (3.47). These constraints restrict the search space to all feasible portfolios that can be reached by starting from the best feasible portfolio found so far, which includes the stocks in the set I^*, and either adding, removing, or exchanging at most Δ stocks. When removing stocks, we ensure that no stocks from the set I_s are removed because these stocks must be included in the portfolio after rebalancing. This results in the MIQP formulation (M-LBH) shown below.

$$
\text{(M-LBH)} \begin{cases} \text{Min.} & (3.1) \\ \text{s.t.} & (3.4), (3.16), (3.17), (3.22), (3.23), (3.24), \\ & (3.25), (3.26), (3.27), (3.28), (3.29), (3.30), \\ & (3.31), (3.32), (3.33), (3.34), (3.35), (3.36), \\ & (3.37), (3.38) \\ & \sum_{i \in I^* \setminus I_s} (1 - z_i) \leq \Delta & (3.46) \\ & \sum_{i \in I \setminus I^*} z_i \leq \Delta & (3.47) \end{cases}
$$

The local branching framework requires the solution of a series of quadratic programs, which is computationally expensive for large indices when all available stocks are considered in each iteration, i.e., when $I = U$, even when the search space is restricted by local-branching cuts. Therefore, we propose a novel approach in which the search space is further restricted by considering only a limited set of promising stocks I. To prevent the exclusion of high-quality solutions from the search space due to a poor preselection of the stocks to be included in the set I, we use a randomly selected set I in each iteration. Because we consider stocks with higher index weights to be more promising for obtaining low objective function values, we define the probability that a stock will be included in the set I in each iteration to be proportional to its weight in the index.

Algorithm 3.2 describes the local branching heuristic. First, we determine an initial feasible portfolio by applying the construction heuristic, and we initialize ν and Δ. Then, we include in the set I the stocks from the set I^*, which contains the stocks that are included in the best feasible portfolio found so far. Subsequently, we iteratively include in the set I stocks until $|I| = \min\{|U|, k + q\}$. In this process, each stock $i \in U \setminus I$ has a probability $\frac{w_i^I}{\sum_{j \in U \setminus I} w_j^I}$ of being included in the set I. Thereafter, the MIQP formulation (M-LBH) is solved. If a better feasible portfolio is found, we update the set I^* to contain the selected stocks in this new best feasible portfolio, and we reset the number of iterations elapsed without finding a better feasible portfolio ν to zero; otherwise, we increase ν by one. If ν reaches $\overline{\nu}$, i.e., the maximum number of iterations without a better feasible portfolio having been found, we increase Δ by one to enlarge the search space. As soon as a new best feasible portfolio is found, we reset Δ to one. This process is repeated until a given termination criterion is satisfied. Finally, the best feasible portfolio found so far is returned.

Algorithm 3.2 Local branching heuristic

procedure LocalBranchingHeuristic(q, $\overline{\nu}$)
    $I^* \leftarrow$ ConstructionHeuristic();
    $\nu \leftarrow 0$; $\Delta \leftarrow 1$;
    while termination criterion not satisfied **do**
        $I \leftarrow I^*$;
        while $|I| < \min\{|U|, k+q\}$ **do**
            $a \leftarrow$ select stock from set $U \setminus I$ with probability $\frac{w_i^I}{\sum_{j \in U \setminus I} w_j^I}$ of the selection of stock $i \in U \setminus I$;
            $I \leftarrow I \cup \{a\}$;
        end while
        Solve (M-LBH) to obtain a feasible portfolio by adding, removing or exchanging at most Δ stocks;
        if new best feasible portfolio found **then**
            $I^* \leftarrow$ set of selected stocks in the new best feasible portfolio;
            $\nu \leftarrow 0$; $\Delta \leftarrow 1$;
        else
            $\nu \leftarrow \nu + 1$;
            if $\nu = \overline{\nu}$ **then**
                $\nu \leftarrow 0$; $\Delta \leftarrow \Delta + 1$;
            end if
        end if
    end while
    return best feasible portfolio found;
end procedure

3.4.3 Iterated greedy heuristic

In an iterated greedy heuristic, two phases are performed repeatedly: deconstruction and reconstruction. During the deconstruction phase, we remove several randomly selected stocks from the current best feasible portfolio. During the subsequent reconstruction phase, we add stocks back into the deconstructed portfolio in a greedy manner to obtain a new feasible portfolio.

In contrast to existing iterated greedy heuristics, we restrict the search space by considering only a limited set of promising stocks I, as in the local branching heuristic, and we repeatedly solve an MIQP formulation during the reconstruction phase to add the myopic best stock to the deconstructed portfolio, which allows all constraints in the MIQP formulation (M-Q) to be easily considered. Specifically, we solve the MIQP formulation (M-IGH) below that corresponds to the MIQP formulation (M-Q) without the cardinality constraint (3.33), but with the additional constraint (3.48). This additional constraint prescribes that at most one stock from the set I that is not included in the set I_s can be added to the portfolio. During the execution of the iterated greedy heuristic, we modify the set I_s such that it contains the stocks that must be included in the reconstructed portfolio, i.e., the stocks that are included in the current best feasible portfolio and were not removed during the most recent deconstruction phase.

$$
\text{(M-IGH)} \begin{cases} \text{Min.} & (3.1) \\ \text{s.t.} & (3.4), (3.16), (3.17), (3.22), (3.23), (3.24), \\ & (3.25), (3.26), (3.27), (3.28), (3.29), (3.30), \\ & (3.31), (3.32), (3.34), (3.35), (3.36), (3.37), \\ & (3.38) \\ & \displaystyle\sum_{i \in I \setminus I_s} z_i \leq 1 \qquad (3.48) \end{cases}
$$

Algorithm 3.3 describes the iterated greedy heuristic. First, we construct an initial feasible portfolio using the construction heuristic. Furthermore, we store the stocks that must be included in the portfolio after rebalancing in the set I'_s because the algorithm modifies the set I_s during its execution. Then, we enter the main loop, which consists of the deconstruction, reconstruction, and acceptance phases. Before beginning the deconstruction phase, we include in the set I the stocks that are included in the best feasible portfolio found so far. Then, during the deconstruction phase, we first remove p randomly selected stocks from the set I. When removing stocks, we must ensure that no stocks from the set I'_s are removed because these stocks must be included in the portfolio after rebalancing. After having removed p stocks from the set I, we define the set of stocks that must be included in the reconstructed portfolio, i.e., the set I_s, as the set of stocks currently in the set I. Then, the reconstruction phase begins. During the reconstruction phase, we expand the set I by adding stocks based on probabilities that depend on the weights of the stocks in the index, as is done in the local branching heuristic. As soon as the set I consists of $\min\{|U|, k + q\}$ stocks, we iteratively solve (M-IGH) to add to the portfolio at most one new stock in each iteration from the set $I \setminus I_s$. If a feasible portfolio with one new stock from the set $I \setminus I_s$ can be found, then the newly selected stock is added to the set I_s. This process is repeated until either k stocks are included in the portfolio, no new stock is added to the portfolio, or it is found that no feasible portfolio for (M-IGH) exists. After the reconstruction phase, we check whether a new best feasible portfolio has been found. If this is the case, we update the set I^* to contain the selected stocks in the new best feasible portfolio. The deconstruction, reconstruction, and acceptance phases are repeated until a specified termination criterion is met. Finally, we reset the set I_s and return the best feasible portfolio found so far.

3.5 Computational results

In this section, we report the results of our computational experiment. We tested the performance of the heuristic solution approaches introduced in Section 3.4 against the performance of three exact solution approaches based on the MIP formulations introduced in Section 3.3 for three scenarios that differ in terms of the composition of the portfolio before rebalancing. For the exact solution approaches, the MIQP formulation (M-Q) and the MILP formulations (M-L1) and (M-L2) were implemented in C, and Gurobi 7.5 was used as the solver; we refer to these MIP approaches as M-Q, M-L1, and M-L2, respectively. In the MIP approaches, we considered the entire set of index constituents, i.e., $I = U$. For the heuristic solution approaches, Algorithms 3.2 and 3.3 were also implemented in C; we refer to these approaches as LBH and IGH, respectively. The MILP formulation (M-C) and the MIQP formulations (M-LBH) and (M-IGH) that are used in Algorithms 3.2 and 3.3 were also implemented in C, and Gurobi 7.5 was again used as the solver.

This section is organized as follows. In Subsection 3.5.1, we explain the design of our experiment. In Subsection 3.5.2, we describe the novel problem instances. In Subsection 3.5.3, we present in-sample and out-of-sample results for the considered scenarios under the assumption that no rebalancing is performed. These results provide a ref-

Algorithm 3.3 Iterated greedy heuristic

procedure ITERATEDGREEDYHEURISTIC(q, d)
    $I^* \leftarrow$ CONSTRUCTIONHEURISTIC();
    $I'_s \leftarrow I_s$;
    while termination criterion not satisfied **do**
        $I \leftarrow I^*$;
        $p \leftarrow$ random integer from set $\{1, \ldots, d\}$; ▷ Deconstruction
        for $i \leftarrow 1$ **to** p **do**
            Remove a randomly selected element from I that is not included in I'_s;
        end for
        $I_s \leftarrow I$;
        while $|I| < \min\{|U|, k + q\}$ **do** ▷ Reconstruction
            $a \leftarrow$ select stock from set $U \setminus I$ with probability $\frac{w_i^I}{\sum_{j \in U \setminus I} w_j^I}$ of the selection of stock $i \in U \setminus I$;
            $I \leftarrow I \cup \{a\}$;
        end while
        while $|I_s| < k$ **do**
            Solve (M-IGH) to add at most one new stock to the portfolio;
            if a feasible portfolio with one new stock has been found **then**
                $I_s \leftarrow$ set of selected stocks in the feasible portfolio;
            else
                break;
            end if
        end while
        if new best feasible portfolio found **then** ▷ Acceptance
            $I^* \leftarrow$ set of selected stocks in the new best feasible portfolio;
        end if
    end while
    $I_s \leftarrow I'_s$
    return best feasible portfolio found;
end procedure

erence against which to assess the results presented in the following subsections. In Subsection 3.5.4 and Subsection 3.5.5, we present in-sample and out-of-sample results, respectively, for the portfolios after rebalancing using the tested solution approaches. Finally, in Subsection 3.5.6, we offer further insights with respect to the compositions of the rebalanced portfolios.

3.5.1 Experimental design

We used an experimental design similar to those of Guastaroba and Speranza (2012) and Filippi et al. (2016). We assumed that the manager of an investment fund rebalances a portfolio at the end of an in-sample period that consists of 104 weeks, i.e., $T = 104$. The portfolio is then left unchanged for the entirety of an out-of-sample period that consists of 52 weeks. We defined three scenarios, I, II, and III, which differ in terms of the composition of the investment fund's portfolio before rebalancing. In scenarios I and II, a portfolio of stocks already exists. In scenario III, a new portfolio must be constructed from cash. In scenarios I and II, the portfolios before rebalancing consist of the k stocks with the highest and lowest weights in the index, respectively, and each portfolio has a value of 10,000,000. The weight of each stock in the portfolio before rebalancing is set such that it is proportional to the weight of that stock in the index and such that the sum of the weights of all stocks in the portfolio is equal to one. The portfolio before rebalancing in scenario I is a portfolio with a rather good index-tracking capability, whereas the portfolio before rebalancing in scenario II is a portfolio with a rather poor index-tracking capability. This claim is supported by the results presented in Subsection 3.5.3. For scenarios I and II, a net change in cash of $\kappa = 0$ is assumed, and for scenario III, κ is assumed to be 10,000,000. Hence, in all three scenarios, the investment budget C is 10,000,000. We also assume that the fixed transaction cost for trading is 12 for all stocks (i.e., $c_i^f = 12$ for all $i \in U$), that the proportional transaction costs for buying and selling are each 1% of the trading value for all stocks (i.e., $c_i^b = c_i^s = 0.01$ for all $i \in U$), and that the budget available for transaction costs is 1.5% of the investment budget (i.e., $\gamma = 0.015$). The values of the parameters n, I_t, P_{it}, w_i^I, σ_{ij}, and $\overline{r}_i$ depend on the problem instance.

The values of the remaining parameters deviate from the values used by Guastaroba and Speranza (2012) and Filippi et al. (2016). The reason for these deviations is that we are considering much larger indices, and thus, we wish to allow the portfolio to have a larger cardinality. Because of the larger portfolio cardinality, we must also allow the portfolio weights to take smaller values. Specifically, we used two different values for the maximum portfolio cardinality, namely, $k = 100$ and $k = 200$. For the minimum and maximum portfolio weights, we adopted values of 0.2% and 20%, respectively, for all stocks (i.e., $\varepsilon_i = 0.002$ and $\delta_i = 0.2$ for all $i \in U$). For the parameters that define the minimum and maximum trading values, we used the same values as for the parameters ε_i and δ_i for all stocks (i.e., $\zeta_i = 0.002$ and $\eta_i = 0.2$ for all $i \in U$).

Since the prescribed minimum expected excess return α and the interest rate i_t on cash were not used by Guastaroba and Speranza (2012) and Filippi et al. (2016), we defined new values. Specifically, we used three α values of 0, 0.0001914, and 0.0005686, which correspond to annualized α values of 0%, 1%, and 2%, respectively, and a value of zero for the interest rate on cash at all time points (i.e., $i_t = 0$ for all $t \in \{2, \dots, T\}$).

After preliminary experiments, we adopted the following values for the input parameters of the heuristic solution approaches for all problem instances. For LBH, we set the number of stocks considered to $k + 50$ and the maximum number of iterations without improvement to ten (i.e., $q = 50$ and $\overline{\nu} = 10$). For IGH, we adopted the same number of stocks considered as for LBH and a value of two as the maximum number of stocks to be removed from the best feasible portfolio found so far (i.e., $q = 50$ and $d = 2$).

All calculations were performed on an HP Z820 workstation with two 3.1 GHz Intel Xeon CPUs and 128 GB of RAM. As the termination criterion for all solution approaches, we defined a computational time limit of one minute. For the MILP formulation (M-C) that is solved in Algorithm 3.1, we stopped the Gurobi solver as soon as the MIP gap reached a value of 10% or lower.

3.5.2 Novel problem instances

To the best of our knowledge, there is no set of instances available in the literature for the EITP. In the existing sets of problem instances (cf., e.g., Beasley et al., 2003; Canakgoz and Beasley, 2008; Guastaroba et al., 2009; Strub and Baumann, 2018), the weights of the stocks in the index at the time of rebalancing, i.e., at the end of week T, are not provided. Furthermore, no available set of problem instances contains very large regional and global stock-market indices. The largest existing problem instance that corresponds to the Russell 3000 index consists of fewer than 2,500 US stocks. Hence, we here provide a set of 18 novel instances of the EITP. These instances are based on real-world data from nine different stock-market indices maintained by Thomson Reuters (TR) for two different time periods. Table 3.1 lists, for each instance, the name of the instance, the name of the index, the number of stocks n in the index, and the considered time period. For each problem instance, we used DATASTREAM to download 156 weekly values of the index and 156 weekly closing prices of the constituents of the index during the corresponding time period. We consider the constituents of the index at the end of the in-sample period, i.e., at the end of week 104. As done by Beasley et al. (2003), Canakgoz and Beasley

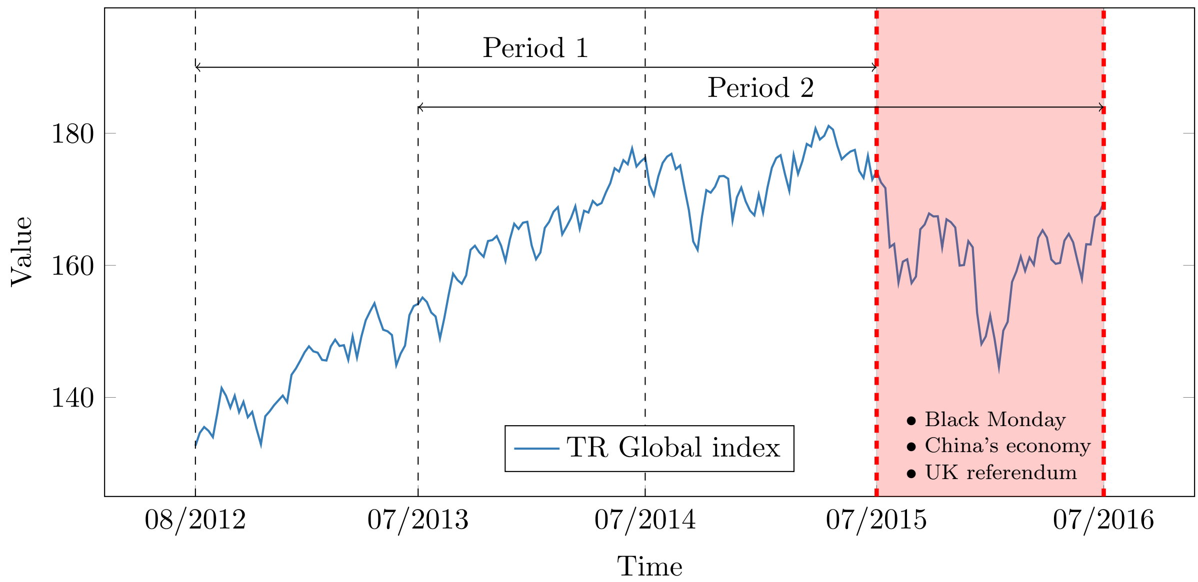
Table 3.1: Problem instances.

	Instance	Index	n	Time period
Period 1	tr1	TR Africa	168	08/2012–07/2015
	tr2	TR Latin America	194	08/2012–07/2015
	tr3	TR Europe	1,310	08/2012–07/2015
	tr4	TR United States	1,592	08/2012–07/2015
	tr5	TR North America	1,866	08/2012–07/2015
	tr6	TR Global Emerging Markets	2,912	08/2012–07/2015
	tr7	TR Asia Pacific	5,018	08/2012–07/2015
	tr8	TR Global Developed Markets	5,965	08/2012–07/2015
	tr9	TR Global	8,877	08/2012–07/2015
Period 2	tr10	TR Africa	168	08/2013–07/2016
	tr11	TR Latin America	246	08/2013–07/2016
	tr12	TR Europe	1,504	08/2013–07/2016
	tr13	TR United States	2,222	08/2013–07/2016
	tr14	TR Global Emerging Markets	2,532	08/2013–07/2016
	tr15	TR North America	2,620	08/2013–07/2016
	tr16	TR Asia Pacific	4,663	08/2013–07/2016
	tr17	TR Global Developed Markets	6,896	08/2013–07/2016
	tr18	TR Global	9,427	08/2013–07/2016

(2008), and Strub and Baumann (2018), we disregard the constituents for which the price data for the considered 156 weeks are incomplete; thus, the number of stocks n in the index can differ between the two different time periods. We also provide the weight of each constituent in the index at the end of the in-sample period. For all 18 problem instances, the sum of the original weights of the stocks with complete price data is at least 95%. The weights of these index constituents are then scaled for each instance such that their sum is equal to one.

Figure 3.1 shows the evolution of the value of the TR Global index over the two overlapping periods from August 2012 to July 2015 (instance tr9) and from August 2013 to July 2016 (instance tr18). In the figure, both periods are split into three equal parts consisting of 52 weeks each. The first two parts of each period correspond to the in-sample period, and the third part corresponds to the out-of-sample period. As Figure 3.1 shows, the Black Monday market crash, a drastic downward revision of the growth expectations for China's economy, and the UK referendum regarding the European Union all led to high market volatility during the time frame marked in red. These events did not affect the out-of-sample period of period 1, i.e., instances tr1–tr9, but strongly impacted the out-of-sample period of period 2, i.e., instances tr10–tr18. Hence, the differentiation between problem instances tr1–tr9 and tr10–18 enables investigation of the performance of portfolios during out-of-sample periods characterized by both low and high market volatility.

Figure 3.1: Considered time periods.



3.5.3 Portfolios without rebalancing: in-sample and out-of-sample performance analysis

For comparative purposes, we present in-sample and out-of-sample results for scenarios I, II, and III under the assumption that the given portfolio is not rebalanced by the investment fund's manager at the end of the in-sample period and thus remains unchanged for the out-of-sample period. This means that in scenarios I and II, the portfolios still consist of the k stocks with the highest and lowest index weights, respectively, and in scenario III, the portfolio still consists only of cash. Note that these portfolios are not necessarily feasible portfolios; for example, the constraint regarding the prescribed minimum excess return or the constraints regarding the minimum and maximum portfolio weights might be violated.

Table 3.2 summarizes the in-sample and out-of-sample results for these portfolios for period 1 and period 2. Columns one and two indicate the considered scenario and the maximum portfolio cardinality k, respectively. Column three shows the number of problem instances for period 1 for which the corresponding value of k is applicable, i.e., the number of problem instances with $n \geq k$. Columns four, five, and six show the average objective function value (i.e., the average expected tracking error), the average out-of-sample tracking error, and the average out-of-sample excess return, respectively, for period 1. Columns seven to ten report the corresponding results for period 2.

From Table 3.2, we can gain the following insights:

Table 3.2: In-sample and out-of-sample results (averages over the instances with $n \geq k$ among instances tr1–tr9 (period 1) and instances tr10–tr18 (period 2)) for portfolios without rebalancing. # INST: number of instances with $n \geq k$; OFV: objective function value (scaled by 100,000); TE: out-of-sample tracking error in [%]; ER: out-of-sample excess return in [%].

		Period 1				Period 2			
		# INST	OFV	TE	ER	# INST	OFV	TE	ER
Scenario I	$k = 100$	9	0.90	3.22	0.84	9	0.99	3.83	−0.08
	$k = 200$	7	0.47	2.40	0.45	8	0.47	2.90	−0.57
Scenario II	$k = 100$	9	22.38	14.58	−0.79	9	18.16	14.26	2.93
	$k = 200$	7	15.12	11.26	−2.01	8	12.73	12.38	3.35
Scenario III	$k = 100$	9	120.32	14.39	6.81	9	134.17	20.40	2.28
	$k = 200$	7	130.15	12.05	2.46	8	140.20	18.47	2.08

- Higher values of k consistently lead to better objective function values and out-of-sample tracking errors, regardless of the considered scenario and period. Note that the portfolios of scenario III are identical for $k = 100$ and $k = 200$ because all these portfolios consist of cash only. The results for $k = 100$ and $k = 200$ differ because for the three smallest instances (tr1, tr2, and tr10) a value of $k = 200$ is not applicable.

- The portfolios of scenario I clearly outperform those of scenarios II and III in terms of the objective function value and out-of-sample tracking error, regardless of the considered period. Therefore, portfolios that consist of stocks with high weights in the index tend to have a better index-tracking capability than portfolios that consist of stocks with low index weights.

- In terms of the out-of-sample excess return, the results are ambiguous. The portfolios of scenario I exhibit positive out-of-sample excess returns during period 1 but negative out-of-sample excess returns during period 2. However, the opposite is true for the portfolios of scenario II. The portfolios of scenario III, i.e., the portfolios that consist of cash only, show positive out-of-sample excess returns for both periods; this occurs because we assume a return of 0% on cash and the average index return over all instances is negative for both periods. Note that these relatively high out-of-sample excess returns are accompanied by very high out-of-sample tracking errors, especially for the portfolios of scenarios II and III.

3.5.4 LBH and IGH: in-sample performance analysis in comparison with the MIQP approach

In this section, we investigate the performance of the heuristic solution approaches LBH and IGH in comparison with the exact solution approach M-Q in terms of the objective function value, i.e., the expected tracking error. For this purpose, we present in-sample results for the portfolios of scenarios I, II, and III after rebalancing using the three considered solution approaches.

Table 3.3 summarizes the main in-sample results. Columns one to four indicate the considered scenario, the maximum portfolio cardinality k, the annualized prescribed minimum expected excess return α, and the number of considered instances, i.e., instances with $n \geq k$, respectively. Columns five, seven, and nine show the objective function values after the imposed computational time limit averaged over the considered instances for the solution approaches M-Q, LBH, and IGH, respectively. Columns six, eight, and ten show the numbers of instances for which M-Q, LBH, and IGH, respectively, were not able to find a feasible portfolio within the prescribed computational time limit.

From Table 3.3, we can gain the following insights:

- Within the prescribed computational time limit, LBH and IGH are able to find much better portfolios in terms of the objective function value than M-Q is.

- LBH and IGH are able to find feasible portfolios for all considered problem instances within the prescribed computational time limit, whereas this is not the case for M-Q. This finding demonstrates the difficulty of constructing a feasible portfolio for the considered problem.

- Compared with the portfolios without rebalancing (cf. Table 3.2), the portfolios found by using LBH and IGH have considerably lower objective function values.

- IGH tends to find better portfolios in terms of the objective function value than LBH does for scenarios I and II. The opposite is true for scenario III, in which the investment fund has a portfolio before rebalancing that consists only of cash. Hence, LBH should be applied to construct a portfolio from cash, and IGH should be applied to rebalance an existing stock portfolio.

Table 3.3: In-sample results (averages over all instances with $n \geq k$ among instances tr1–tr18) for rebalanced portfolios. # INST: number of instances with $n \geq k$; OFV: objective function value (scaled by 100,000); # NFP: number of instances for which no feasible portfolio could be found within the prescribed computational time limit. The best objective function value in each row is shown in bold.

				M-Q		LBH		IGH	
		α p.a.	# INST	OFV	# NFP	OFV	# NFP	OFV	# NFP
Scenario I	$k = 100$	0%	18	13.05	3	0.55	0	**0.49**	0
		1%	18	11.75	4	0.56	0	**0.49**	0
		2%	18	12.95	6	0.58	0	**0.50**	0
	$k = 200$	0%	15	6.55	4	**0.24**	0	0.26	0
		1%	15	4.85	4	**0.25**	0	0.26	0
		2%	15	5.32	4	0.27	0	**0.27**	0
Scenario II	$k = 100$	0%	18	28.08	4	14.35	0	**12.90**	0
		1%	18	24.04	5	13.21	0	**12.34**	0
		2%	18	26.69	3	13.93	0	**12.04**	0
	$k = 200$	0%	15	26.35	5	6.01	0	**5.86**	0
		1%	15	24.86	6	**5.77**	0	6.30	0
		2%	15	24.02	6	**5.56**	0	5.69	0
Scenario III	$k = 100$	0%	18	42.59	0	**0.42**	0	0.43	0
		1%	18	41.61	1	**0.42**	0	0.43	0
		2%	18	41.39	2	**0.43**	0	0.44	0
	$k = 200$	0%	15	49.97	0	**0.17**	0	0.17	0
		1%	15	50.53	1	**0.17**	0	0.17	0
		2%	15	50.94	2	**0.17**	0	0.18	0

3.5.5 LBH and IGH: out-of-sample performance analysis in comparison with benchmark MILP approaches

For the assessment of the out-of-sample performance of the heuristic solution approaches LBH and IGH, we present the out-of-sample tracking errors and out-of-sample excess returns of the portfolios of scenarios I, II, and III after rebalancing using LBH, IGH, and the exact benchmark solution approaches M-L1 and M-L2.

Table 3.4 summarizes the out-of-sample results for period 1 (instances tr1–tr9) and period 2 (instances tr10–tr18) individually. Column 1 of this table indicates the considered period. The contents of columns two to five are the same as those of columns one to four of Table 3.3. Columns six to nine show the average out-of-sample tracking errors for the considered instances, i.e., instances with $n \geq k$, for the solution approaches M-L1, M-L2, LBH, and IGH, respectively. Columns ten to thirteen present the average out-of-sample excess returns for the same instances and the same solution approaches.

From Table 3.4, we can gain the following insights:

- Regardless of the period considered, the average out-of-sample tracking error and the worst-case out-of-sample tracking error for the portfolios obtained with LBH and IGH are much lower than those for the portfolios obtained with M-L1 and M-L2.
- Because LBH and IGH lead to lower out-of-sample tracking errors (i.e., lower relative risk) than M-L1 and M-L2 do, we would also expect the out-of-sample excess returns to be consequently lower. However, for period 1, LBH and IGH devise considerably better portfolios in terms of the average out-of-sample excess return. In period 2, all considered solution approaches yield slightly negative average out-of-sample excess returns.
- LBH and IGH lead to considerably lower worst-case out-of-sample tracking errors and considerably higher worst-case out-of-sample excess returns than M-L1 and M-L2 do, regardless of the period considered. These empirical findings support the arguments presented in Subsection 3.3.2.
- For all considered solution approaches, the average out-of-sample tracking errors are higher for period 2 than for period 1 because the out-of-sample period of period 2 exhibits higher market volatility than that of period 1.
- For scenario II, in which the index-tracking capability of the portfolio before rebalancing is rather poor, the out-of-sample tracking errors are higher than those for the other scenarios. This is consistent with the higher objective function values, i.e., the higher expected tracking errors, for scenario II (cf. Table 3.3).

- An increase in α does not consistently lead to a higher out-of-sample excess return. This shows that a portfolio that achieves the prescribed minimum expected excess return during the in-sample period is not necessarily guaranteed to achieve an excess return during the out-of-sample period.

3.5.6 LBH and IGH: portfolio compositional characteristics in comparison with benchmark MILP formulations

In this subsection, we offer further insights with respect to the compositions of the rebalanced portfolios. For this purpose, we report various compositional characteristics of portfolios that have been rebalanced using the solution approaches LBH, IGH, M-L1, and M-L2.

Figure 3.2 shows the following compositional characteristics for the portfolios of scenarios I, II, and III after rebalancing: the active share, i.e., the sum of the absolute differences between the weights of the assets in the portfolio and the weights of the assets in the index; the portfolio cardinality, i.e., the number of different stocks that are selected after rebalancing; the transaction costs, i.e., the sum of the fixed and proportional transaction costs relative to the transaction cost budget; and the weight of the cash asset. We present the compositional characteristics for all instances with $n \geq k$ among the instances tr1–tr18, sorted in non-decreasing order of n, with $k = 200$ and $\alpha = 0\%$. The compositional characteristics for $k = 100$ and $\alpha > 0\%$ are not shown because they are similar to those presented in Figure 3.2. For reference, we also present the compositional characteristics in the case that no rebalancing is performed.

From Figure 3.2, we can gain the following insights:

- LBH and IGH lead to portfolios with lower active shares and higher cardinalities than M-L1 and M-L2 do, regardless of the considered scenario. Thus, portfolios that have been rebalanced using LBH and IGH are more diversified. The superior performance of LBH and IGH in terms of the out-of-sample tracking error and the out-of-sample excess return (cf. Table 3.4) may be attributable to these findings.

- In scenario I, LBH and IGH incur lower transaction costs than M-L1 and M-L2 do because the portfolio before rebalancing is already invested in the k stocks with the highest index weights. In scenarios II and III, the transaction costs for LBH and IGH are higher than those for M-L1 and M-L2. To rebalance the portfolios with a poor index-tracking capability of scenario II, LBH and IGH completely exhaust the transaction cost budget for all problem instances.

Table 3.4: Out-of-sample results (averages over the instances with $n \geq k$ among instances tr1–tr9 (period 1) and instances tr10–tr18 (period 2)) for rebalanced portfolios. # INST: number of instances with $n \geq k$; TE: out-of-sample tracking error in [%]; ER: out-of-sample excess return in [%]. The best TE and ER in each row are shown in bold.

					TE				ER			
			α p.a.	# INST	M-L1	M-L2	LBH	IGH	M-L1	M-L2	LBH	IGH
Period 1	Scenario I	$k = 100$	0%	9	3.36	3.20	2.87	**2.75**	0.97	**1.00**	0.87	0.78
			1%	9	3.51	3.22	2.89	**2.74**	**1.74**	1.28	1.29	0.85
			2%	9	3.39	3.32	2.88	**2.71**	1.10	**1.43**	1.39	**1.43**
		$k = 200$	0%	7	2.14	2.22	**2.01**	2.04	−0.31	−0.44	0.47	**0.65**
			1%	7	2.23	2.29	**2.06**	**2.06**	0.10	−0.33	**0.65**	0.40
			2%	7	2.26	2.22	2.11	**2.04**	−0.51	0.02	**0.82**	0.79
	Scenario II	$k = 100$	0%	9	6.29	6.23	4.46	**4.21**	−0.45	**0.53**	−0.95	−1.09
			1%	9	6.17	6.18	4.29	**4.15**	**−0.28**	−0.29	−1.57	−0.51
			2%	9	6.21	6.28	**4.21**	4.26	−0.40	0.11	**1.21**	−0.51
		$k = 200$	0%	7	5.85	5.92	**3.73**	3.79	−2.45	−2.33	−1.84	**−1.46**
			1%	7	5.91	5.96	**3.76**	3.77	−2.90	−2.52	**−0.99**	−1.18
			2%	7	6.04	5.93	3.79	**3.77**	−2.91	−2.05	**−0.60**	−0.92
	Scenario III	$k = 100$	0%	9	5.26	5.17	**2.70**	2.73	−0.79	−1.26	0.60	**0.77**
			1%	9	5.25	5.31	**2.68**	2.73	−1.57	−0.71	0.52	**0.73**
			2%	9	5.22	5.26	2.86	**2.70**	−2.22	−1.30	**0.80**	0.74
		$k = 200$	0%	7	5.00	4.97	**1.92**	1.94	−1.83	−3.20	**0.31**	−0.01
			1%	7	4.93	4.99	**1.92**	**1.92**	−3.33	−3.37	0.31	**0.64**
			2%	7	4.86	4.98	1.95	**1.93**	−3.96	−2.65	**0.29**	0.21
				Average	4.70	4.68	2.99	**2.94**	−0.96	−0.77	**0.23**	0.16
				Worst case	10.04	9.51	5.98	**5.75**	−19.82	−17.73	−9.00	**−8.90**
Period 2	Scenario I	$k = 100$	0%	9	3.38	3.64	3.12	**2.93**	**0.95**	0.64	−0.60	−0.61
			1%	9	3.37	3.45	3.15	**2.92**	**2.11**	0.43	−0.79	−1.09
			2%	9	3.23	3.60	3.15	**2.85**	0.86	**0.88**	−0.69	−0.15
		$k = 200$	0%	8	3.12	3.25	2.33	**2.29**	**1.18**	1.00	−0.81	−0.58
			1%	8	2.93	3.16	**2.28**	2.31	0.42	**0.55**	−0.79	−0.80
			2%	8	2.88	3.21	2.33	**2.30**	0.23	**1.03**	−0.76	−0.93
	Scenario II	$k = 100$	0%	9	6.05	5.83	5.24	**4.98**	−1.18	−1.53	−0.94	**−0.12**
			1%	9	6.13	5.85	5.16	**4.83**	−0.80	−1.91	**−0.58**	−1.18
			2%	9	6.11	5.85	5.04	**4.65**	−0.45	−1.65	**0.31**	−0.29
		$k = 200$	0%	8	7.25	6.97	**4.10**	4.21	−4.40	−3.32	−0.74	**−0.05**
			1%	8	7.23	7.04	**4.06**	4.15	−4.93	−3.23	−0.52	**−0.08**
			2%	8	7.38	8.23	**4.04**	4.14	−4.59	−3.45	**−0.11**	−0.57
	Scenario III	$k = 100$	0%	9	5.40	5.42	**2.81**	2.82	**0.01**	−0.34	−0.44	−0.41
			1%	9	5.71	5.03	2.84	**2.82**	0.01	**0.93**	−0.28	−0.76
			2%	9	5.58	4.90	**2.80**	2.84	**0.50**	0.38	−0.66	−0.91
		$k = 200$	0%	8	5.47	5.15	2.25	**2.23**	**0.40**	−0.42	−0.73	−0.57
			1%	8	5.65	4.97	**2.22**	**2.22**	0.11	**1.52**	−0.86	−0.75
			2%	8	5.52	4.90	**2.20**	2.25	0.82	**1.04**	−0.96	−0.99
				Average	5.12	5.02	3.31	**3.23**	−0.44	**−0.41**	−0.60	−0.60
				Worst case	9.92	15.34	7.60	**6.61**	−15.14	−15.08	**−4.39**	−5.31

Note: M-L1 and M-L2 were not able to find a feasible portfolio for 2 instances each.

- For scenarios I and III, LBH and IGH lead to portfolios in which the weight of the cash asset is almost zero for all instances. By contrast, M-L1 and M-L2 lead to portfolios that hold a substantial amount of cash after rebalancing for some larger instances. For scenario II, in which the index-tracking capacity of the portfolio before rebalancing is rather poor, LBH and IGH also lead to portfolios that contain a considerable proportion of cash. This is because the transaction cost budget is not sufficiently large to sell all currently held stocks with low index weights and exchange them for stocks with high index weights. In this case, it is more beneficial in terms of the objective function value to sell the stocks with low index weights and maintain the revenue in cash.

From the results provided in this section, the six main findings are as follows: 1) LBH and IGH lead to better in-sample portfolios than M-Q does. 2) LBH should be used to construct a portfolio from cash. 3) IGH should be used to rebalance an existing stock portfolio. 4) In terms of the out-of-sample tracking error, LBH and IGH lead to better portfolios than the two benchmark approaches M-L1 and M-L2 do. 5) In periods of low market volatility, LBH and IGH also lead to better out-of-sample excess returns than the benchmark approaches M-L1 and M-L2 do. 6) In periods of high market volatility, all approaches lead to negative out-of-sample excess returns; however, the worst-case out-of-sample returns are considerably higher for LBH and IGH than for the benchmark approaches M-L1 and M-L2.

3.6 Conclusions

In this paper, we considered the problem of determining the portfolio for an enhanced index-tracking fund. We proposed two novel matheuristics that are based on a novel mixed-integer quadratic programming formulation. In a computational experiment, the proposed matheuristics outperformed an exact solution approach based on this novel mixed-integer quadratic programming formulation in terms of the objective function value within a limited computational time, and they outperformed two benchmark mixed-integer linear programming approaches in terms of the out-of-sample tracking error and the out-of-sample excess return.

In future research, it would be interesting to investigate whether the out-of-sample tracking error and out-of-sample excess return achieved here could be further improved by using more sophisticated estimators for the covariance matrix and the expected returns of single stocks. Periodic portfolio rebalancing could be applied to further improve the out-of-sample performance of the presented solution approaches, especially in market

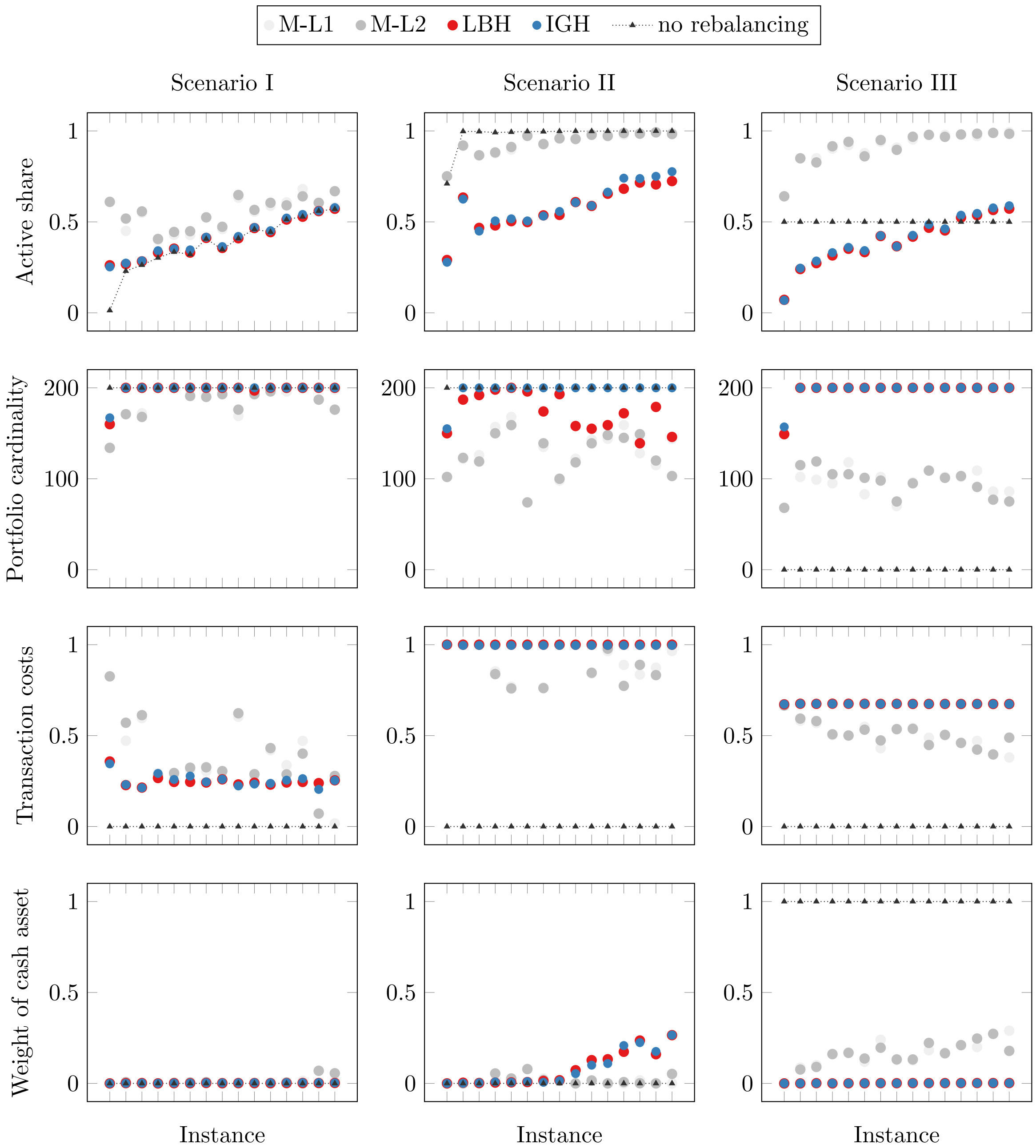


Figure 3.2: Compositional characteristics of portfolios rebalanced using M-L1, M-L2, LBH, and IGH (for all instances with $n \geq k$ among instances tr1–tr18, sorted in non-decreasing order of n, with $k = 200$ and $\alpha = 0\%$). Active share: sum of the absolute differences between the weights of the assets in the portfolio after rebalancing and the weights of the assets in the index ($\frac{1}{2}\sum_{i\in U\cup\{n+1\}}\left|\frac{P_{iT}X_i}{C} - w_i^I\right|$); Portfolio cardinality: number of different stocks selected after rebalancing ($|\{i \in I : X_i > 0\}|$); Transaction costs: sum of the fixed and proportional transaction costs relative to the budget for transaction costs ($\frac{1}{\gamma C}(\sum_{i\in I} G_i + \sum_{i\in U\setminus I: Y_i>0}(c_i^s Y_i P_{iT} + c_i^f))$); Weight of cash asset: weight of the cash asset after rebalancing ($\frac{P_{n+1,T}X_{n+1}}{C}$).

environments with high volatility. Moreover, a promising direction for future research is to combine the two presented matheuristics based on the findings regarding their individual strengths to obtain an even more powerful matheuristic.

Bibliography

Andriosopoulos, K., Nomikos, N., 2014. Performance replication of the Spot Energy Index with optimal equity portfolio selection: Evidence from the UK, US and Brazilian markets. European Journal of Operational Research 234 (2), 571–582.

Beasley, J. E., Meade, N., Chang, T.-J., 2003. An evolutionary heuristic for the index tracking problem. European Journal of Operational Research 148 (3), 621–643.

Bertsimas, D., Shioda, R., 2009. Algorithm for cardinality-constrained quadratic optimization. Computational Optimization and Applications 43 (1), 1–22.

Bruni, R., Cesarone, F., Scozzari, A., Tardella, F., 2015. A linear risk-return model for enhanced indexation in portfolio optimization. OR Spectrum 37 (3), 735–759.

Canakgoz, N. A., Beasley, J. E., 2008. Mixed-integer programming approaches for index tracking and enhanced indexation. European Journal of Operational Research 196 (1), 384–399.

Chiam, S. C., Tan, K. C., Al Mamun, A., 2013. Dynamic index tracking via multi-objective evolutionary algorithm. Applied Soft Computing 13 (7), 3392–3408.

Filippi, C., Guastaroba, G., Speranza, M., 2016. A heuristic framework for the bi-objective enhanced index tracking problem. Omega 65, 122–137.

Fischetti, M., Lodi, A., 2003. Local branching. Mathematical Programming 98 (1–3), 23–47.

Gaivoronski, A. A., Krylov, S., Van der Wijst, N., 2005. Optimal portfolio selection and dynamic benchmark tracking. European Journal of Operational Research 163 (1), 115–131.

Guastaroba, G., Mansini, R., Ogryczak, W., Speranza, M. G., 2016. Linear programming models based on Omega ratio for the enhanced index tracking problem. European Journal of Operational Research 251 (3), 938–956.

Guastaroba, G., Mansini, R., Speranza, M. G., 2009. On the effectiveness of scenario generation techniques in single-period portfolio optimization. European Journal of Operational Research 192 (2), 500–511.

Guastaroba, G., Speranza, M. G., 2012. Kernel Search: An application to the index tracking problem. European Journal of Operational Research 217 (1), 54–68.

Jansen, R., Van Dijk, R., 2002. Optimal benchmark tracking with small portfolios. The Journal of Portfolio Management 28 (2), 33–39.

Jorion, P., 2003. Portfolio optimization with tracking-error constraints. Financial Analysts Journal 59 (5), 70–82.

Konno, H., Wijayanayake, A., 2001. Minimal cost index tracking under nonlinear transaction costs and minimal transaction unit constraints. International Journal of Theoretical and Applied Finance 4 (6), 939–957.

Krink, T., Mittnik, S., Paterlini, S., 2009. Differential evolution and combinatorial search for constrained index-tracking. Annals of Operations Research 172 (1), 153–176.

Kwiatkowski, J. W., 1992. Algorithms for index tracking. IMA Journal of Management Mathematics 4 (3), 279–299.

Ledoit, O., Wolf, M., 2004a. Honey, I shrunk the sample covariance matrix. The Journal of Portfolio Management 30 (4), 110–119.

Ledoit, O., Wolf, M., 2004b. A well-conditioned estimator for large-dimensional covariance matrices. Journal of Multivariate Analysis 88 (2), 365–411.

Maringer, D., Oyewumi, O., 2007. Index tracking with constrained portfolios. Intelligent Systems in Accounting, Finance and Management 15 (1-2), 57–71.

Mutunge, P., Haugland, D., 2018. Minimizing the tracking error of cardinality constrained portfolios. Computers & Operations Research 90, 33–41.

Roll, R., 1992. A mean/variance analysis of tracking error. The Journal of Portfolio Management 18 (4), 13–22.

Rudd, A., 1980. Optimal selection of passive portfolios. Financial Management, 57–66.

Rudolf, M., Wolter, H.-J., Zimmermann, H., 1999. A linear model for tracking error minimization. Journal of Banking & Finance 23 (1), 85–103.

Ruiz, R., Stützle, T., 2007. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. European Journal of Operational Research 177, 2033–2049.

Sant'Anna, L. R., Filomena, T. P., Caldeira, J. F., 2017a. Index tracking and enhanced indexing using cointegration and correlation with endogenous portfolio selection. The Quarterly Review of Economics and Finance 65, 146–157.

Sant'Anna, L. R., Filomena, T. P., Guedes, P. C., Borenstein, D., 2017b. Index tracking with controlled number of assets using a hybrid heuristic combining genetic algorithm and non-linear programming. Annals of Operations Research 258 (2), 849–867.

Scozzari, A., Tardella, F., Paterlini, S., Krink, T., 2013. Exact and heuristic approaches for the index tracking problem with UCITS constraints. Annals of Operations Research 205 (1), 235–250.

Strub, O., Baumann, P., 2018. Optimal construction and rebalancing of index-tracking portfolios. European Journal of Operational Research 264 (1), 370–387.

Strub, O., Trautmann, N., 2016. An iterated greedy heuristic for the 1/N portfolio tracking problem. In: Vitoriano, B., Parlier, G., de Werra, D. (Eds.), Proceedings of the 5th International Conference on Operations Research and Enterprise Systems. Rome, pp. 424–431.

Takeda, A., Niranjan, M., Gotoh, J., Kawahara, Y., 2013. Simultaneous pursuit of out-of-sample performance and sparsity in index tracking portfolios. Computational Management Science 10 (1), 21–49.

Wu, D., Kwon, R. H., Costa, G., 2017. A constrained cluster-based approach for tracking the S&P 500 index. International Journal of Production Economics 193, 222–243.