Pós-Graduação em Ciência da Computação

Julio Cezar Soares Silva

Index tracking model through an enhanced GRASP approach for the financial portfolio problem

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
http://cin.ufpe.br/~posgraduacao

Recife
2020

Julio Cezar Soares Silva

Index tracking model through an enhanced GRASP approach for the financial portfolio problem

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Teoria da computação
Orientador: Adiel Teixeira de Almeida Filho

Recife
2020

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S586i Silva, Julio Cezar Soares
Index tracking model through an enhanced GRASP approach for the financial portfolio problem / Julio Cezar Soares Silva. – 2020.
100 f.: il., fig., tab.

Orientador: Adiel Teixeira de Almeida Filho.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2020.
Inclui referências.

1. Teoria da computação. 2. Revisão sistêmica. I. Almeida Filho, Adiel Teixeira de (orientador). II. Título.

004 CDD (23. ed.) UFPE - CCEN 2020 - 183

Julio Cezar Soares Silva

"Index Tracking Model Through an Enhanced GRASP Approach for the Financial Portfolio Problem"

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 25/09/2020.

BANCA EXAMINADORA

Prof. Dr. Adriano Lorena Inacio de Oliveira
Centro de Informática / UFPE

Prof. Dr. Mauricio Guilherme de Carvalho Resende
Mathematical Optimization and Planning / Amazon

Prof. Dr. Adiel Teixeira de Almeida Filho
Centro de Informática / UFPE
(Orientador)

Dedico este trabalho à minha família

ACKNOWLEDGEMENTS

Agradeço aos meus pais Eleno e Josy e à toda minha família pelo apoio incondicional.

Agradeço ao meu orientador Adiel Teixeira de Almeida Filho pelo suporte, oportunidades, conselhos e momentos de descontração proporcionados.

Agradeço ao Centro de Informática da UFPE que disponibilizou uma ótima infraestrutura e também aos professores pelo conhecimento fornecido ao longo deste período.

Agradeço ao Grupo de Pesquisa em Gerenciamento de Riscos, Estudos de Engenharia Financeira e Otimização (GREEFO) pelas diversas contribuições em trabalhos e momentos de aprendizado.

Agradeço aos meus colegas do laboratório de pesquisa pelos momentos de descontração e discussão sobre diversos temas.

Agradeço ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e FACEPE (Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco) que financiaram este estudo parcialmente.

ABSTRACT

Financial portfolio optimization problems may become computationally infeasible when some practical constraints are considered in the model. In these circumstances, it is difficult to find an optimal solution in a reasonable time. An investment strategy that aims to replicate the performance of a stock market index, whose model solution is included in this class of difficult problems, is called index tracking. This work brings an analysis, spanning the last decade, about the advances in solution approaches for index tracking. The systematic literature review covered important issues, such as the most relevant research areas, solution methods, and model structures. Also, the author presents a novel application of *Greedy Randomized Adaptive Search Procedure* (GRASP) for index tracking. It was sought to implement and adapt a heuristic that was not yet applied to the index tracking problem and evaluate its performance relative to a commercial solver. It was necessary to develop a new greedy function and to compare the results after greedy and random solution construction. Besides, a way is proposed to improve a local search component in the selected GRASP metaheuristic. By conducting computational experiments, GRASP and a general-purpose solver have been compared using benchmark instances. The results showed that GRASP found solutions with almost the same quality as those of CPLEX solver in a smaller time. Moreover, it was observed that the proposed local search component implied in obtaining better solutions relative to those of the reference GRASP metaheuristic. Not performing statistical tests when comparing solution methods, using only benchmark instances and one index tracking model can be considered as limitations of this work. The practical implication of this research is the achievement of good solutions for the index tracking problem in a smaller time and new perspectives for building GRASP heuristics for portfolio optimization problems. As far as we know, this is the first time that a GRASP heuristic was used in this type of problem. GRASP has a great potential in portfolio optimization, more specifically in solving index tracking problems. With a simple parameter tuning procedure, it was possible to obtain good solutions in a smaller time.

Keywords: GRASP. Index tracking. Heuristic. Systematic Review.

RESUMO

Formulações de problemas de otimização de portfólio de investimento podem se tornar computacionalmente inviáveis a partir da inserção de determinadas restrições práticas, tornando o processo de obtenção da solução ótima mais custoso e até impossível, dadas as limitações de recursos físicos e temporais. Uma estratégia de investemento que visa replicar o desempenho de um índice de mercado de ações, cuja solução do modelo está contida nesta classe de problemas difíceis, é denominada *index tracking*. Este trabalho traz uma análise, abrangendo a última década, sobre os avanços nas abordagens de solução para o problema de *index tracking*. A revisão sistemática da literatura abordou questões importantes, como as áreas de pesquisa mais relevantes, métodos de solução e estruturas de modelos. Também foi apresentada uma nova aplicação da metaheurística GRASP para modelos de index tracking. Buscou-se implementar e adaptar uma heurística ainda não aplicada ao problema de *index tracking* e avaliar seu desempenho com relação à um solver comercial. Foi necessário desenvolver uma nova *greedy function* e comparar, respectivamente, os resultados após as construções gananciosa e aleatória da solução. Além disso, foi proposta uma melhoria para o componente de busca local da metaheuristica GRASP adotada. Através de experimentos computacionais, a heurística e um solver comercial foram comparados utilizando instâncias da literatura. Os resultados mostram que a heurística desenvolvida encontrou soluções com qualidade próxima das soluções do solver CPLEX em um menor período de tempo. Também foi possível observar que o componente de busca local desenvolvido implica na obtenção de melhores soluções que àquelas da metaheurística GRASP escolhida como base. A não realização de testes estatísticos nas comparações entre os métodos de solução, utilização exclusiva de dados da literatura e de um único modelo de index tracking podem ser consideradas limitações deste trabalho. A implicação prática desta pesquisa é a obtenção de boas soluções para o problema de index tracking em um instante de tempo reduzido e novas perspectivas para construção de soluções GRASP para o problema de portfólio. Até onde sabemos esta foi a primeira vez que uma heurística GRASP foi utilizada neste tipo de problema. GRASP tem um grande potencial no problema de otimização de portfólio, mais especificamente em problemas de index tracking, onde a partir de um procedimento simples de calibração dos parâmetros foi possível obter boas soluções em um instante de tempo menor.

Palavras-chaves: GRASP. Index tracking. Heurística. Revisão sistemática.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ARCGA	*Adaptive Real-coded Genetic Algorithm*
CCPO	*Cardinality Constrained Portfolio Optimization*
CVaR	*Conditional Value-at-Risk*
ETFs	*Exchange Trading Funds*
FA	*Firefly Algorithm*
GA	*Genetic Algorithm*
GQ	*GRASP-QUAD*
GRASP	*Greedy Randomized Adaptive Search Procedure*
ICA	*Imperialist competitive algorithm*
MAD	*Mean Absolute Deviation*
MaLS	*Major Local Search*
MILP	*Mixed-Integer Linear Programming*
MiLS	*Minor Local Search*
MINLP	*Mixed-Integer non-linear programming*
MIP	*Mixed-Integer Programming*
MOEA	*Multi-objective Evolutionary Algorithm*
MOEA/D	*Multi-objective Evolutionary Algorithm based on Decomposition*
MVO	*Mean-Variance Optimization*
NSGA-II	*Nondominated sorting genetic algorithm II*
PSO	*Particle Swarm Optimization*
RCL	*Restricted Candidate List*
SA	*Simulated Annealing*
SAA	*Sample Average Approximation*
SI	*Swarm Intelligence*
SV	*Semi-Variance*
TEV	*Variance of Tracking-Error*
VaR	*Value-at-Risk*

LIST OF SYMBOLS

t	Time index
i	Asset index
q_t^+	Variable used to linearize the Mean Absolute Deviation function
q_t^-	Variable used to linearize the Mean Absolute Deviation function
R^t	Index return at time t
r_i^t	Stock i return at time t
x_i	Stock i proportion in the portfolio
Z_i	Indicates if stock i is included in the portfolio or not
K	Number of assets in the portfolio
L_i	Lower bound on stock i
U_i	Upper bound on stock i
RCL_{size}	Restricted Candidate List Size
ϕ	ϕ-DMiLS depth (maximum number of stocks to be exchanged in the local search)
g_i	Greedy function
$iter_{MAX}$	Maximum number of iterations performed by GRASP-QUAD
S	GRASP-QUAD's trial solution
j	Index of stocks contained in GRASP-QUAD's trial solution
p_{MiLS}	Probability of MiLS event
UB_{MiLS}	Maximum number of MiLS iterations
f_w	Solution obtained by CPLEX
f_g	Solution obtained by GRASP-QUAD
GAP_{CPLEX}	Output provided by CPLEX solver after the end of the optimization
$CPUTime_{CPLEX}$	Time elapsed by CPLEX solver to optimize decision variables of the selected model

CONTENTS

1 INTRODUCTION

Investment analysis is a strategic process for an organization and individual investors as it involves capital allocations that need to be made efficiently. In this scenario, financial agents constantly deal with the tradeoff between walking through a riskier path (under the perspective of higher returns) and taking less risk and being satisfied with lower (but less uncertain) returns. The uncertainty and ambiguity regarding information (LOTFI; ZENIOS, 2018; YOSHIDA, 2020), fluctuations in stock prices (DUTTA et al., 2018; ARAUJO et al., 2018; ARAUJO et al., 2019), and adequation of the model constraints to reflect real world issues (FERREIRA et al., 2018) represent some of the challenges that researchers face.

Since the seminal studies by Markowitz (1952) and Roy (1952), which represented the starting points of Modern Portfolio Theory, several papers have been published and advances have been observed in the portfolio selection problem, as shown in Kolm, Tuetuencue & Fabozzi (2014). The classical Markowitz *Mean-Variance Optimization* (MVO) consists of a quadratic optimization model, the objectives of which are the maximization of returns and the minimization of risk. Markowitz (1952) showed that, for each particular level of risk, an optimal return can be obtained. Therefore, an efficient frontier is obtained. The same result can be achieved by retaining levels of return and obtaining a minimal risk for them.

Kolm, Tuetuencue & Fabozzi (2014) affirm that financial experts are apprehensive about the application of the classical MVO in real data, especially because of the sensitivity of the optimal weight allocation relative to the perturbation of the model inputs. Also, the generalized difficulty intrinsic to parameter estimation, as discussed by Fama (1970), being impossible to predict future returns one or more days before the portfolio's rebalancing day, brings more difficulties concerning the consistency and robustness of results produced using this model. In this way, the result of the optimization using this historical data approach leads to counter-intuitive portfolios, therefore more robust forecasts are needed (FABOZZI et al., 2007). Besides criticisms, MVO formulations are largely applied in portfolio optimization studies, where the most commonly used risk measure is variance (ERTENLICE; KALAYCI, 2018).

1.1 MOTIVATION

Among the advances found in the literature concerning the Portfolio Theory, several applications consider implementing new constraints to make the model more realistic (JALOTA; THAKUR, 2018; CACADOR; DIAS; GODINHO, 2020; GUPTA et al., 2020). Examples include transaction costs, class and cardinality constraints. Cardinality constraints limit the portfolio size (control the number of assets) and are important to reduce transaction costs (FABOZZI et al., 2007). On the other hand, the additional considerations can make the model computationally infeasible, thereby increasing computational complexity and making classical exact models inefficient. For instance, Mutunge & Haugland (2018) calls attention to the strong computational costs imposed by cardinality constraints. These extended problems are NP-hard (FABOZZI et al., 2007). Thus, using metaheuristics as alternative approaches has grown in past years to deal with complex portfolio optimization models (DUTTA et al., 2018; LIAGKOURAS; METAXIOTIS, 2018b; EFTEKHARIAN; SHOJAFAR; SHAMSHIRBAND, 2017).

A fund manager normally needs to choose between an active strategy and a passive strategy. An active strategy consists of trying to outperform the market by moves, such as picking the winner stocks (ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015). In the passive strategy, known as index tracking, the aim is to reproduce the performance of a chosen benchmark by allocating capital to a subset of assets that represent the index (MUTUNGE; HAUGLAND, 2018; WANG; XU; DAI, 2018; SCOZZARI et al., 2013). The use of a subset (partial allocation) is necessary to deal with the high transaction costs, as discussed in Wang, Xu & Dai (2018) and Sant'Anna et al. (2017), that can negatively impact accumulated returns. Thus, constraints on the number of assets are incorporated in this type of model (MUTUNGE; HAUGLAND, 2018), making them computationally infeasible as the number of constituent assets in the benchmark grows.

1.2 RELEVANCE OF THE RESEARCH PROBLEM

Passive fund management emerged from the efficient market hypothesis of Fama (1970), where the best strategy an investor can perform is to follow the market, otherwise, he/she will perform worse than the market. This type of approach is an alternative for conservative investors since it is less risky and usually brings returns close to the benchmark index that is being tracked by the model (RUIZ-TORRUBIANO; SUAREZ, 2009). Stock market indexes contain hundred of assets and its full replication implies high transaction costs, therefore harming portfolio returns (FABOZZI et al., 2007; SANT'ANNA et al., 2019; SANT'ANA; CALDEIRA; FILOMENA, 2020).

Index tracking models are used to replicate the performance of a stock market index by investing in a subset of the constituent stocks. This problem is difficult for a computer to solve as the number of stocks grows. Recently, many computational studies have been developed for this kind o problem, since it is important to find a good solution within a reasonable period of computational time. Those studies were developed mainly by the popularization of metaheuristics/heuristics as tools to search for approximate optimal solutions (ANDRIOSOPOULOS et al., 2019), and also the importance of considering rebalancing process strategies (STRUB; BAUMANN, 2018; SANT'ANNA et al., 2019).

The relevance of this problem is important for the maintenance of investment opportunities, since the solutions of index tracking models contribute to the construction and decisions relative to the management of *Exchange Trading Funds* (ETFs) (SANT'ANNA et al., 2019). This work brings to light the state-of-the-art solution methods developed for the index tracking problem in the last decade and the development of a novel application of a heuristic that was not yet tested in this type of problem. The GRASP heuristic was chosen since it has not yet been deeply explored in the modern portfolio theory context. In section 3.3 the only three works on GRASP related to portfolio selection are presented, indicating that GRASP has a competitive performance to deal with financial portfolio selection (ANAGNOSTOPOULOS; CHATZOGLOU; KATSAVOUNIS, 2010; BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015; OTKEN et al., 2019). Amongst these works, only Baykasoglu, Yunusoglu & Ozsoydan (2015) compared GRASP with other heuristic approaches, such as genetic algorithms, particle swarm optimization and tabu search, showing that GRASP a competitive performance. Thus, since until now there is no application or study with regard to GRASP for solving the index tracking problem, and given the potential of the GRASP based metaheuristic developed by Baykasoglu, Yunusoglu & Ozsoydan (2015), this dissertation will study the adaptation and evaluation of a GRASP approach for building index tracking portfolios.

1.3 OBJECTIVES

1.3.1 General objective

Contribute to the financial portfolio optimization literature by investigating the state-of-the-art solution approaches to the index tracking problem and proposing a GRASP heuristic for the index tracking problem

1.3.2 Specific objectives

- Build a conceptual literature review regarding advances in solution approaches for the financial portfolio optimization problem
- Build a systematic literature review concerning advances in solution approaches for the index tracking problem in the last decade
- Build a systematic literature review on current GRASP applications to financial portfolio problems
- Implement and adapt a reference GRASP metaheuristic for the index tracking problem
- Implement a GRASP heuristic containing incremental modifications
- Search and collect data in the literature
- Select an index tracking model to evaluate the solutions approaches

- Test the proposed heuristic approaches against a benchmark commercial solver

1.4 METHODOLOGY

The methodology of this work is presented in Figure 1. The first stage consisted of determining the chronogram of the research, selecting scientific databases and books and defining the research questions for systematic literature reviews. In the second stage a theoretical foundation is constructed, conceptual (portfolio selection) and systematic literature reviews (index tracking and GRASP for portfolio selection) are developed. Stage 3 consists of bibliometric analysis and identification of gaps in the literature. At stage 4 the tools and materials are selected and prepared. In the next stage, the experimentation and calculation of performance metrics are made. In the final stage, experimental data is consolidated and interpreted. Finally, discussions and conclusions are drawn.

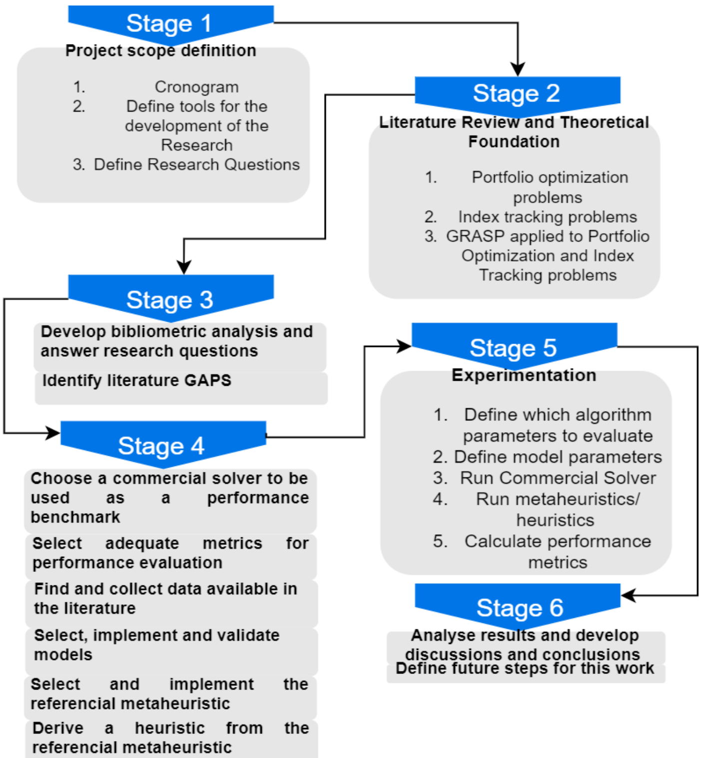

Figure 1 – Methodology

1.5 CONTRIBUTIONS

We present a literature review comprising solution approaches for the index tracking problem over the last decade and an implementation of a metaheuristic named GQ to solve an index tracking problem. The first contribution of this dissertation is the literature review, which raises relevant questions such as most used model structures, most used solution approaches, most relevant solution methods and most used data sources. This brings an overview of the development of index tracking research over the last decade.

The GRASP procedure was first explored in specific problems, such as the set covering problem (FEO; RESENDE, 1989) and the maximum independent set problem (FEO; RESENDE; SMITH, 1994). It then became a broad approach for combinatorial optimization problems, as discussed in FEO & RESENDE (1995), and several applications have been presented in the literature for different fields, such as scheduling, routing, logic, partitioning, location, layout, and manufacturing (FESTA; RESENDE, 2009b; RESENDE; RIBEIRO, 2016). On the other hand, GRASP has not been broadly explored in portfolio selection. The studies by Anagnostopoulos, Chatzoglou & Katsavounis (2010), Baykasoglu, Yunusoglu & Ozsoydan (2015) and Otken et al. (2019) are the ones that were identified. To the best of our knowledge, applications of GRASP for index tracking problems were not found.

Contributions relative to the implementation of GRASP to the index tracking problem are threefold. First, it introduces the use of GRASP for index tracking problems, thereby extending the state of art considering GRASP applications for portfolio selection. Second, it proposes a greedy function considering the goal of following a specific index. Finally, it modifies a local search procedure to reduce the computational cost necessary to find a good solution. The efficiency of the proposed algorithm is compared with CPLEX commercial solver. This solver was also used for this purpose in Sant'Anna et al. (2017), Lejeune & Samatli-Pac (2013), Guastaroba & Speranza (2012). The experiments to evaluate a GRASP metaheuristic adapted for the index tracking problem are designed in the light of the goals of theoretical studies about stochastic algorithms (MARTI; PARDALOS; RESENDE, 2018).

1.6 ORGANIZATION OF THE DISSERTATION

The rest of this work is organized as follows:

2 Theoretical Foundation: Presents theoretic concepts which the author considered necessary as basic knowledge to start developing this work

3 Literature Review: Chapter 3 gives a brief review of recent metaheuristic, heuristic and other solution applications for portfolio selection. Also, the systematic literature reviews that support the development and innovation of this work are presented.

4 Proposed GRASP approach for the Index Tracking Problem: details the proposed GRASP.

5 Experiments and results: presents results and discussions.

6 Conclusions and Future Work: The last chapter draw some conclusions and make suggestions for future lines of research.

2 THEORETICAL FOUNDATION

2.1 COMBINATORIAL OPTIMIZATION, TRACTABLE AND INTRACTABLE PROBLEMS

Mathematical models of optimization can be represented using the following general form:

$$\text{minimize} \quad f(x) \tag{2.1}$$

$$\text{subject to} \quad X \tag{2.2}$$

Where $f : X \rightarrow \Re$ represents a cost function and X is a set of available choices for x, in other words, it is a constraint set. What we want is to find an optimal solution $x^* \in X$ for this problem, such that:

$$f(x^*) \leq f(x), \vee x \in X \tag{2.3}$$

One of the most important features of an optimization problems is if it is discrete or continuous. Continuous problems are those where the constraint set X is infinite. Discrete or combinatorial problems are those where the constraint set X is finite. Linear programming problems occur when f is specified by linear equations and X is specified by linear inequations: $f = c^T x$ and $X = \{x|Ax \leq b \text{ and } x \geq 0\}$, where x is a n-dimensional column vector and the instance of the problem (input data) is defined by $c_{n\times1}$, $A_{m\times n}$ and $b_{m\times1}$. Nonlinear programming problems occur when f or X are specified by nonlinear equations, i.e. quadratic optimization problems where $f = x^T Bx$ and B is a n-dimensional square coefficient matrix (BERTSEKAS, 1999).

Consider an instance of an one-dimensional optimization problem with $X = [0, S]$ and cost function f shown in figure 2. The points A, B and C are locally optimal among their respective neighbors, but only B is globally optimal. Depending on the instance of some problems (linear or nonlinear) finding an optimal solution can be very difficult but it is often possible to find a good solution, in a region of the search space defined by the constraint set X, which is the best relative to its neighbors (PAPADIMITRIOU; STEIGLITZ, 1998).

Some applications require formulating and solving Integer Programming, i.e. x_i is the number of items of type i produced or bought by a company. Linear Integer Programming consists in the following optimization problem:

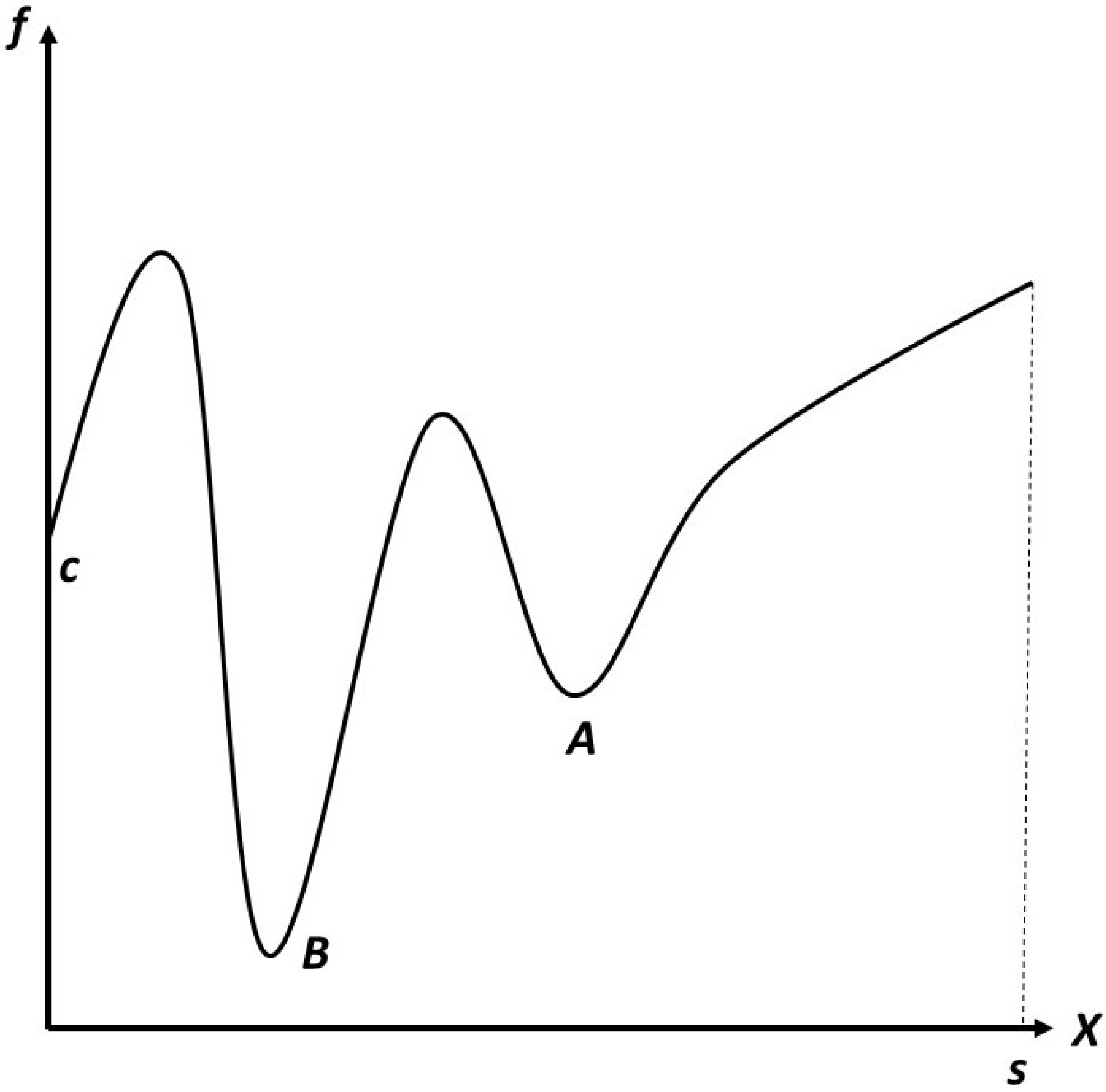

Figure 2 – A 1-dimensional Euclidian optimization problem. Adapted from Papadimitriou & Steiglitz (1998).

$$\text{minimize} \quad c'x \tag{2.4}$$

$$\text{subject to} \quad Ax \leq b \tag{2.5}$$

$$x \geq 0 \tag{2.6}$$

$$x \text{ is integer} \tag{2.7}$$

Integer programming can assume different complex forms, depending on the specifications of the problem, which can result in a *Mixed-Integer Programming* (MIP) formulation, i.e. x_i indicates if company's i stock is included in a portfolio ($x_i = 1$) or not ($x_i = 0$) and w_i is the proportion of company i stock in a portfolio (if included). The Mixed-Integer Programming structure is represented by the following optimization problem (FLOUDAS, 1995):

$$\text{minimize} \quad f(x, y) \tag{2.8}$$

$$\text{subject to} \quad \mathbf{h}(x, y) \leq b \tag{2.9}$$

$$x \in X \subset \mathbb{R} \tag{2.10}$$

$$y \text{ is integer} \tag{2.11}$$

Where x is a vector of n continuous variables, y is a vector of k integer variables, $f(x, y)$ is the objective function and $\mathbf{h}(x, y)$ are m inequality constraints. Some problems of this type cannot be solved in a reasonable time. Before defining what is an "intractable problem", some other concepts must be presented.

2.1.1 Polynomial time algorithms

A problem is defined by (GAREY; JOHNSON, 1979):

1. A general description of all its parameters
2. A statement of what properties the answer (solution) is needed to satisfy

Consider the MIP formulation as an example on how to define a problem. A general instance of this problem is given by vectors $c \in \mathbb{Q}^{n+k}$, $b \in \mathbb{Q}^m$ and a matrix of constraints coefficients $A \in \mathbb{Q}^{m \times (n+k)}$. A solution to this problem is a concatenated vector of float and integer-valued vectors $[x_1, x_2, ..., x_n, y_1, y_2, ..., y_k]$, that gives the minimum value of f, subject to $\mathbf{h}$. An algorithm is said to solve a problem if this algorithm can be applied to any instance of this problem and produce a solution for that instance. The most efficient algorithm is the fastest to solve a problem (GAREY; JOHNSON, 1979).

The time requirements of an algorithm depend on the "size" of a problem instance. Thus, in order to define time requirements in a more precise manner, it is necessary to define the size of an instance in a formal mathematical way (GAREY; JOHNSON, 1979). Other concepts related to this formalization are defined below (GRÖTSCHEL; LOVáSZ; SCHRIJVER, 1988):

- **Alphabet**: A finite set of symbols, i.e. binary encoding: {0, 1}
- **Encoding scheme**: Maps problem instances into finite strings using an alphabet. The length of these strings, named encoding length, is the size of the problem.

A **time complexity function** $f : \mathbb{N} \rightarrow \mathbb{N}$ models the maximum time $f(n)$, measured in steps, required by an algorithm to solve any problem instance size of at most $n \in \mathbb{N}$. A **polynomial time algorithm** is an algorithm whose time complexity function $f(n)$

satisfies $f(n) \leq p(n)$, for all $n \in \mathbb{N}$ and a polynomial function p (GRÖTSCHEL; LOVáSZ; SCHRIJVER, 1988).

2.1.2 The classes P and NP

It is convenient to analyse an optimization problem as a decision problem, so that we can distinguish between "easy" and "hard" problems. A **decision problem** is one that has only two possible solutions: "yes" or "no". An example of the decision version of an integer programming optimization problem is: Given a $m \times n$ matrix A of integer numbers, integer vectors $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ and an integer B, is there a vector $x \geq 0$, $x \in \mathbb{Z}^n$, such that $Ax \leq b$ and $cx \leq B$? (RESENDE; RIBEIRO, 2016).

The decision version of an optimization problem cannot be harder than the optimization version. Thus, if a decision problem cannot be solved in polynomial time, then its optimization version cannot be solved in polynomial time (RESENDE; RIBEIRO, 2016). Considering this fact, a special class of decision problems is defined (GRÖTSCHEL; LOVáSZ; SCHRIJVER, 1988; RESENDE; RIBEIRO, 2016):

- **Class P**: The class of decision problems that can be solved in polynomial time. It is the class of "easy" decision problems.

Some examples of problems solved in polynomial time are the shortest path problem and the minimum spanning tree problem (RESENDE; RIBEIRO, 2016). To identify a more extensive class of decision problems, some concepts will be defined. Given a decision problem $\mathcal{P}$ and a "yes" instance $\mathcal{I}$, a certificate $c(\mathcal{I})$ is a string that encodes a solution and proves that the answer for this instance $\mathcal{I}$ is affirmative (RESENDE; RIBEIRO, 2016). The class NP is defined as follows (RESENDE; RIBEIRO, 2016):

- **Class NP**: A decision problem $\mathcal{P}$ belongs to this class if there exists a certificate-checking algorithm, that can prove the answer "yes" for any certificate and associated "yes" instance $\mathcal{I}$ of $\mathcal{P}$ in polynomial time. The certificate-checking algorithm can be a combination of a recognition algorithm (feasibility checking) with the cost function algorithm (computes cost function).

Now NP-complete and NP-hard problems will be defined (RESENDE; RIBEIRO, 2016):

- **Polynomial-time reduction**: Consider two decision problems $\mathcal{P}_1$ and $\mathcal{P}_2$. There is a polynomial-time reduction from $\mathcal{P}_1$ to $\mathcal{P}_2$ iff the first can be solved by an algorithm $\mathcal{A}_1$ that is equivalent to a polynomial number of calls to solve $\mathcal{P}_2$ by an algorithm $\mathcal{A}_2$

- **Polynomial-time transformation**: A special case of a polynomial-time reduction. There is a polynomial-time transformation from $\mathcal{P}_1$ to $\mathcal{P}_2$ if an instance of $\mathcal{P}_2$ can be constructed in polynomial time from any instance of $\mathcal{P}_1$, such that the instance of $\mathcal{P}_1$ is a "yes" instance *iff* the instance of $\mathcal{P}_2$ is a "yes" instance.

- **NP-complete**: A problem $\mathcal{P} \in$ NP is NP-complete if any other problem in NP can be transformed to $\mathcal{P}$ in polynomial time. Then, if there is a polynomial-time algorithm for $\mathcal{P}$, there are also polynomial-time algorithms for all other problems in NP.

- **NP-hard**: A problem $\mathcal{P}$ is NP-hard if any other problem in NP can be transformed to $\mathcal{P}$ in polynomial time, but P is not proved to be in NP. Optimization problems whose decision versions are NP-Complete are NP-hard because they are not in NP (they are not decision problems). An NP-hard problem is at least as hard as an NP-complete problem.

2.1.3 Solution approaches to intractable problems

The majority of combinatorial optimization problems are NP-complete and NP-hard. The fact that these problems are intractable does not except the necessity of finding a solution to them. Some solution approaches are described below (PAPADIMITRIOU; STEIGLITZ, 1998; MARTI; PARDALOS; RESENDE, 2018; RESENDE; RIBEIRO, 2016; BURIOL et al., 2020, no prelo):

- **Exact algorithms**: Routinely applied to solve large instances of combinatorial problems in an accessible CPU time, i.e. *branch-and-bound* (B&B). B&B enumerates all the feasible points of a combinatorial optimization problem using binary trees. Fortunately, not all the feasible solutions are going to be reached due to the *bound* mechanism, so that only the most promising *leaves/nodes* are *branched* to generate other children. A more detailed algorithm is presented in Section 5.2. When a feasible integer solution is found, a *bound* will occur and this node is designated as *fathomed*. Considering the minimization case, if this *fathomed* is better than the current *incumbent* solution (the best feasible integer solution of the search tree), then it becames the current *incumbent* solution and its objective function value becomes the new *upper bound*, otherwise it is killed and no *incumbent* update is performed. A *Lower bound/best bound* is the minimum of the objective function values of the linear programming relaxation of all open search tree nodes. Using the *Lower bound* value, it is possible to construct a proof of optimality without exhaustive search. Considering the upper U and lower L bounds, the ratio in which a feasible integer solution is within the optimal integer solution is calculated as follows: $GAP = \frac{U-L}{L}$. When the GAP value is zero, then the optimality has been proved.

- **Heuristics**: Any algorithm that provides a feasible solution without quality and computational time guarantees. Heuristics are classified as constructive, local search or metaheuristic.

2.2 PORTFOLIO OPTIMIZATION

Traditionally, the problem of portfolio selection, which possess a central role in financial management, involves the computation of proportions of capital that must be allocated in a set of available assets with the objective of maximizing return and minimizing risk of an investment portfolio. For this problem, the rule of expected return-variance or E-V rule provides efficient and, often, diversified portfolios (MARKOWITZ, 1952). Given that p_{it} is the price of asset i a time t, where $t \in \{1, ..., T\}$, the return of asset i after d periods is defined as $r_{it} = \frac{p_{it} - p_{i(t-d)}}{p_{i(t-d)}}$.

Portfolio selection problems that give optimal utility based on expected return and variance, as proposed by Markowitz (1952), are classified as mean-variance optimization problems (MVO), a case of quadratic optimization problem (KOLM; TUETUENCUE; FABOZZI, 2014). The risk aversion formulation of the classical MVO is presented below.

$$\text{minimize} \quad w^T \Omega w - \lambda w^T \mu \tag{2.12}$$

$$\text{subject to} \quad \sum_{i=1}^{N} w_i = 1 \tag{2.13}$$

$$w_i \geq 0 \tag{2.14}$$

Equation (2.12) is the objective function and it expresses preferences of the decision-maker relative to risk and return. λ is the risk aversion factor, which reflects the investor's objectives, ranging from 0 (risk-averse investor) to 10 (highly tilted toward higher returns) (FABOZZI et al., 2007), w_i represents stock's i proportion in the portfolio, Ω is the covariance matrix of the returns of assets composing the index, μ_i is the expected return of stock i and N is the number of assets in the universe. (2.13) is the budget constraint. In this model, non-dominated solutions are generated and a set of portfolios, named efficient frontier, is formed. If one chooses a solution contained in the set of non-dominated portfolios, one cannot change to another solution contained in this set without deteriorating one of the objectives (BEASLEY, 2013; RESENDE; RIBEIRO, 2016) (return or risk). Then, if one wants to improve return, it must deteriorate risk and vice-versa. A generic plot of this frontier is illustrated in Figure 3, where the horizontal and vertical axis represents the standard deviation and the expected return of the portfolio, respectively.

Investors' attitude towards risk and return is used as an indicator for the choice of a portfolio belonging to this frontier. If the investor is more tilted to high return, then it may select an asset that produces the highest expected return possible, otherwise, it may diversify his/her portfolio to mitigate risk or, if there is full risk aversion, choose an asset that contains the minimum standard deviation possible (FABOZZI et al., 2007).

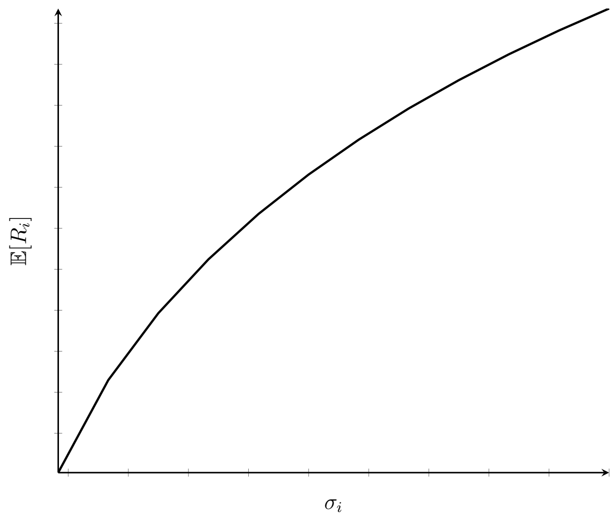

Figure 3 – The efficient frontier

2.2.1 Practical Portfolio Optimization

It is possible to modify or extend this classical framework, incorporating, for example, additional criteria or constraints, such that it becomes more realistic, reflecting the context and objectives of the financial agent, and also to obtain a more diversified portfolio. Extensions inserted in the model can include lot sizing, transaction costs, portfolio cardinality, various types of constraints reflecting specific characteristics of the investment, financial agent or country involved; alternative risk measures or modeling and quantification of the impact of wrong estimates of risk and return (FABOZZI et al., 2007; KOLM; TUETUENCUE; FABOZZI, 2014; BEASLEY, 2013).

2.2.1.1 Alternative risk measures

The advances of research on portfolio management include the development of new risk measures. Artzner et al. (1999) defined properties that a measure of risk ρ must satisfy to be considered coherent:

1. Monotonicity: Consider two asset returns X and Y, which are random variables. If $X \geq Y$, then $\rho(X) \leq \rho(Y)$. Fabozzi et al. (2007) shows another way of seeing this property. If $X \geq 0$, then $\rho(X) \leq 0$. In other words, if there are only positive returns, then the risk should be non-positive.

2. Subadditivity: Merging assets in a portfolio do not create extra risk. $\rho(X+Y) \leq \rho(X) + \rho(Y)$

3. Positive homogeneity: For any positive real number λ, $\rho(\lambda X) = \lambda\rho(X)$. Portfolio size influences risk. In other words, large portfolio positions implies that their required liquidation time will also be large.

4. Translational invariance: For any real number α, $\rho(X + \alpha r) = \rho(X) - \alpha$. The main function of a risk measure is to rank risks. Therefore, inclusion of cash or any risk free asset does not contribute to portfolio risk,

Fabozzi et al. (2007) categorizes risk measures in two classes: Dispersion and Downside. Dispersion measures are measures of uncertainty that equally penalize overperformance and underperformance relative to the mean. Some types of these measures are listed below:

- **Variance of return**: Incorporated in the works of Markowitz (1952) and one of the most known dispersion measures.

$$VAR(R_p) = E\Bigg[\Big(\sum_{i=1}^{N} w_i R_i - E\Big[\sum_{i=1}^{N} w_i R_i\Big]\Big)^2\Bigg] \tag{2.15}$$

- ***Mean Absolute Deviation*** **(MAD)**: Introduced by KONNO & YAMAZAKI (1991). Uses absolute deviations in place of squared deviations. It is a dispersion measure based on the absolute deviations from the mean. The resulting optimization problem is linear, which is more solver-friendly than the MVO problem.

$$MAD(R_p) = E\Bigg[\Bigg|\sum_{i=1}^{N} w_i R_i - \sum_{i=1}^{N} w_i \mu_i\Bigg|\Bigg] \tag{2.16}$$

The resulting optimization problem is:

$$\text{minimize} \quad \frac{\sum_{t=1}^{T} y_t}{T} \tag{2.17}$$

$$\text{subject to} \quad y_t + \sum_{i=1}^{N} a_{it} w_i \geq 0 \tag{2.18}$$

$$y_t - \sum_{i=1}^{N} a_{it} w_i \geq 0 \tag{2.19}$$

$$\sum_{i=1}^{N} r_i w_i \geq \rho \tag{2.20}$$

$$\sum_{i=1}^{N} w_i = 1 \tag{2.21}$$

$$w_i \geq 0 \tag{2.22}$$

Where r_{it} is the realization of the random variable R_i in period $t \in (1, ..., T)$, $r_i = E[R_i]$ and $a_{it} = r_{it} - r_i$.

A model that uses a downside risk measure aims to maximize the probability of satisfying a certain return threshold.

- ***Semi-Variance*** **(SV)**: Variance equally penalizes overperformance and underperformance. Markowitz (1959) presented this measure as an increment to its original variance measure of risk.

- ***Value-at-Risk*** **(VaR)**: Measures the predicted maximum loss at a specified probability threshold $1 - \alpha$ (eg. $\alpha = 5\%$).

- ***Conditional Value-at-Risk*** **(CVaR)**: Rockafellar & Uryasev (2000) proposed this measure in order to repair VaR deficiencies. This measure is a coherent risk measure, as it satisfies all the properties established by Artzner et al. (1999). Also, its optimization model is linear and can be solved very efficiently by optimization software.

2.2.1.2 Practical constraints

Depending on the institution or investor context, some constraints can be incorporated to the model in order to reflect practical issues. The optimization output may contain a few large positions and many small positions. This is undesirable due to extra transaction costs (FABOZZI et al., 2007). In this case, threshold/holding/floor-ceiling constraints can be included in the model:

$$\epsilon Z_i \leq w_i \leq \sigma_i Z_i \text{ for each } i \in 1, ..., N \tag{2.23}$$

Where Z_i are binary variables. $Z_i = 1$ if asset i is included in the portfolio and $Z_i = 0$ if i is not included. Equation (2.23) ensures that if an asset i belongs to the portfolio, then its proportion w_i must lie between ϵ_i and σ_i, otherwise, if i is not contained in the portfolio, $w_i = 0$. Investors might want to restrict their portfolio size, for example to build an index tracking portfolio and keep transactions cost low (FABOZZI et al., 2007; SANT'ANNA et al., 2017).

$$\sum_{i=1}^{N} Z_i = K \tag{2.24}$$

Equation (2.24) restrict the number of assets contained in the portfolio to K. Those constraints generate discontinuous efficient frontiers (CHANG et al., 2000). This is because some non-dominated portfolios of the original continuous efficient frontier would not be considered by any rational investor, as Fabozzi et al. (2007) exemplifies, there can be portfolios with less risk and greater returns. Thus, from incorporating these new constraints to MVO, a *Cardinality Constrained Portfolio Optimization* (CCPO) model is assembled and shown below.

$$\text{minimize} \quad w^T\Omega w - \lambda w^T\mu \tag{2.25}$$

$$\text{subject to} \quad \epsilon Z_i \leq w_i \leq \sigma_i Z_i \text{ for each } i \in 1, ..., N \tag{2.26}$$

$$\sum_{i=1}^{N} Z_i = K \tag{2.27}$$

$$\sum_{i=1}^{N} w_i = 1 \tag{2.28}$$

$$Z_i \in \{0, 1\} \tag{2.29}$$

Other practical constraints can be incorporated into this model, i.e. class/sector, lot size, and transaction costs constraints.

2.2.1.3 Active and Passive strategies

Fund management and portfolio can be classified in two broad approaches: (BEASLEY; MEADE; CHANG, 2003; FABOZZI et al., 2007; JORION, 2003; ROLL, 1992):

- **Active management**: Markets are not fully efficient and management teams can work to make the portfolio achieve better performance than the market, using available information and their experience. Incurs high fixed costs, paid to the managers, and high transaction costs, because of frequent trading.
- **Passive management**: It is assumed that markets are efficient, so that market prices fully reflect risk and return. Incurs in lower fixed costs and lower transaction costs, but, if the market falls, the return falls.

Those two strategies are pure, but mixed strategies are possible too. This is the case when a portion is invested passively and the remainder is invested actively. This alternative can be illustrated with a model that aims to minimize *Variance of Tracking-Error* (TEV) conditional to a certain excess return target (ROLL, 1992):

$$\text{minimize} \quad x^T\Omega x \tag{2.30}$$

$$\text{subject to} \quad x^T\mu = G \tag{2.31}$$

$$\sum_{i=1}^{N} x_i = 0 \tag{2.32}$$

$$\tag{2.33}$$

Where $\mathbf{x} = \mathbf{w}_p - \mathbf{w}_b$ is a vector representing the difference between the managed portfolio and the benchmark proportions. G is the manager's expected performance relative to the benchmark.

2.2.2 Index Tracking problems

A passively managed fund is known as *index fund/tracker fund*. A manager that adopts this strategy could buy all the stocks of a given stock index and reproduce it perfectly (full replication), but this strategy has some disadvantages (BEASLEY; MEADE; CHANG, 2003; CANAKGOZ; BEASLEY, 2009; SANT'ANNA et al., 2017):

- The composition of the index is revised periodically. Therefore, the holdings of all stocks will change periodically to reflect the new composition's weights of the index.
- Transaction costs associated with the index's stocks cannot be limited since it is necessary to trade all stocks to reduce tracking error periodically.

The index tracking problem is concerned with index replication, but limiting transaction costs by using fewer stocks. Decisions concerning the maintenance of the tracking portfolio are enclosed in a decision support system. Important components of this system are (GAIVORONSKI; KRYLOV; WIJST, 2005):

- Benchmark or Index to be tracked
- Risk measure relative to deviations from the index (tracking error)
- Rebalancing strategies to reflect price changes in the market in portfolio weights
- Specify trade-off: maximum portfolio size and maximum tracking error allowed
- Decision rules regarding the cash flow generated by the portfolio (i.e. dividends)
- Decision rules regarding changes in the benchmark composition (i.e. merges)

There is a variety of country/world indexes to be tracked and these are provided by firms such as S&P (2019) and B3 (2019), that compute indexes' theoretical weights using their own methodology. Some risk measures relative to the benchmark, which are going to be identified as TE, are presented below (GAIVORONSKI; KRYLOV; WIJST, 2005; RUDOLF; WOLTER; ZIMMERMANN,):

- **MAD relative to an index**: Absolute deviations between the benchmark and portfolio returns are minimized. Implies in a linear model.

$$TE = \frac{1}{T} \sum_{t=1}^{T} \left| R^t - \sum_{i=1}^{N} r_i^t w_i \right| \tag{2.34}$$

  Where R^t is the benchmark return in period $t \in \{1, ..., T\}$.

- **Mean squared error**: Quadratic deviations between the benchmark and portfolio returns are minimized.

$$TE = E\left[\left(R^t - \sum_{i=1}^{N} r_{it}w_i\right)^2\right] \tag{2.35}$$

- **TEV**: Its formulation is presented in Section 2.2.1.3. The associated formulation is a quadratic optimization problem and it requires the benchmark weights.

$$TE = (w_p - w_b)^T \Omega (w_p - w_b) \tag{2.36}$$

  w_p is a vector that represents the portfolio weights to be optimized and $\mathbf{w}_b$ is a vector representing the benchmark proportions

- **VaR relative to an index**: Similar to VaR measure. It is the largest value w by which the portfolio return can miss the index target in $1 - \alpha$ fraction of cases.

$$TE = VaRI_\alpha = \inf_w \{w | P(\mu^T w \geq R_b - w) \geq 1 - \alpha\} \tag{2.37}$$

- **CVaR relative to an index**: It shows the mean deviation relative to the benchmark in the worst α cases.

$$TE = E\Big(R_b - \mu^T w \Big| \mu^T w < R_b - VaRI_\alpha\Big) \tag{2.38}$$

Even though constraint (2.24) is obligatory for this kind of problem, these risk measures can be combined with other practical constraints, such as those presented in Section 2.2.1.2, to reflect the context in which the tracking portfolio of an investor/institution is applied.

2.3 GRASP METAHEURISTIC

GRASP is a metaheuristic procedure that consists of an iterative process in which two phases are considered for each step: the construction phase and the local search phase (FEO; RESENDE, 1995). During the construction phase, a *Restricted Candidate List* (RCL) is formulated and a feasible solution is interactively constructed, one element at a time, according to its benefit, measured by a greedy function (FEO; RESENDE, 1995; FESTA; RESENDE, 2009a). As the solution constructed is not guaranteed to be a local optimum (FESTA; RESENDE, 2009a), this solution can be improved by the use of the local search procedure (FEO; RESENDE, 1995). Since GRASP was first presented in 1995, different GRASP formulations have been proposed. Several studies have proposed adaptations to GRASP, such as using alternative construction mechanisms, path-relinking (backward, forward, truncated, evolutionary, etc), and hybridizations of GRASP (GENDREAU; POTVIN, 2010; FESTA; RESENDE, 2009a).

2.3.1 Basic GRASP theory

Combinatorial optimization problems are characterized by a solution S and a cost function that must be minimized $f(S)$. Solution approaches to this type of problem construct S from scratch, choosing an unpicked element $i \in S_c$ based on a criterion c_i and inserting it on the partial solution. The elements are inserted, one at a time, in the partial solution until S is feasible. The cost function could assume, for example, $f(S) = \sum_{i \in S} c_i$. The criterion c_i can indicate the cost contribution of an element i if it is included in the partial solution S, i.e. $c_i =$ the cost variation in f by including i in S. Some algorithms designed for solution construction of combinatorial optimization problems (minimization case) will be described (RESENDE; RIBEIRO, 2016):

- **Greedy algorithm**: constructs S by choosing elements containing the least cost c_i
- **Adaptive greedy algorithm**: A greedy algorithm, but the cost c_i of an element i is affected by previous choices. Then, a constant cost is replaced by a greedy choice function $g(i)$ that measures the suitability of including i in S. This choice function produces an ordered list from the most suitable to the least suitable element. The greedy algorithm will always choose the most suitable element.
- **Semi-greedy/randomized-greedy algorithm**: The former algorithms choose the best element based on a ordering generated by a criterion (c_i or $g(i)$). The greedy approach is not suitable for all problems since there's a chance that the algorithm fails to construct a feasible solution or that the constructed solution is bad (far from the optimal). To get better-constructed solutions, one can adopt a randomized approach. In this approach, only a subset of the ordered list S_c is used. Then, elements are randomly selected from this subset, named RCL, to increase the chances of constructing a "better" S.

After building a feasible solution by using one of the previous algorithms, it is possible to increment this solution towards a better or the optimal solution. This procedure is called local search. Assume that F is the set of feasible solutions. $N(S) \subseteq F$ is a **neighborhood** of a solution $S \in F$. Each solution $S' \in N(S)$ can be reached by the **move** operator. This operator moves from S to S' by exchanging one or more elements in S. The main phases of a local search method are shown below (RESENDE; RIBEIRO, 2016):

1. **Start**: Build an initial solution using a solution construction algorithm
2. **Neighborhood search**: Apply a secondary search to improve the initial solution
3. **Stop**: Define a criterion to terminate the search. One example of a stopping criterion is when a locally optimal solution S^+ is encountered with respect to neighborhood N.

The locally optimal solution is defined as a solution S^+ with respect to neighborhood N iff $f(S^+) \leq f(S)$, $\forall S \in N(S)$.

Metaheuristics are procedures that coordinate simple heuristics and rules to find good or even optimal solutions for hard optimization problems. The customization of some metaheuristic to a specific problem is denoted as a heuristic to the problem. GRASP is a metaheuristic that uses a multistart framework to incorporate a hybridization of a semi-greedy algorithm with a local search method (RESENDE; RIBEIRO, 2016).

2.3.2 Multistart procedure

A multistart procedure is an algorithm that applies a solution construction algorithm until a maximum number of iterations is reached. Two types can be used (RESENDE; RIBEIRO, 2016):

- **Random multistart**: Uses a semi-greedy algorithm adopting an RCL size equal to the size of the unpicked elements list S_c. Then, all elements are randomly allocated to S, since when $|RCL| = |S_c|$ there is an equal chance to pick any available element.
- **Semi-greedy multistart**: Uses a semi-greedy algorithm with an RCL size less than the size of the unpicked elements list S_c. As $|RCL|$ decreases, the chance to get the most suitable elements increases. In other words as $|RCL|$ decreases, the multistart procedure will perform a more greedy construction.

2.3.3 GRASP procedure

Algorithm 1: GRASP

$f^* \leftarrow \infty$
$S^* \leftarrow \{\}$
while *stopping criterion not satisfied* **do**
    $S \leftarrow$ SEMI-GREEDY
    if S is not feasible **then**
        $S \leftarrow$ Repair(S)
    end if
    $S \leftarrow$ LOCAL-SEARCH(S)
    if $f(S) < f^*$ **then**
        $S^* \leftarrow S$
        $f^* \leftarrow f(S)$
    end if
end while
return S^*

GRASP is the product of embedding solution construction and local search procedures into a multistart procedure. The pseudo-code for this metaheuristic is shown in Algorithm 1 (RESENDE; RIBEIRO, 2016).

SEMI-GREEDY function calls for the semi-greedy procedure described in subsection 2.3.1. LOCAL-SEARCH function calls any implementation of a local search method. As any metaheuristic, GRASP offers a way to overcome some of the main limitations of construction and local search algorithms, such as the sensitivity to the initial solution and tendency to fall in local optima (RESENDE; RIBEIRO, 2016).

2.4 CHAPTER CONCLUSION

In this chapter, the necessary concepts to develop this work were presented. Combinatorial optimization, Portfolio optimization and GRASP were the main topics that build the theoretical foundations to guide the literature reviews and proposed solution method for the index tracking problem.

3 LITERATURE REVIEW

Metaheuristics consist of methods that perform interactions between local search and higher-level strategies, resulting in a process that can escape from local optima and apply a robust search in a solution space (GENDREAU; POTVIN, 2010). They are used for portfolio optimization in situations in which traditional exact models are computationally costly to conduct. This chapter illustrates recent developments of solution approaches for portfolio selection and GRASP.

3.1 RECENT DEVELOPMENTS IN SOLUTION APPROACHES FOR CCPO

An investigation concerning the current solution approaches to the CCPO problem was developed and is presented in this section. The majority of the solution approaches for portfolio optimization are focused on solving MVO deterministic models with approximate algorithms (KALAYCI; ERTENLICE; AKBAY, 2019). Ertenlice & Kalayci (2018) present a survey on algorithms and applications of *Swarm Intelligence* (SI) for portfolio optimization. In their survey, they found that *Particle Swarm Optimization* (PSO) was the most used kind of SI algorithm.

Although most solution approaches are concentrated in some metaheuristics, researchers continue to explore and propose different heuristics to deal with specific characteristics of practical financial portfolio optimization models, such as non-convexity generated by practical constraints. Taking into account the possible inconsistencies expected under Utility Theory, Gong et al. (2018) proposed a *Adaptive Real-coded Genetic Algorithm* (ARCGA) to deal with a portfolio selection problem under the Cumulative Prospect Theory. Fairbrother, Turner & Wallace (2018) proposed a problem-driven scenario-generation approach to the single-period portfolio selection problem involving tail risk measures. The author presented a new heuristic based on the *Sample Average Approximation* (SAA) method. Bacevic et al. (2019) proposed a Variable Neighborhood Search heuristic to deal with constraints that introduce nonconvexity to a cardinality constrained portfolio optimization problem. The results show that this method can be applied to large-scale data sets. Salehpoor & Molla-Alizadeh-Zavardehi (2019) combined a diversification mechanism with many evolutionary algorithms. The metaheuristic algorithms were implemented and tested in four cardinality constrained models: MVO, MAD, SV and variance with skewness. *Simulated Annealing* (SA) had the best overall performance. Boudt & Wan (2020) developed a binary PSO algorithm and documented the performance of this algorithm parameters when the cardinality of the portfolio is small compared to the universe of assets.

Besides that finding the true CCPO efficient frontier becomes impractical when the size of the asset universe increases, since it is a NP-hard problem (FABOZZI et al., 2007), the manager may want to include more objectives in the model. Thus, the CCPO frontier can be computed iteratively by transforming the single-objective optimization problem into a multi-objective optimization problem. Different configurations of *Multi-objective Evolutionary Algorithm* (MOEA) and the impact of real-world portfolio selection constraints are tested in Liagkouras & Metaxiotis (2018a). It includes a discussion of the budget, floor/ceiling, cardinality, class, turnover and transaction constraints. Meghwani & Thakur (2018) applied evolutionary algorithms to address a multi-objective approach for a multi-period problem with practical constraints, which had three objectives: to minimize risk and transaction costs and to maximize returns. An investigation into whether the 2-Phase *Nondominated sorting genetic algorithm II* (NSGA-II) outperforms NSGA-II is made by Eftekharian, Shojafar & Shamshirband (2017), indicating a better performance of 2-Phase NSGA-II according to some of the metrics tested. Silva, Herthel & Subramanian (2019) brings a unified multi-objective PSO approach to solve the MVO constrained with cardinality, round-lot, floor-ceiling and pre-assingment constraints. Babazadeh & Esfahanipour (2019) developed a new repairing mechanism for NSGA-II applied to a MVO model subject to a set of practical constraints, using a real data set from the S&P100 index. Liagkouras (2019) designed a MOEA for large-scale multi-objective portfolio optimization problems, evaluated in five alternative formulations using real data. Zhou et al. (2019) presented a novel MOEA named Mucard algorithm, which enables knowledge reuse and exchange among genetic individuals applied in multi-scenario CCPO. Akbay, Kalayci & Polat (2020) (parallel variable neighborhood search) and developed Kalayci, Polat & Akbay (2020) (combined components of artificial bee colony, ant colony optimization and genetic algorithms) hybrid search algorithms to compute efficient frontiers. The algorithms achieved a competitive performance compared with state-of-the-art solution approaches, but couldn't outperform some of these approaches in certain instances, such as Baykasoglu, Yunusoglu & Ozsoydan (2015) GRASP-QUAD algorithm.

Fuzzy set theory is being adopted by researchers as a way to represent the uncertainty of model parameters. Jalota & Thakur (2018), proposed a new algorithm named BEXPM-RM and considered fuzzy returns in the model. Fuzzy multi-period portfolio optimization problems, considering transaction costs, were discussed and solved using evolutionary algorithms in Liu, Zhang & Zhao (2018) and Liagkouras & Metaxiotis (2018b). Liagkouras & Metaxiotis (2019) proposed an information-based evolutionary algorithm to solve a multi-period fuzzy model with cardinality and liquidity constraints. A fuzzy formulation is also developed by Dutta et al. (2018), who implemented a *Genetic Algorithm* (GA) to solve a portfolio problem by using a stochastic price scenario. A fuzzy portfolio model for international investments is discussed in Liu, Zhang & Gupta (2018). Rangel-Gonzalez et al. (2020) performed a fuzzy multi-objective particle swarm optimization using three criteria and proposed a mechanism to automatically adjust the fuzzy rules.

Recent research also investigates the performance of solution approaches considering the uncertainty representation of multi-objective model parameters. Solares et al. (2019) presented a novel application of *Multi-objective Evolutionary Algorithm based on Decomposition* (MOEA/D) applied in the context of representing uncertainty with interval analysis. This approach presented robust performance even in the 2008 crisis period. Chen, Li & Liu (2019) developed a hybrid algorithm of a *Imperialist competitive algorithm* (ICA) and a *Firefly Algorithm* (FA) to solve the transformation of a cardinality constrained multiperiod multi-objective uncertain portfolio model into a single-objective problem. The authors considered that stock returns are uncertain variables and are subjectively determined by experts.

3.2 SOLUTION APPROACHES FOR INDEX TRACKING IN THE LAST DECADE

The literature covers applications of heuristic/metaheuristic and non-heuristic for solving these types of problems. The rest of this section focuses on a comprehensive literature review concerning these applications on index tracking problems.

3.2.1 Methodology

The procedures and rules that conducted the systematic literature review were based on Kalayci, Ertenlice & Akbay (2019). The authors presented a comprehensive literature review about solution approaches for the portfolio selection problem guided by research questions.

3.2.1.1 Research questions set up

In order to guide the result analysis and scope of this work, some research questions were defined. The research questions are presented in Table 18

3.2.1.2 Material collection and selection

The search was conducted by covering papers published from 2010 to 2019. The bibliography collection process took place at the Web of Science database using the keyword structure presented in Table 2. This keyword combination structure was built from the author's a priori knowledge of the field. Levels 1 and 2 guides the search for financial portfolio optimization problems. Level 3 restricts the financial portfolio problem class to index tracking.

The filtering process is shown in Figure 4. The first filter was used to select only articles that were written in English language. The final filter was performed by fully reading the papers and selecting those that fitted in the scope of this work. Many of the out of the scope works appeared because of the first and second levels keywords. A total of 53 articles were selected after applying all the filters. Some research questions needed analysis

RQ	Description
#1	Are index tracking solution methods more relevant to journals focusing on operations research and computer science?
#2	Are there specific heuristic methods applied to specific quantitative modelling frameworks?
#3	Has there been a growth in the number of non-heuristic methods applied to the index tracking problem?
#4	Has there been a growth in the number of heuristics/metaheuristics applied to the index tracking problem?
#5	Are heuristic approaches more used than non-heuristic approaches for index tracking problems?
#6	Do heuristic approaches have more cite impact than non-heuristic approaches for index tracking problems?
#7	Is there a prevalence of using a specific heuristic/metaheuristic in index tracking problems?
#8	Is there an integration between heuristic and general purpose solvers?
#9	Is there a prevalence of using specific evaluation metrics for heuristic approaches?
#10	Is there a prevalence of using a specific solution method to compare with heuristic approaches?
#11	Is there a prevalence of solving for a specific tracking error objective function when using heuristic approaches?
#12	Is there a prevalence of solving for specific practical constraints when using heuristic approaches?
#13	Which databases were most adopted in heuristic approaches?

Table 1 – Research questions for index tracking systematic literature review

Level	Search Terms
1	Portfolio OR Investment OR Asset
	AND
2	Optimization OR Management OR Selection
	AND
3	"Index Tracking" OR "Tracking Error" OR Tracking-Error

Table 2 – Proposed keyword combination structure for the index tracking systematic review

of bibliometric tools. Bibliometrix (ARIA; CUCCURULLO, 2017) and Web Of Science were adopted to perform these analyses.

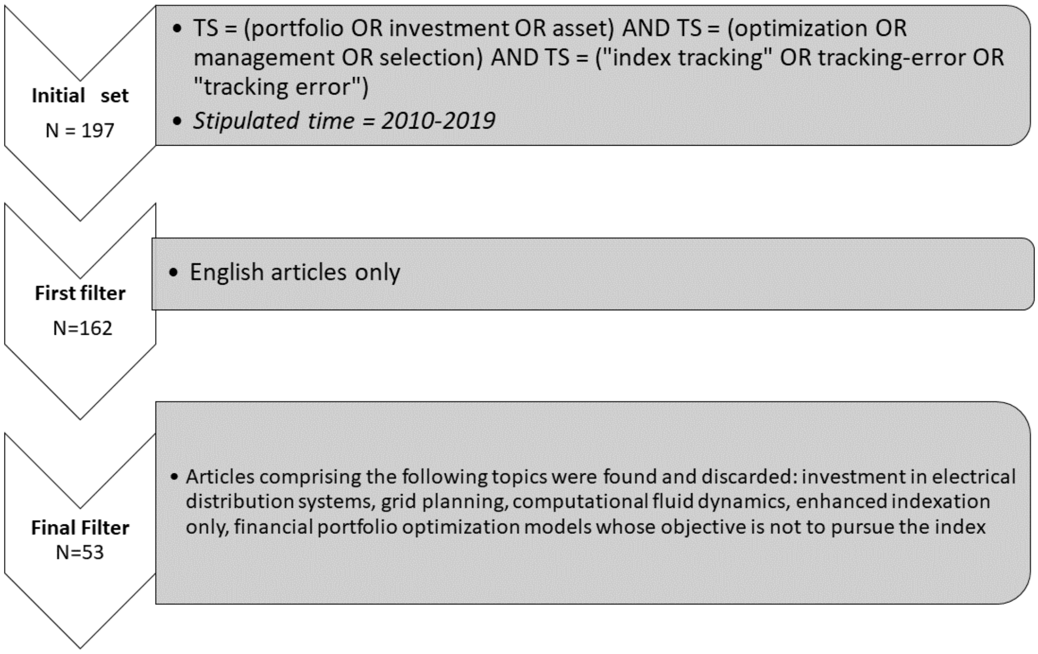

Figure 4 – Filtering process of the index tracking literature review

3.2.2 Results and discussion of the index tracking systematic review

3.2.2.1 Most relevant sources for index tracking applications

The answer to RQ#1 was constructed by considering scientific journals that are responsible for publications involving index tracking applications. Table 3 presents the percentage of sources according to their associated number of publications.

Publications	Number of journals	(%)Journals	(%) Accumulated
1	28	75.68	75.68
2	4	10.81	86.49
3	3	8.12	94.61
4	2	5.39	100.00
Total	37	100	

Table 3 – Percentage of sources according to their publication number

The majority of article sources (28), which represents 75.68% of the total sources, are responsible for only one publication. This result shows that, despite being new literature, where the first research papers developed by Consiglio & Zenios (2001) and Konno & Wijayanayke (2001), index tracking papers are scattered in a wide range area spectrum, such as economy, statistics, computer science and operations research. that investigate this kind of model. the wide. A more specific analysis was developed by considering 9 journals that had published more than one article (14.62% of the total sources) in the considered period. The total number of articles published on the selected sources is equal to 25. Table 4 shows the name of the source, the number of publications, the relative percentage of publications and the associated Web of Science research category.

Name	Publications	% of 25	Categories
European Journal of Operational Research	4	16.00	Operations Research and Management
Quantitative Finance	4	16.00	Business & Finance, Economics , Mathematics and Mathematical Methods in Social Sciences
Annals of Operations Research	3	12.00	Operations Research
Computers & Operations Research	3	12.00	Computer Science, Engineering Industrial and Operations Research
Journal of the Operational Research Society	3	12.00	Management and Operations Research
Applied Soft Computing	2	8.00	Computer Science and Artificial Intelligence
Computational Statistics & Data Analysis	2	8.00	Computer Science and Mathematics
Journal of Economic Dynamics & Control	2	8.00	Economics
Optimization Letters	2	8.00	Operations Research and Applied Mathematics

Table 4 – Selected sources

The following Web of Science research categories were reached: Operations Research, Management, Business & Finance, Economics, Mathematics, Mathematical Methods in Social Sciences, Computer Science, Engineering Industrial, Artificial Intelligence and Applied Mathematics. Table 5 contains the number of journals and publications associated with each of the categories that were found. The most relevant journals are those covering operations research and computer science research areas.

3.2.2.2 Categorization - Quantitative modeling frameworks and solution approaches

To answer RQs #2, #3, #4, #5 and #6 it was necessary to categorize the modeling framework first and then to categorize the solution approach. Three categories of quantitative modeling frameworks were defined: mathematical programming frameworks, statistical techniques frameworks, and other frameworks. Table 6 shows the publications associated with each category.

Figure 5 depicts the relative percentage of published articles per quantitative modeling framework. It can be observed that the majority of works on the index tracking problem are

Category	Number of related journals	Number of related publications
Operations Research	5	15
Computer Science	3	7
Management	2	7
Economics	2	6
Mathematics	2	4
Business & Finance	1	4
Mathematical Methods in Social Sciences	1	4
Engineering Industrial	1	3
Artificial Intelligence	1	2
Applied Mathematics	1	2

Table 5 – The relevant categories

exploring more mathematical programming frameworks than any other type of modeling framework for this problem. The total number of articles allocated to the heuristic group and non-heuristic is 21 and 32, respectively. Thus, the majority of works on index tracking adopted non-heuristic solution methods.

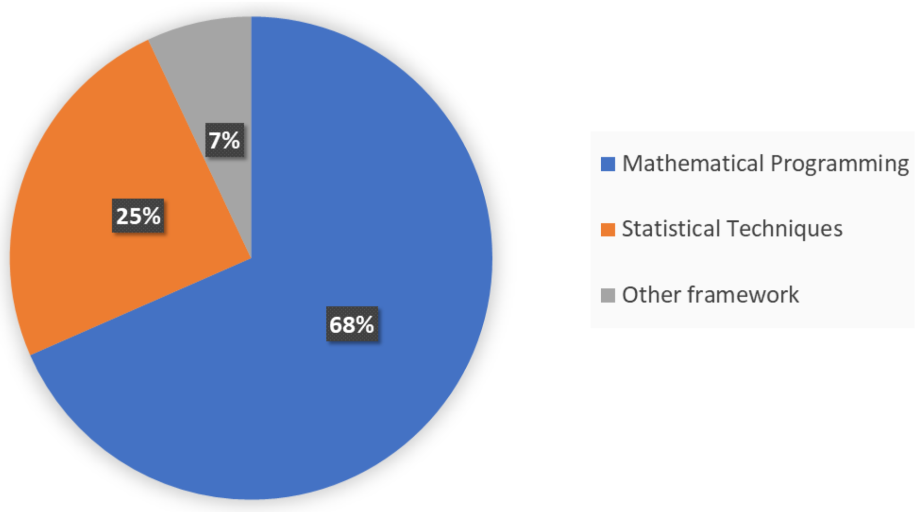

Figure 5 – Percentage of works that used each modelling framework

Framework	Modelling technique	Publications
Mathematical Programming	Linear Programming	(GOEL; SHARMA; MEHRA, 2018); (KUSIAK, 2013); (SHARMA; UTZ; MEHRA, 2017); (THEOBALD; YALLUP, 2010);
	Mixed-integer linear programming	(CHEN; KWON, 2012); (GOEL; SHARMA; MEHRA, 2018); (GUASTAROBA; SPERANZA, 2012); (HUANG; LI; YAO, 2018); (MEZALI; BEASLEY, 2013); (MEZALI; BEASLEY, 2014); (WANG et al., 2012); (WU; KWON; COSTA, 2017)
	Mixed-integer Non-linear programming	(ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (GARCIA; GUIJARRO; OLIVER, 2018); (FASTRICH; PATERLINI; WINKER, 2014); (GRISHINA; LUCAS; DATE, 2017); (HUANG; LI; YAO, 2018); (MUTUNGE; HAUGLAND, 2018); (NI; WANG, 2013); (SANT'ANNA et al., 2017); (SANT'ANNA; FILOMENA; CALDEIRA, 2017); (SANT'ANNA et al., 2019); (SCOZZARI et al., 2013); (STRUB; TRAUTMANN, 2019); (VALLE; MEADE; BEASLEY, 2015); (WANG; XU; DAI, 2018); (XU; LU; XU, 2016)
	Conic Programming	(LING; SUN; YANG, 2014); (SHARMA; UTZ; MEHRA, 2017)
	Mixed-integer Conic Programming	(WU; WU, 2019)
	Stochastic programming	(BARRO; CANESTRELLI, 2014); (BARRO; CANESTRELLI; CONSIGLI, 2019); (STOYAN; KWON, 2010)
	Dynamic programming	(CHENG; CHEN; LIU, 2013); (STOYAN; KWON, 2010)
	Multi-Objective Optimization	(BILBAO-TEROL; ARENAS-PARRA; CANAL-FERNANDEZ, 2012);, (CHIAM; TAN; MAMUN, 2013); (GARCIA; GUIJARRO; MOYA, 2011); (GARCIA; GUIJARRO; MOYA, 2013); (LI; BAO; ZHANG, 2014); (NI; WANG, 2013); (WU; TSAI, 2014)
Statistical Techniques	Cointegration	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (PAPANTONIS, 2016); (SANT'ANNA; FILOMENA; CALDEIRA, 2017); (SANT'ANNA et al., 2019)
	Regression with regularization	(BENIDIS; FENG; PALOMAR, 2018); (FASTRICH; PATERLINI; WINKER, 2014) (GIUZIO; FERRARI; PATERLINI, 2016); (GIUZIO, 2017); (TAS; TURKAN, 2018); (XU; LU; XU, 2016)
	Regression with regularization and variable selection	(WU; YANG, 2014); (WU; YANG; LIU, 2014); (YANG; WU, 2016); (ZHAO; LIAN, 2016)
Other framework	Machine Learning	(OUYANG; ZHANG; YAN, 2019); (NAKAYAMA; YOKOUCHI, 2018)
	Sampling	(DJOKO; TILLE, 2015)
	Pure heuristics	(AFFOLTER et al., 2016)

Table 6 – Publications associated with each framework category

3.2.2.3 Comparison of production and impact of the solution approaches

After associating each article with its solution approach, an analysis of production growth and citation impact was developed. Figure 6 shows the total publications per year for each solution approach. Publications adopting non-heuristic methods had an annual growth rate of 12.98%. Publications using heuristic/metaheuristic solution approaches had an annual growth rate of 9.05%. Since there has been a growth in the number of non-heuristic and heuristic methods applied to the index tracking problem, the answer for both RQ #3 and #4 is positive. The answer to RQ #5 is negative since the production rate of papers adopting non-heuristic approaches is bigger than that of papers adopting heuristic approaches.

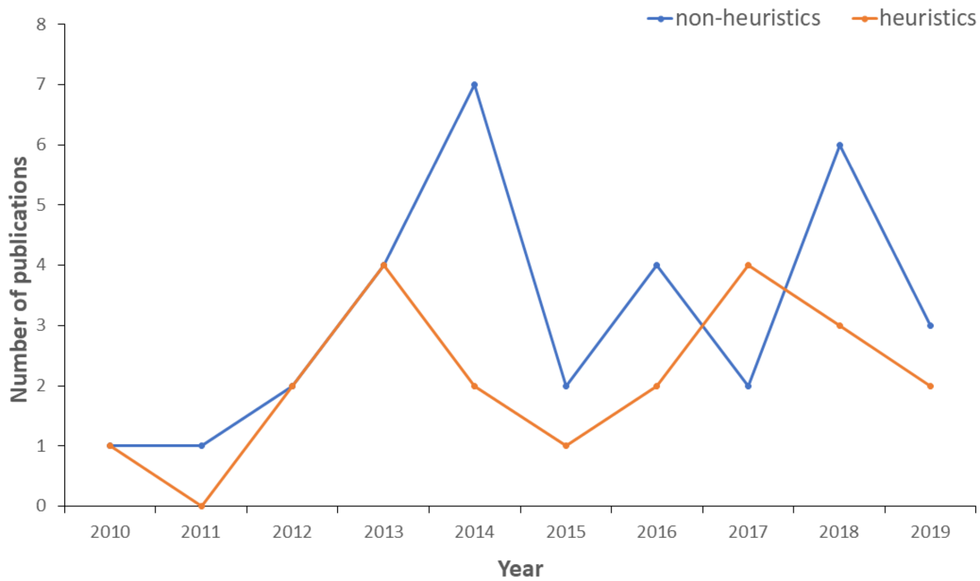

Figure 6 – Comparison of annual total publications for papers using heuristic and non-heuristic approaches

Three metrics were evaluated to answer RQ #6: total citations in the last decade, annual mean total citations and annual mean total citations per article. Figure 7 show the annual total mean citations for each solution approach and Figure 8 shows the annual mean total citations per article. It is obvious that the mean total citations tend to fall over the years because the number of citable years decreases.

Papers that adopted heuristic approaches obtained a total of 216 citations over the last decade. The non-heuristic group obtained a total of 160 citations. Yearly mean total citations of heuristic solution approaches overcome that of non-heuristic solution approaches in most years. This result is also observed for the annual total mean citations per article metric. The answer to RQ #6 is positive because the heuristic group surpassed the non-heuristic group in all three metrics. An interesting observation concerning the production and citation impact result is that although the production of heuristic papers is lower, their impact is higher.

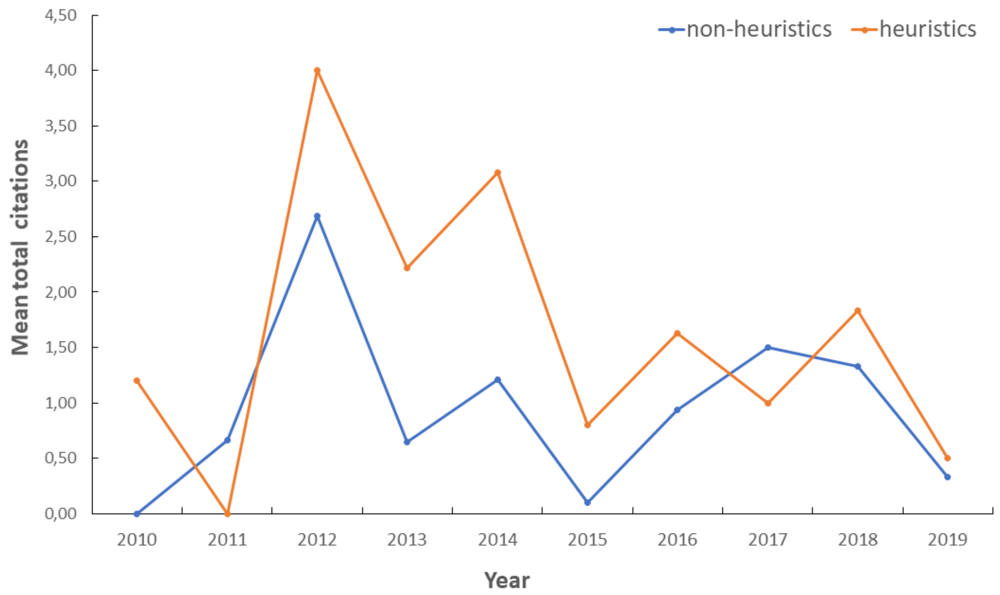

Figure 7 – Comparison of annual total mean citations for heuristic and non-heuristic approaches

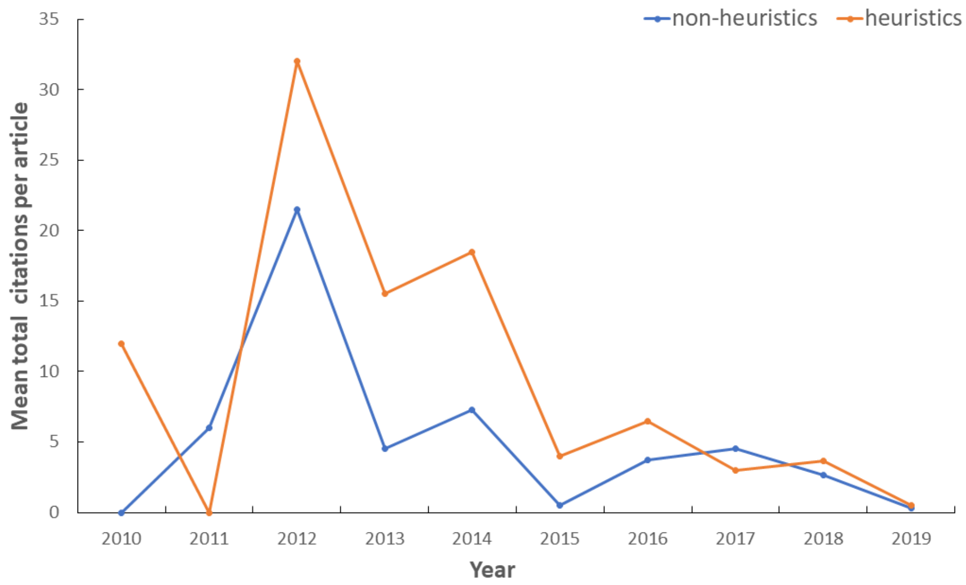

Figure 8 – Comparison of annual total mean citations per article for heuristic and non-heuristic approaches

3.2.2.4 Categorization - adopted heuristics/metaheuristics

An exploratory work was made to find all heuristics used in each of the 21 articles to answer RQs #2 and #7. Also, an investigation concerning the hybridization of the heuristics with general-purpose solvers was performed to answer RQ #8. The heuristics found were: Genetic Algorithm (GA), Differential Evolution (DE), Tabu Search (TS), Greedy Construction (GC), Best Exchange by one (BEBO), Combinatorial Search (CS), Local Branching (LB), Lagrangian-based (LGR), Kernel Search (KS), Variable Neighborhood Search (VNS), Decomposition Algorithm (DA), Multi-objective Genetic Algorithm (MOGA) and Invasive Weed Optimization (IWO). In Table 7 the allocation result of papers to a heuristic is shown.

	Not Hybrid	Hybrid
GA	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (FASTRICH; PATERLINI; WINKER, 2014); (GARCIA; GUIJARRO; OLIVER, 2018); (GIUZIO, 2017); (GRISHINA; LUCAS; DATE, 2017); (NI; WANG, 2013);	(SANT'ANNA et al., 2017); (WANG et al., 2012); (WANG; XU; DAI, 2018); (XU; LU; XU, 2016); (STRUB; TRAUTMANN, 2019)
DE	(ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (GRISHINA; LUCAS; DATE, 2017);	(SCOZZARI et al., 2013)
TS	(GARCIA; GUIJARRO; OLIVER, 2018);	
GC	(MUTUNGE; HAUGLAND, 2018);	
BEBO	(MUTUNGE; HAUGLAND, 2018);	
CS		(SCOZZARI et al., 2013)
LB		(STRUB; TRAUTMANN, 2019)
LGR	(WU; WU, 2019);	
KS		(GUASTAROBA; SPERANZA, 2012);
VNS		(WU; KWON; COSTA, 2017)
DA		(STOYAN; KWON, 2010)
IWO	(AFFOLTER et al., 2016)	
MOGA	(CHIAM; TAN; MAMUN, 2013);	

Table 7 – Heuristic categorization result

Since there are 11 hybridized heuristics, then the answer to RQ #8 is positive. Solvers are integrated with heuristics mainly to perform the capital allocation. Then, in those cases, heuristics were used to perform asset selection only. Table 8 summarizes the associated quantitative framework subcategory, the number of times that a specific heuristic was applied to a specific model. An interesting result is that all heuristics/metaheuristics applied to *Mixed-Integer Linear Programming* (MILP) are hybridized. Figure 9 presents the relative number of works that performed/did not performed hybridization of heuristics/metaheuristics.

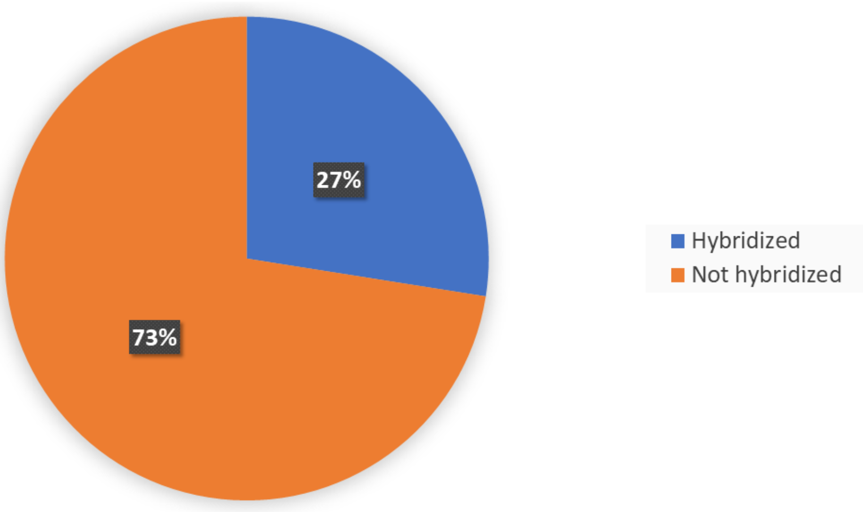

Figure 9 – Percentage of works that performed/did not performed hybridization

The answer to RQ #2 is positive. From Table 8 it can be observed that the vast amount of the developed heuristics/metaheuristics solutions were applied to mathematical programming formulations more often, more specifically to *Mixed-Integer non-linear programming* (MINLP) formulations. Figure 10 shows the relative number of works per heuristics/metaheuristics. The two main heuristics were highlighted and the other heuristics (applied only once) were clustered as a single group.

The answer to RQ #7 is positive. The works that use heuristic/metaheuristic solution approaches are focused on the application of two heuristics: Genetic and Differential Evolution algorithms.

3.2.2.5 Analysis of heuristic/metaheuristic performance evaluations

An heuristics/metaheuristics success is indicated by evaluation metrics. In this subsection, all evaluation metrics used to analyze heuristics in the index tracking problem in the last decade were identified. 34 different metrics were found. Then, it was decided to highlight the most used (applied at least 3 times) metrics among the 34. 9 out of 34 metrics were applied at least 3 times, which were: CPU time, GAP, Correlation With Respect to the Index (CWRTI), Mean Squared Error (MSE), Information Ratio (IR), Root Mean Squared Error (RMSE), Mean Excess Return (MER), Annualized Return (ANNR) and Annualized Tracking Error (ANNTE). Table 9 shows the highlighted metrics and the associated articles.

Framework	Algorithm	N	N_{hybrid}
Mixed-integer Non-linear programming	GA	9	4
	DE	4	1
	TS	1	0
	GC	1	0
	BEBO	1	0
	CS	1	1
	LB	1	1
Mixed-integer linear programming	GA	1	1
	KS	1	1
	VNS	1	1
Mixed-integer Conic programming	LGR	1	0
Stochastic Programming	DA	1	1
Multi-objective Optimization	GA	1	0
	MOGA	1	0
Cointegration	GA	1	0
Regression with regularization	GA	2	0
Pure heuristic	IWO	1	0
Total		29	11

Table 8 – Heuristic applications summary

Figure 11 presents the percentage of works that applied each of the nine highlighted metrics and the other 25 metrics. The answer to RQ #9 is positive since those nine metrics are prevalent among the other 25 metrics.

To answer RQ #10, a search about comparison types was performed. The identified comparison types were: Heuristic against Heuristic, Heuristic against CPLEX, Heuristic against Gurobi, Heuristic against Projected Gradient algorithm, Heuristic against interior point algorithm, Heuristic against Cyclic Coordinate Descent algorithm and Heuristic against Random Selection. Table 10 shows the comparison types and related papers.

Figure 12 presents the percentage of articles that performed each type of comparison. It can be observed that the answer to RQ #10 is positive since most of the authors that developed a heuristic/metaheuristic compared it against another heuristic/metaheuristic or the commercial solver CPLEX.

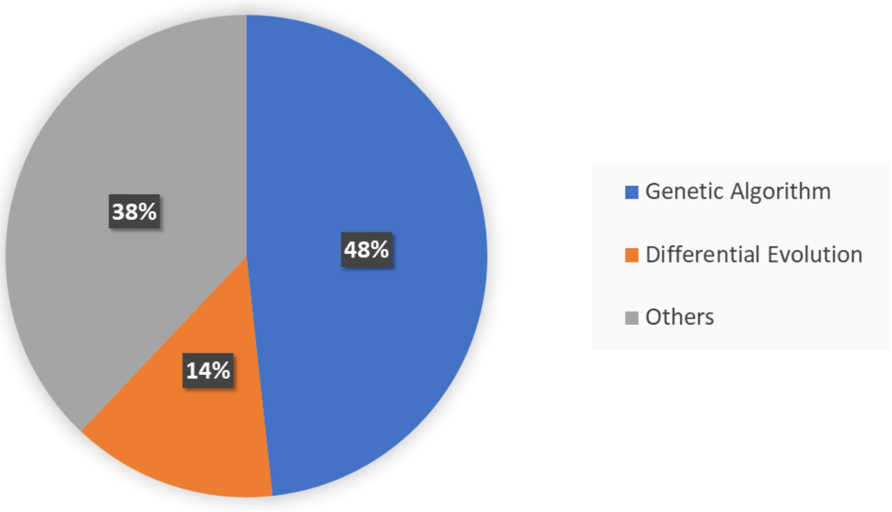

Figure 10 – Percentage of works per heuristic. The two main heuristics were highlighted.

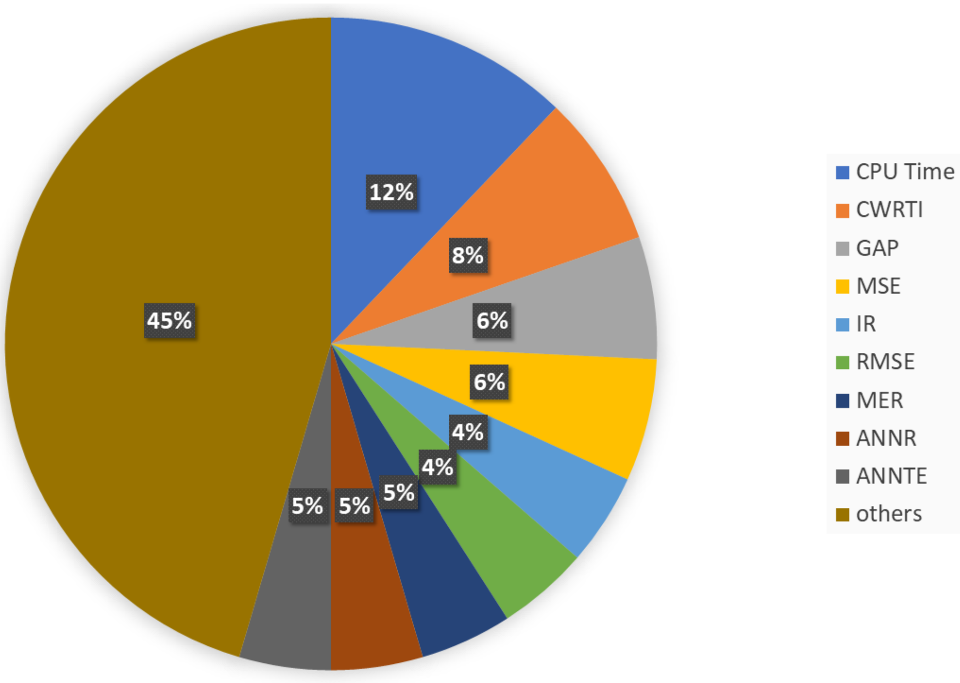

Figure 11 – Percentage of works per evaluation metrics. The nine main evaluation metrics were highlighted.

3.2.2.6 Analysis of index tracking models solved by metaheuristics/heuristics

The objective functions found were: Root Mean Squared Error (RMSE), Mean Squared Error (MSE), Mean Absolute Deviation (MAD), Absolute Deviation (AD), Tracking Error Variance (TEV), Sum of Errors Squares (SES), Correlation, Mean Index Excess Return

Metric	Articles
CPU Time	(GARCIA; GUIJARRO; OLIVER, 2018); (GRISHINA; LUCAS; DATE, 2017); (MUTUNGE; HAUGLAND, 2018); (SANT'ANNA et al., 2017); (STRUB; TRAUTMANN, 2019); (WANG; XU; DAI, 2018); (GUASTAROBA; SPERANZA, 2012); (STOYAN; KWON, 2010);
GAP	(GUASTAROBA; SPERANZA, 2012); (SANT'ANNA et al., 2017); (SCOZZARI et al., 2013); (WU; KWON; COSTA, 2017);
CWRTI	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (SCOZZARI et al., 2013);
MSE	(CHIAM; TAN; MAMUN, 2013); (NI; WANG, 2013); (GARCIA; GUIJARRO; OLIVER, 2018); (STRUB; TRAUTMANN, 2019);
IR	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS; NOMIKOS, 2014); (GIUZIO, 2017);
RMSE	(ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (SANT'ANNA et al., 2017)
MER	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014);
ANNR	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014);
ANNTE	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (GIUZIO, 2017); (SCOZZARI et al., 2013);

Table 9 – Evaluation metrics summary

(MIER), Accumulated Excess Return (AER), Augmented Dickey-Fuller t statistic (ADF), return, utility, Transaction Costs (TC). The classification of each article in an objective function is presented in table 11.

The relative frequency of works per objective function is shown in Figure 13. The answer to RQ #11 is positive since a good part of the works adopts RMSE and MSE.

Two main constraints are usually applied to index tracking problems: cardinality and holding. These constraints were adopted by articles that used mathematical programming only since their use is obligatory for the index tracking model in this framework. In this work, more practical constraints were taken into account to answer RQ #12. The constraints found were: Transaction Costs, Tracking Error, Turnover, Market Regulations, Class, CVaR and Round-lot. As well as the two main constraints, these practical constraints only occurred in the mathematical programming modeling framework. The relative number

Comparison Type	Articles
Heuristic Vs Heuristic	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS; NOMIKOS, 2014); (CHIAM; TAN; MAMUN, 2013); (GARCIA; GUIJARRO; OLIVER, 2018); (GRISHINA; LUCAS; DATE, 2017); (NI; WANG, 2013); (WU; KWON; COSTA, 2017);
Heuristic Vs CPLEX	(MUTUNGE; HAUGLAND, 2018); (SANT'ANNA et al., 2017); (SCOZZARI et al., 2013); (STOYAN; KWON, 2010);
Heuristic Vs Gurobi	(STRUB; TRAUTMANN, 2019); (WANG; XU; DAI, 2018);
Heuristic Vs ProjGrad	(GIUZIO, 2017); (XU; LU; XU, 2016);
Heuristic Vs IntPoint	(GIUZIO, 2017);
Heuristic Vs CycD	(GIUZIO, 2017);
Heuristic Vs Random	(AFFOLTER et al., 2016);

Table 10 – Comparison types summary

of works per constraint is shown in Figure 14.

Since the constraints are well distributed among the articles, there is no prevalence of a specific practical constraint when applying heuristics/metaheuristics and the answer relative to RQ #12 is negative. Table 12 shows the mathematical programming subcategory, constraint name and associated articles.

Even though this literature review focuses on the characteristics of models solved by heuristics/metaheuristics, the results presented in this part of the review may also be useful for researchers that are developing index tracking models. They can compare their objective functions and practical constraints with other authors' models that also included them in a specific quantitative modeling framework.

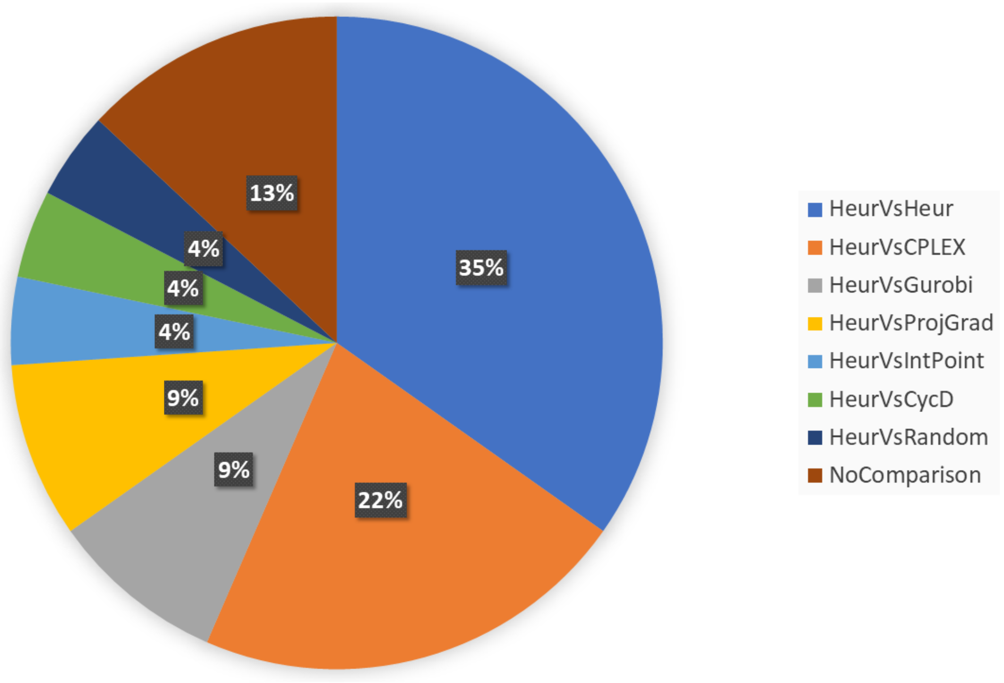

Figure 12 – Percentage of works per comparison type.

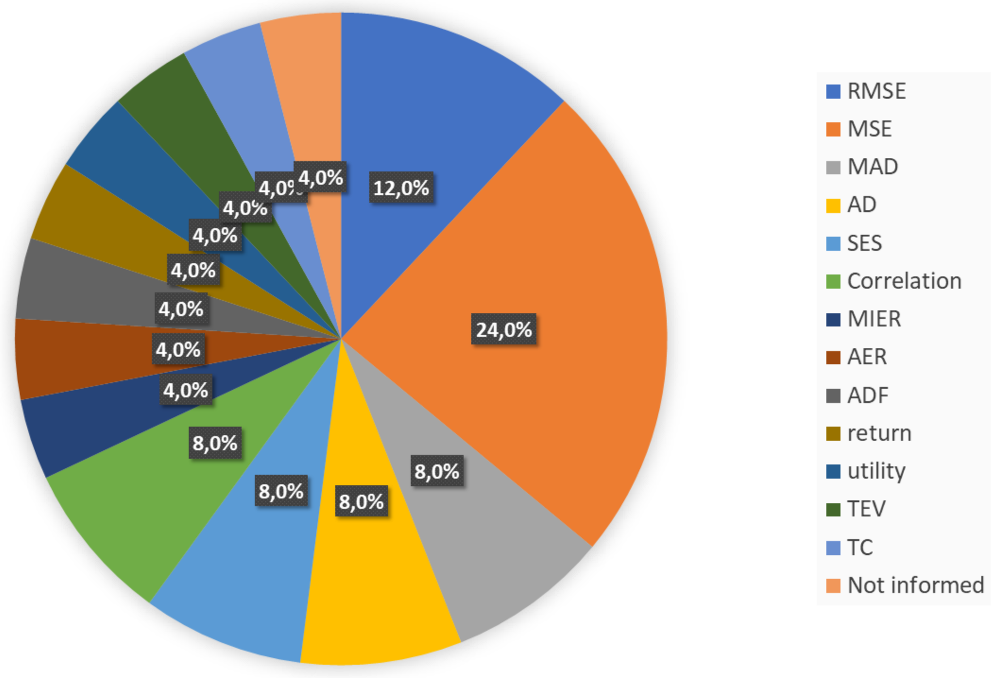

Figure 13 – Percentage of works per objective function.

3.2.2.7 Categorization - data sources for index tracking problems solved by heuristic/meta-heuristic

To answer research question RQ #13 it was necessary to identify all data sources. The following data sources were identified: OR-library, Datastream, Historical Stock Data Downloader (HSDD), Economatica, Bloomberg, Markit, Academic research lab (University)

Framework	Objective	Articles
Mixed-integer Non-linear programming	RMSE	(ANDRIOSOPOULOS; NOMIKOS, 2014); (ANDRIOSOPOULOS et al., 2013);
	MSE	(GARCIA; GUIJARRO; OLIVER, 2018); (SANT'ANNA et al., 2017); (SCOZZARI et al., 2013); (STRUB; TRAUTMANN, 2019);
	MAD	(GRISHINA; LUCAS; DATE, 2017);
	SES	(WANG; XU; DAI, 2018); (XU; LU; XU, 2016);
	TEV	(MUTUNGE; HAUGLAND, 2018);
Mixed-integer linear programming	MAD	(WANG et al., 2012)
	AD	(GUASTAROBA; SPERANZA, 2012);
	Correlation	(WU; KWON; COSTA, 2017);
Mixed-integer Conic programming	MIER	(WU; WU, 2019);
Stochastic Programming	AD	(STOYAN; KWON, 2010);
Multi-objective Optimization	RMSE	(NI; WANG, 2013);
	MSE	(CHIAM; TAN; MAMUN, 2013);
	AER	(NI; WANG, 2013);
	TC	(CHIAM; TAN; MAMUN, 2013);
Cointegration	ADF	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015);
	Correlation	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015);
Regression with regularization	SES	(XU; LU; XU, 2016);
	MSE	(GIUZIO, 2017);
Pure heuristics	Return	(AFFOLTER et al., 2016);

Table 11 – Objective function occurrence summary

and Yahoo Finance. Table 13 shows all the databases and associated articles.

Figure 15 shows the percentage of works per data source. OR-library and Datastream were adopted by most of the works that are transparent about their databases. Then, the answer to RQ #13 is OR-library and Datastream.

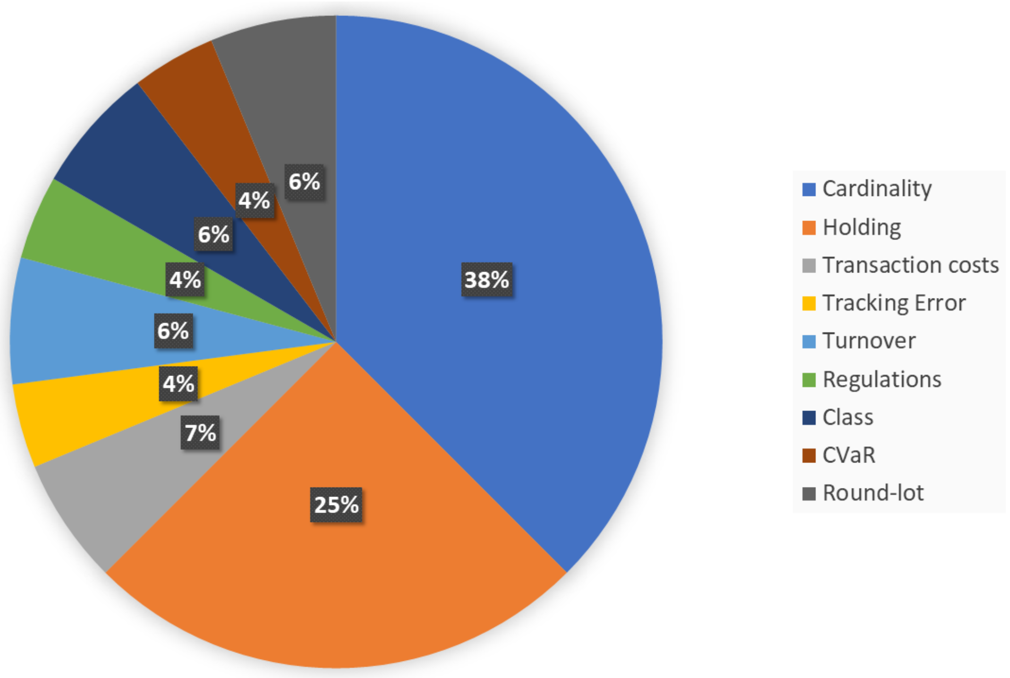

Figure 14 – Percentage of works per constraint.

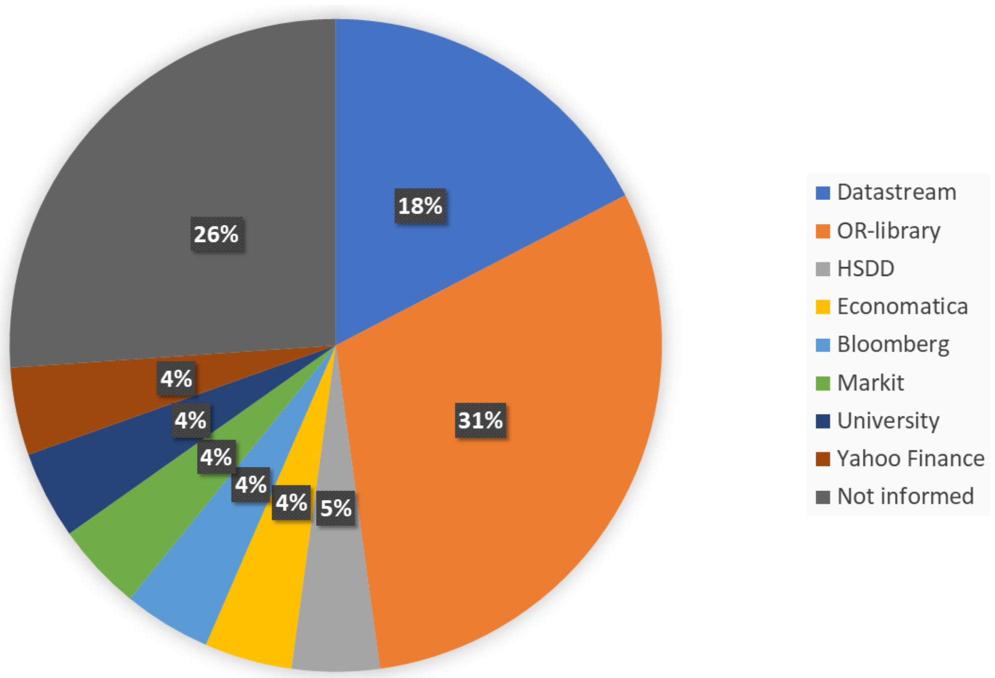

Figure 15 – Percentage of works per database.

Framework	Constraint	Articles
Mixed-integer Non-linear programming	Tracking Error	(SANT'ANNA et al., 2017);
	Turnover	(SCOZZARI et al., 2013);
	Market regulation	(SCOZZARI et al., 2013); (STRUB; TRAUTMANN, 2019);
	Class	(WANG; XU; DAI, 2018);
	CVaR	(WANG; XU; DAI, 2018);
	Round-lot	(WANG; XU; DAI, 2018);
Mixed-integer linear programming	Transaction costs	(GUASTAROBA; SPERANZA, 2012); (WU; KWON; COSTA, 2017)
	Class	(WU; KWON; COSTA, 2017);
	CVaR	(WANG et al., 2012);
Mixed-integer Conic programming	Tracking Error	(WU; WU, 2019);
Stochastic Programming	Turnover	(STOYAN; KWON, 2010);
	Class	(STOYAN; KWON, 2010);
Multi-objective Optimization	Transaction costs	(NI; WANG, 2013);
	Turnover	(NI; WANG, 2013);
	Round-lot	(CHIAM; TAN; MAMUN, 2013);

Table 12 – Practical constraint occurrence summary

Data source	Articles
Datastream	(AFFOLTER et al., 2016); (ANDRIOSOPOULOS et al., 2013); (ANDRIOSOPOULOS et al., 2013); (STRUB; TRAUTMANN, 2019);
OR-library	(GARCIA; GUIJARRO; OLIVER, 2018); (GRISHINA; LUCAS; DATE, 2017); (MUTUNGE; HAUGLAND, 2018); (STRUB; TRAUTMANN, 2019); (GUASTAROBA; SPERANZA, 2012); (WANG et al., 2012); (CHIAM; TAN; MAMUN, 2013);
HSDD	(MUTUNGE; HAUGLAND, 2018);
Economatica	(SANT'ANNA et al., 2017);
Bloomberg	(SANT'ANNA et al., 2017);
Markit	(WU; WU, 2019);
University	(WU; KWON; COSTA, 2017);
Yahoo Finance	(ACOSTA-GONZALEZ; ARMAS-HERRERA; FERNANDEZ-RODRIGUEZ, 2015);

Table 13 – Data sources summary

3.3 GRASP APPROACH FOR THE FINANCIAL PORTFOLIO PROBLEM.

A systematic literature review concerning GRASP applied to the financial portfolio optimization problem was developed in this work. The methodology and results will be presented and discussed in the next subsections.

3.3.1 Methodology

The procedures and rules that conducted the systematic literature review were based on Kalayci, Ertenlice & Akbay (2019).

3.3.1.1 Research questions set up

In order to guide the result analysis and scope of this work, some research questions were defined. The research questions are presented in Table 14

RQ	Description
#1	Are there any GRASP applications in the index tracking problem?
#2	In which financial portfolio optimization problems GRASP was applied?
#3	What were the most used types of objective functions?
#4	Non-convex or convex models?
#5	Was CPU time performance taken into consideration?
#6	Did any solvers, metaheuristics or heuristics were used as benchmark for performance comparisons?
#7	Which databases were adopted?

Table 14 – Research questions for GRASP systematic literature review

3.3.1.2 Answering the first research question

The most important research question related to this work is the first one since it represents the core contribution. The keyword structure used to answer RQ#1 is presented in Table 15

The search was conducted by covering papers published from 2000 to 2019 and written in English language. No papers were found on the selected databases. This result shows that using GRASP for index tracking models brings innovation to the literature since this metaheuristic was never experimented in this kind of problem. Now, another search will be performed to investigate the state-of-the-art of GRASP applications in the financial portfolio optimization context and aiming to answer the other research questions.

Level	Search Terms
1	GRASP OR "Greedy Randomized Adaptive Search Procedure"
	AND
2	Heuristic* OR Metaheuristic* OR Meta-heuristic*
	AND
3	Portfolio OR Investment OR Asset
	AND
4	Optimization OR Management OR Selection
	AND
5	"Index Tracking" OR "Tracking Error" OR Tracking-Error

Table 15 – Proposed keyword combination structure for the GRASP applied to index tracking systematic review

3.3.1.3 Material collection and selection to answer the remaining research questions

The search was conducted by covering papers published from 2000 to 2019 and written in English language. The bibliography collection process took place at Scopus and Web of Science databases using the keyword structure presented in Table 16. This keyword combination structure was built from the author's a priori knowledge of the field. Levels 1 and 2 refer to the main solution approach. Levels 3 and 4 guides the search for financial portfolio optimization problems. Level 5 restricts the financial portfolio problem class to index tracking.

Level	Search Terms
1	GRASP OR "Greedy Randomized Adaptive Search Procedure"
	AND
2	Heuristic* OR Metaheuristic* OR Meta-heuristic*
	AND
3	Portfolio OR Investment OR Asset
	AND
4	Optimization OR Management OR Selection

Table 16 – Proposed keyword combination structure to accquire GRASP material

A total of 9 and 13 papers were found at Web of Science and Scopus databases, respectively.

3.3.1.4 Filtering process

The filtering was performed by fully reading the papers. This filter removes articles that do not match the scope of this work. Many of the filtered works appeared in the results because of the "optimization" and "investment" keywords. A total of two (Web of Science) and three (Scopus) articles were selected by this filter. An extra article was found at the Scopus database, the other two appeared on both databases. Then, 3 GRASP applications to the financial portfolio optimization problem were found.

3.3.2 Results and discussion of the GRASP systematic review

The first work regarding the adaptation of GRASP in portfolio optimization was conducted by Anagnostopoulos, Chatzoglou & Katsavounis (2010) when the authors proposed a reactive GRASP to solve the classical formulation of the portfolio selection model with the addition of a cardinality constraint. The proposed algorithm was not applied to any instance of the problem.

A more recent application of GRASP for portfolio optimization was developed by Baykasoglu, Yunusoglu & Ozsoydan (2015). The authors proposed a metaheuristic named GQ to solve cardinality constrained problems. These authors used GRASP to select stocks for compiling a portfolio, and a quadratic programming model for determining the proportions of each stock. The results showed GQ successfully solved the problem and achieved competitive performances when compared with alternatives in the literature (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015).

Based on GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015), Otken et al. (2019) designed a fast GRASP & SOLVER and GRASP & SOLVER with turnover cost and applied them on an extended MVO problem, on an active portfolio management approach. The model decision variables and objective function structure is equivalent to those used by ROLL (1992), representing an active manager aiming to beat a benchmark, but extended with practical constraints. The practical constraints were tracking error, active share and cardinality and the heuristic solutions were applied in a real-life S&P500 data set ranging from 2007 to 2016. The turnover cost algorithm performed better in terms of annual return. The summary of the answers to the research questions is presented in Table 17.

In summary, GRASP has not been much explored for solving portfolio selection problems. More specifically, a research gap was identified regarding the application of GRASP for passive fund management formulations, which is covered in the present paper.

RQ	Anagnostopoulos, Chatzoglou & Katsavounis (2010)	Baykasoglu, Yunusoglu & Özsoydan (2015)	Ötken et al. (2019)
#2	Markowitz (1952)	Markowitz (1952)	ROLL (1992)
#3	Cardinality & holding	Cardinality & holding	Cardinality, holding turnover, active share, risk factor, class & tracking error
#4	No	Yes	No
#5	No	No	Yes
#6	No	Yes	No
#7	None	Beasley OR library	Principal Global Investors (S&P500)

Table 17 – Summary of the answers to the remaining RQ's of the GRASP applied to the financial portfolio problem literature review

3.4 CHAPTER CONCLUSION

This chapter presented current developments of the financial portfolio optimization problem and two systematic literature reviews. The objective of the first systematic review was to develop an investigation concerning the current solution approaches for the index tracking problem using a set of research questions. The set of research question was divided in two parts. The first part refers to general solution approaches for the index tracking problem and comparison among two groups: heuristic and non-heuristic methods. The second part refers to a specific analysis of heuristic/metaheuristic approaches applications developed for this problem.

The second systematic literature review refers to GRASP applications on the financial portfolio optimization field. There are three applications of the GRASP metaheuristic on the mean-variance CCPO framework. Applications of GRASP to the index tracking problem were not found. Then, taking the results presented in this chapter into consideration, this dissertation proposes an innovation for the portfolio optimization field by developing an adaptation and application of the GRASP metaheuristic for the index tracking problem.

RQ	Description
#1	Index tracking solution methods are more relevant to journals focusing on operations research and computer science
#2	The vast amount of the developed heuristics/metaheuristics solutions were applied to mathematical programming formulations more often
#3	There has been a growth in the number of non-heuristic methods applied to the index tracking problem
#4	There has been a growth in the number of heuristics/metaheuristics applied to the index tracking problem
#5	Heuristic approaches are not more used than non-heuristic approaches for index tracking problems
#6	Heuristic approaches have more cite impact than non-heuristic approaches for index tracking problems
#7	There is a prevalence of using Differential Evolution and Genetic algorithms in index tracking problems
#8	Solvers are integrated with heuristics. A total of 11 hybridized heuristics were found
#9	There is a prevalence of using specific evaluation metrics for heuristic approaches. The most used metrics were RMSE and MSE
#10	Yes heuristics are more compared against other heuristics or against the CPLEX solver
#11	There is a prevalence of solving for a specific tracking error objective function when using heuristic approaches. A good part of the works adopted RMSE and MSE
#12	Is there a prevalence of solving for specific practical constraints when using heuristic approaches?
#13	The most used databases were OR-library and datastream

Table 18 – Summary of the answers to each research question of the index tracking systematic literature review

4 PROPOSED GRASP APPROACH FOR THE INDEX TRACKING PROBLEM.

The systematic literature about GRASP applied to the financial portfolio problem developed in this work showed that GRASP has been applied in the portfolio optimization problem. There are three applications of GRASP to this problem, where two among them use the hybrid GQ metaheuristic (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015). The formulation considered by Baykasoglu, Yunusoglu & Ozsoydan (2015) is a cardinality constrained MVO problem. This metaheuristic was applied in the mathematical programming modeling framework, more specifically in a MINLP formulation.

As it can be observed from the index tracking literature review, hybrid metaheuristics/heuristics are mostly compared with general-purpose solvers such as CPLEX and Gurobi (GUASTAROBA; SPERANZA, 2012; SANT'ANNA et al., 2017; SCOZZARI et al., 2013; STOYAN; KWON, 2010; STRUB; TRAUTMANN, 2019; WANG; XU; DAI, 2018). That is because hybrid heuristics simulate a branch-and-bound procedure that select active search nodes in a more objective way, also, these methods can be instructed to never compute incomplete solutions. The trade-off is that hybrid heuristics cannot guarantee optimality and behave in a stochastic way, but can find a good solution in less time. Then, it is interesting to evaluate the gain of betting on the searchability of a hybrid heuristic instead of trying to find the optimal solution using a general-purpose solver. For instance, compare the quality of the solution produced by the hybrid heuristic and the general-purpose solver, taking into consideration the required time to generate each solution.

We select the GQ metaheuristic to evaluate an adaptation of GRASP for the index tracking problem. To the best of our knowledge, this is the first time GRASP is used to solve the index tracking problem. This section presents an index tracking model to test the adapted modification of the GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015) metaheuristic for index tracking. Also, an alternative algorithm for a component of the local search procedure is presented. The proposed approaches were developed so that their performance could be evaluated against that of a solver when trying to find a good solution for an index tracking problem.

4.1 THE INDEX TRACKING MODEL.

The first criteria considered for the index tracking model selection were adaptation necessity and applicability. In other words, the selected model must have been solved by a hybrid heuristic/metaheuristic and its quantitative modeling framework must be the same as the one that was solved by GQ. GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015) was designed for problems within the mathematical programming quantitative modeling framework. Index tracking models solved by hybrid heuristics are mostly modeled as

MINLP or MILP. That is the perfect match, since GQ can be adapted to the majority of available index tracking models. An initial set of models was assembled by considering MINLP and MILP index tracking models approached by hybrid heuristics/metaheuristics that were compared with general-purpose solvers. The works and their respective objective functions are presented in Table 19.

Article	Objective Function
(GUASTAROBA; SPERANZA, 2012)	AD
(SANT'ANNA et al., 2017)	MSE
(SCOZZARI et al., 2013)	MSE
(STRUB; TRAUTMANN, 2019)	MSE
(WANG et al., 2012)	MAD
(WANG; XU; DAI, 2018)	SES

Table 19 – First model set

Three models contain the same type of objective function MSE. It was decided to pick one model to represent this MSE class of models. (STRUB; TRAUTMANN, 2019) adopted a model similar to that of (SCOZZARI et al., 2013). Both of them used constraints to reflect the European Union Derivative UCITS (Undertaking for Collective Investment in Transferable Securities) rules. Since those models were designed for European Union markets, then they were discarded to prevent loss of generality in the application developed in this work. Therefore, (SANT'ANNA et al., 2017) model will represent the MSE class of models. Table 20 shows the final model set and their notation.

Article	Notation
(GUASTAROBA; SPERANZA, 2012)	ADG
(SANT'ANNA et al., 2017)	MSES
(WANG et al., 2012)	MADW
(WANG; XU; DAI, 2018)	SESW

Table 20 – Final model set

Next, the models contained in this final alternative set are depicted in alphabetical order. All the models contain constraints (2.23) and (2.24) which were presented in Section 2.2. (GUASTAROBA; SPERANZA, 2012) ADG index tracking model is shown below.

$$\text{minimize} \quad TE = \sum_{t=1}^{T} \left(\theta I_t - \sum_{j=1}^{n} q_{jt} X_j^1 + 2u_t \right) \tag{4.1}$$

$$\text{subject to} \quad \sum_{j=1}^{n} q_{jT} X_j^1 \le C \tag{4.2}$$

$$\sum_{j=1}^{n} Z_j \le K \tag{4.3}$$

$$L_i Z_i \le X_j^1 q_{jT}/C \le U_i Z_i \ , \qquad i = 1, 2, ..., N \tag{4.4}$$

$$\left(X_j^1 - X_j^0 \right) q_{jT} + 2s_j \le U_j C w_j \ , \qquad j = 1, 2, ..., n \tag{4.5}$$

$$\sum_{j=1}^{n} \left[\left(X_j^1 - X_j^0 \right) c_j^b q_{jT} + (c_j^b + c_j^s) s_j + f_j w_j \right] \le \gamma C \tag{4.6}$$

$$\theta I_t - \sum_{j=1}^{n} q_{jt} X_j^1 + u_t \ge 0 \qquad t = 1, 2, ..., T \tag{4.7}$$

$$\left(X_j^1 - X_j^0 \right) q_{jT} + s_j \ge 0 \ , \qquad j = 1, 2, ..., n \tag{4.8}$$

$$X_j^1 \ge 0, \ s_j \ge 0, \ Z_j \in \{0, 1\}, \ w_j \in \{0, 1\}, \ j = 1, 2, ..., n \tag{4.9}$$

$$u_t \ge 0, \ t = 1, 2, ..., T \tag{4.10}$$

Where:

TE = tracking error measure represented by positive and negative deviations from the index return

$t = 1, 2, ..., T$ is the time index

$j = 1, 2, ..., n$ is the asset index

q_{jt} = Value of asset j at time t

I_j = Index value at time t

b_j = Total purchase value of asset j

s_j = Total selling value of asset j

c_b^j = Proportional transaction cost paid for buying asset j

c_s^j = Proportional transaction cost paid for selling asset j

f_j = Fixed transaction cost paid for selling or buying asset j

w_j = Is equal to 1 if the investor buys or sells any quantity of asset j. It assumes 0 otherwise.

X_j^0 = proportion of stock j in the portfolio before rebalancing

X_j^1 = proportion of stock j in the portfolio after rebalancing

Z_j = Indicates if stock j is included in the portfolio or not

K = Maximum number of assets in the portfolio

L_j = Lower bound on stock i

U_j = Upper bound on stock i

C = Capital available for investment in scenario T

Both Equation (4.1) and constraint (4.7) model an AD objective function. The objective is to minimize the AD tracking error between the portfolio value and index value. Inequality (4.2) is the budget constraint, (4.3) is the cardinality constraint and (4.4) is the holding constraint. Inequality (4.6) imposes a limit on the total transaction costs. Inequality (4.5) represents a transaction indicator, forcing w_j to be 1 when a transaction occurs, thus adding the associated fixed cost of asset j. (4.8) is used to connect selling quantities and buying quantities of each asset j.

(SANT'ANNA et al., 2017) MSES index tracking model is shown below.

$$\text{minimize} \quad TE = \frac{1}{T}\sum_{t=1}^{T}\Big[\sum_{i=1}^{N} x_i r_{it} - R_t\Big]^2 \tag{4.11}$$

$$\text{subject to} \quad \sum_{i=1}^{N} x_i r_{it} - R_t \geq \gamma \ , \qquad t = 1, 2, ..., T \tag{4.12}$$

$$\sum_{i=1}^{N} x_i r_{it} - R_t \leq \theta \ , \qquad t = 1, 2, ..., T \tag{4.13}$$

$$\sum_{i=1}^{N} Z_i \leq K \tag{4.14}$$

$$0 \leq x_i \leq Z_i \ , \qquad i = 1, 2, ..., N \tag{4.15}$$

$$\sum_{i=1}^{N} x_i = 1 \tag{4.16}$$

$$Z_i \in \{0, 1\}, \ i = 1, 2, ..., N \tag{4.17}$$

Where:

TE = tracking error measure represented by positive and negative deviations from the index return

$t = 1, 2, ..., T$ is the time index

$i = 1, 2, ..., N$ is the asset index

$\gamma = minimumtrackingerror$

$\theta = maximumtrackingerror$

$i = 1, 2, ..., N$ is the asset index

R_t = Index return at time t

r_{it} = Stock i return at time t

x_i = Stock i proportion in the portfolio

Z_i = Indicates if stock i is included in the portfolio or not

K = Maximum number of assets in the portfolio

The objective function (4.11) aims to minimize the mean squared error of portfolio returns relative to the index returns. Constraints (4.12) and (4.13) allow the tracking error to vary between γ and θ at each time t (4.14) is the cardinality constraint. (4.22) is the budget constraint.

(WANG et al., 2012) MADW index tracking model is shown below.

$$\text{minimize} \qquad TE = \frac{1}{T}\sum_{t=1}^{T}(q_t^+ + q_t^-) \tag{4.18}$$

$$\text{subject to} \qquad (q_t^+ + q_t^-) = R_t - \sum_{i=1}^{N} r_{it}x_i \ , \qquad t = 1, 2, ..., T \tag{4.19}$$

$$\sum_{i=1}^{N} Z_i = K \tag{4.20}$$

$$L_i Z_i \leq x_i \leq U_i Z_i \ , \qquad i = 1, 2, ..., N \tag{4.21}$$

$$\sum_{i=1}^{N} x_i = 1 \tag{4.22}$$

$$Z_i \in \{0, 1\},\ i = 1, 2, ..., N \tag{4.23}$$

$$q_t^+ \geq 0,\ q_t^- \geq 0,\ t = 1, 2, ..., T \tag{4.24}$$

Where:

TE = tracking error measure represented by positive and negative deviations from the index return

$t = 1, 2, ..., T$ is the time index

$i = 1, 2, ..., N$ is the asset index

q_t^+, q_t^- = Variables used to linearize the Mean Absolute Deviation function

R_t = Index return at time t

r_{it} = Stock i return at time t

x_i = Stock i proportion in the portfolio

Z_i = Indicates if stock i is included in the portfolio or not

K = Number of assets in the portfolio

L_i = Lower bound on stock i

U_i = Upper bound on stock i

The objective function (4.18) and constraint (4.19) represent a piecewise and not differentiable MAD objective function. The constraint (4.20) is the cardinality constraint, (4.21) limits the proportion of stocks with lower and upper bounds and (4.22) is the budget constraint. This model was originally applied for a single period optimization.

(WANG; XU; DAI, 2018) SESW index tracking model is shown below.

$$\begin{aligned}
\text{minimize} \quad & (1-\lambda)V(Z) + \lambda TE && (4.25)\\
\text{subject to} \quad & \sum_{i=1}^{N} Z_i = K && (4.26)\\
& l_j \leq \sum_{i=S_{j-1}+1}^{S_j} Z_i \leq u_j \ , \qquad j = 1, 2, ..., M && (4.27)\\
& L_i Z_i \leq x_i \leq U_i Z_i \ , \qquad i = 1, 2, ..., N && (4.28)\\
& \sum_{i=1}^{N} x_i = 1 && (4.29)\\
& Z_i \in \{0, 1\}, \ i = 1, 2, ..., N && (4.30)\\
& && (4.31)
\end{aligned}$$

Where:

TE = tracking error measure. $TE(w) = \frac{(I-Rw)^T(I-Rw)}{T}$
V = Estimation accuracy
λ = Tuning parameter
$j = 1, 2, ..., M$ is the class index
$i = 1, 2, ..., N$ is the asset index
x_i = Stock i proportion in the portfolio
Z_i = Indicates if stock i is included in the portfolio or not
K = Number of assets in the portfolio
L_i = Lower bound on stock i
U_i = Upper bound on stock i
l_i = Lower bound on the sample size of te jth class
u_i = Upper bound on the sample size of te jth class

The objective (4.25) aims to minimize a merit function. This merit function combines the estimation accuracy and tracking error using a parameter λ defined by the investor. Constraint (4.27) gives the scope of sample size of the jth class. (4.28) is the floor-ceiling constraint, (4.26) is the cardinality constraint and (4.29) is the budget constraint.

The selected metaheuristic GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015) was originally designed to handle cardinality and floor-ceiling constraints only. Even though these two obligatory constraints occur in almost all MINLP and MILP models, RQ #12, answered in Section 3.2, confirms that, apart from these obligatory constraints, there was no prevalence in the inclusion of other specific restrictions whatsoever. Considering these two facts, it was decided to adopt (WANG et al., 2012) model to test the adaptation of GQ, since it contains only these two obligatory constraints and hence avoids loss of generality of the application. Other observations concerning advantages of the (WANG et al., 2012) model is that its relaxation (model without the cardinality constraint) is convex and linear, thereby it can be solved very easily by algorithms contained in general-purpose solvers, such as simplex and interior point.

4.2 ADAPTED GRASP-QUAD METAHEURISTIC

GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015) was originally designed for the mean-variance CCPO, which is an NP-Hard problem, also the case of index tracking optimization. One of the main advantages of using GQ is that it reduces the computational cost of solving the combinations of stocks and allocating weights to stocks. The name GQ comes from linking GRASP and quadratic programming, but adjustments can make it work with linear programming too. It constructs feasible solutions at two levels: stock selection and proportion/weight optimization. After constructing the solution, it performs a local search using two components, namely a MiLS and a MaLS. When MiLS occurs, a search is performed in the following manner: one stock from S is exchanged by another stock from S_c. When the multistart procedure ends, MaLS occurs. In MaLS, every stock contained in S is exchanged by all those stocks contained in S_c, one by one. GQ algorithm is shown in Algorithm 2. Further details are given in Baykasoglu, Yunusoglu & Ozsoydan (2015):

The first level of GQ occurs in the following manner: first, stocks are sorted according to their associated greedy values, obtained by using a greedy function; and after that, K stocks are chosen from the RCL, and this meets the cardinality constraint. The RCL is the main component of the first level and is the primal search mechanism of GRASP. The size of this component influences the greediness and randomness of the search.

The extreme scenarios of RCL_{size} are: to build a solution within a local optimum or blindfolded. Using a relatively small RCL_{size}, the search will be conducted in a greedy way, choosing the most desired stocks, which in the case of minimization are the ones with the smallest greedy values. Adopting a relatively bigger RCL_{size} results in a random search, thereby diluting the effects of the greedy values. Figure 16 shows distributions of solutions generated by semi-greedy and random searches.

This result shown in Figure 16 was presented in Resende & Ribeiro (2016) for an instance of the *max covering problem* and will be used to illustrate the effect on the distribution of solutions for different sizes of the RCL. Solutions generated by multistart procedures using a relatively smaller RCL_{size} (semi-greedy search) are on average much better than solutions generated by multistart procedures using a relatively larger RCL_{size} (random search). Also, the semi-greedy search produces solutions with less variability. Nevertheless, it is interesting to choose a RCL_{size} that balances the trade-off between greediness and randomness.

The greedy function g_i proposed in this work was built based on the original non-linear objective function of the index tracking problem of Wang et al. (2012). This greedy function is presented in equation (4.32) and it computes the sum of absolute deviations of the

Algorithm 2: GRASP-QUAD - Minimization case

Input: $iter_{MAX}$, RCL_{size}, p_{MiLS}, K, UB_{MiLS}
Output: $fval$, S_{final}
$iter \leftarrow 1$, $fval \leftarrow \infty$, $S_{final} \leftarrow \{\}$
while $iter \leq iter_{MAX}$ **do**
  $S \leftarrow \{\}$, $S_c \leftarrow \{stock_1, stock_2, ..., stock_N\}$
  while $|S| < K$ **do**
    $g \leftarrow getGreedyVals(S, S_c)$
    $RCL \leftarrow getRCL(g, S_c, RCL_{size})$
    $s \leftarrow randSample(|RCL|, 1)$
    $S \leftarrow S \cup RCL(s)$
    $S_c \leftarrow S_c - RCL(s)$
  end while
  $fvaltemp \leftarrow optimizeProportions(S)$
  if $rand(0, 1) \leq p_{MiLS}$ **then**
    $[fval_{MiLS}, S_{MiLS}] \leftarrow MiLS(S, UB_{MiLS})$
    if $fval_{MiLS} < fvaltemp$ **then**
      $fvaltemp \leftarrow fval_{MiLS}$
      $S \leftarrow S_{MiLS}$
    end if
  end if
  if $fvaltemp < fval$ **then**
    $fval \leftarrow fvaltemp$
    $S_{final} \leftarrow S$
  end if
end while
$[fval, S_{final}] \leftarrow MaLS(S_{final})$

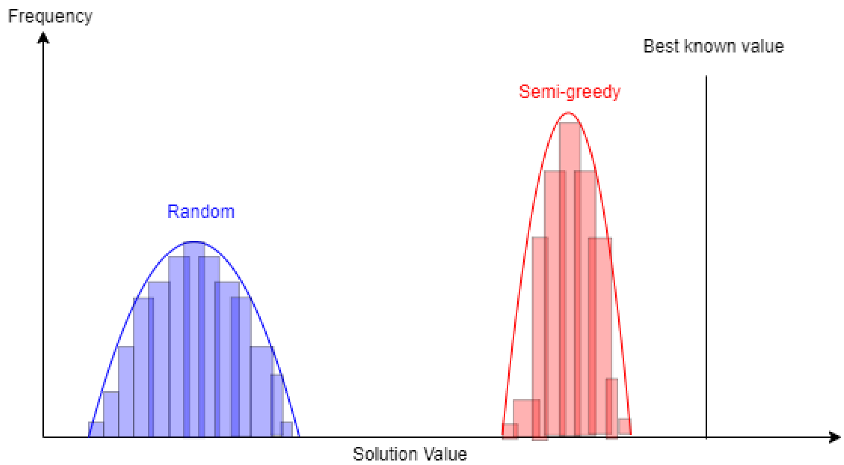

Figure 16 – Comparison of solution distribution generated by semi-greedy and random search. Adapted from Resende & Ribeiro (2016)

return of a naïve portfolio, composed by a candidate stock i and stocks j which were

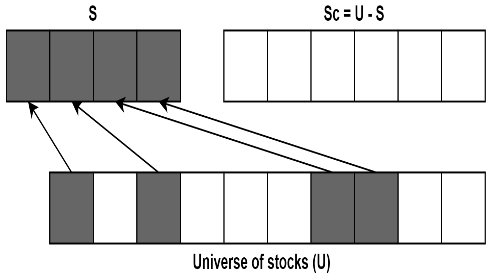

Figure 17 – The trial solution S and unselected stocks S_c, when $K = 4$ and $|U| = 10$

already included in the trial solution, in relation to the index return at time t.

$$g_i = \sum_{t=1}^{T} |R^t - \frac{1}{|S|+1}\left(r_i^t + \sum_{j \in S} r_j^t\right)| \quad (4.32)$$

where T is the in-sample period size. S and S_c are illustrated in Figure 17, considering that the universe U is a set of assets from a stock market. It would have a a considerably high computational cost to consider all assets available in an specific stock market, thus, it is necessary to consider a reduced set with a selected number of stocks. There is a rule of thumb that recommends to include those stocks that compose the index to be tracked.

An equally-weighted allocation was considered because there is no clue on how to allocate weights according to the problem formulation before the capital allocation phase. After choosing K stocks, the second level of GQ is initiated. At this level, the proportion of S is adjusted by a linear/quadratic solver, based on the mathematical programming model of choice, and $fvaltemp$ will be computed. The commercial solver CPLEX was set for this level.

After the first weight optimization of S, MiLS can be performed with probability p_{MiLS}, performing a local search in the neighborhood of the newly found trial solution. MiLS is simple because the search for better solutions is made by exchanging a stock from the trial solution S with a stock from the complement of the trial solution S_c. Finally, the objective function value of the trial solution, $fvaltemp$, is compared with the objective function value of the best solution found until then, $fval$. If $fvaltemp$ is better than $fval$, the current best trial solution S_{final} is updated.

When the number of iterations reaches its maximum value, MaLS is applied to the best trial solution found. This second component of the local search tests new combinations with all ignored stocks (S_c) until it is no longer possible to improve the trial solution anymore. The GQ process flowchart is shown in Figure 18.

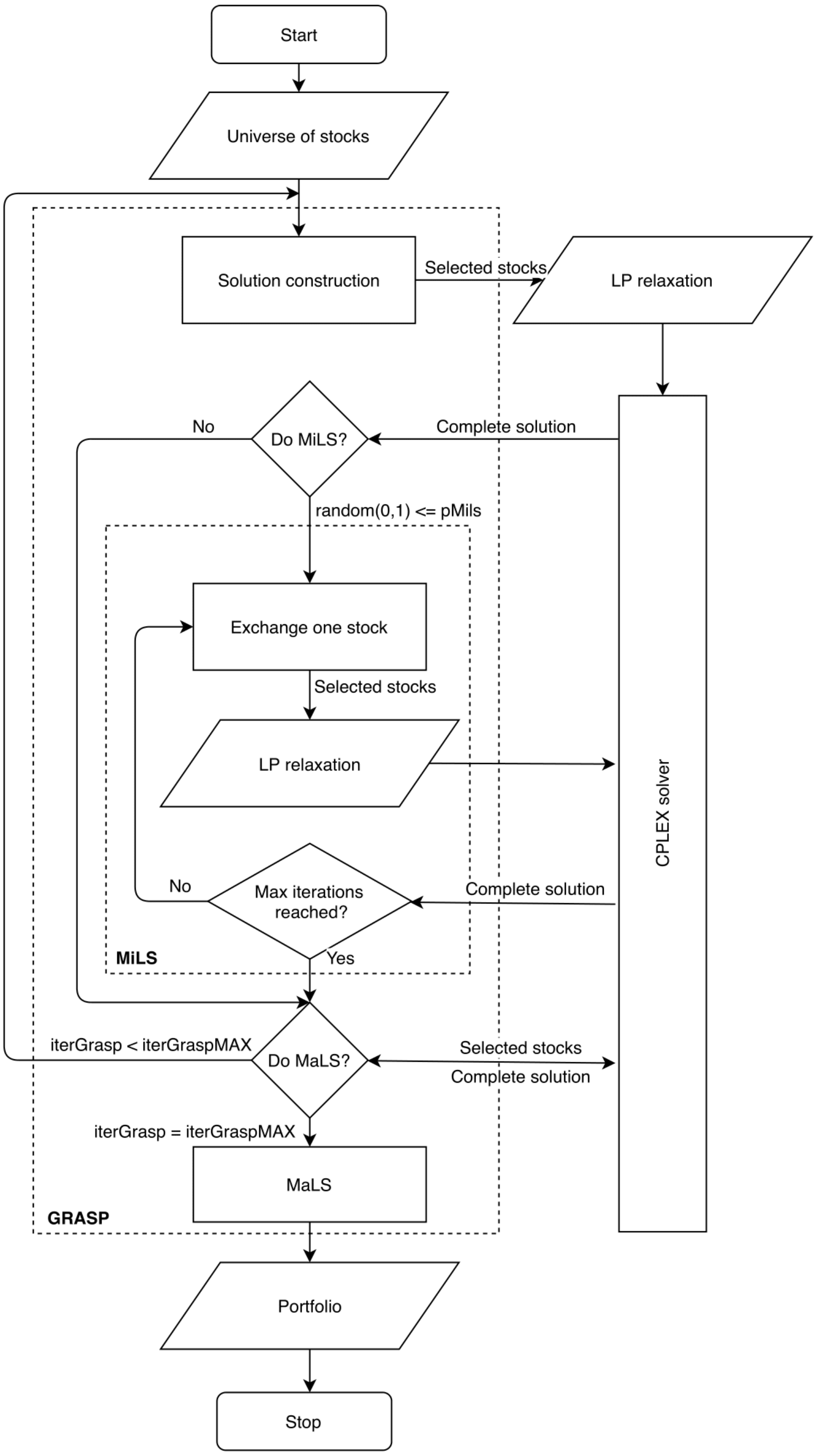

Figure 18 – GQ flowchart.

4.3 PROPOSED ϕ-DEPTH MINOR LOCAL SEARCH (ϕ-DMILS)

The ϕ-Depth Minor Local Search (ϕ-DMiLS) proposed in this work is a modification of GQ original MiLS (GQ-MiLS). The development of ϕ-DMiLS was inspired by the fact that it is possible to input a poor trial solution in MaLS that will make it produce a poor final solution, which means that this solution will not be within the neighborhood of the optimal solution.

It was decided to perform a deeper search in the neighborhood of a solution constructed by the RCL, exchanging one or more assets simultaneously, at the cost of a higher CPU time relative to GQ-MiLS, in order to input better solutions in MaLS. The algorithm of ϕ-DMiLS is shown in Algorithm 3.

The main principle that governs ϕ-DMiLS is to keep a deep search, exchanging more than one stock, while the solution is poor, and reduce the depth as the solution improves. The modified GQ flowchart is shown in Figure 19

Algorithm 3: ϕ-Depth Minor Local Search - Minimization Case

Input: S, S_c, $fval$, ϕ, UB_{MiLS}

Output: $fval_{\phi-Depth}$, $S_{\phi-Depth}$

while $\phi > 0$ **do**

$S_{MiLS} \leftarrow S$, $fval_{MiLS} \leftarrow fval$, $bool\_improved \leftarrow 0$

while $i < UB_{MiLS}$ **do**

$S_t \leftarrow S$

$s_1 \leftarrow randSample(|S|, \phi)$ # *choose* ϕ *assets without replacement*

$s_2 \leftarrow randSample(|S_c|, \phi)$

for $index_s = 1 : \phi$ **do**

$S_t(s_1(index_s)) \leftarrow S_c(s_2(index_s))$ # *exchange* s_1 *by* s_2

end for

$fval_t \leftarrow optimizeProportions(S_t)$

if $fval_t < fval_{MiLS}$ **then**

$bool\_improved \leftarrow 1$

$fval_{MiLS} \leftarrow fval_t$

$S_{MiLS} \leftarrow S_t$

end if

$i \leftarrow i + 1$

end while

if $bool\_improved$ **then**

maintain depth

$S \leftarrow S_{MiLS}$, $S_{\phi-Depth} \leftarrow S_{MiLS}$

$S_c \leftarrow \{stock_1, ..., stock_N\} - S_{MiLS}$

$fval \leftarrow fval_{MiLS}$, $fval_{\phi-Depth} \leftarrow fval_{MiLS}$

else

get shallower

$\phi \leftarrow \phi - 1$

end if

end while

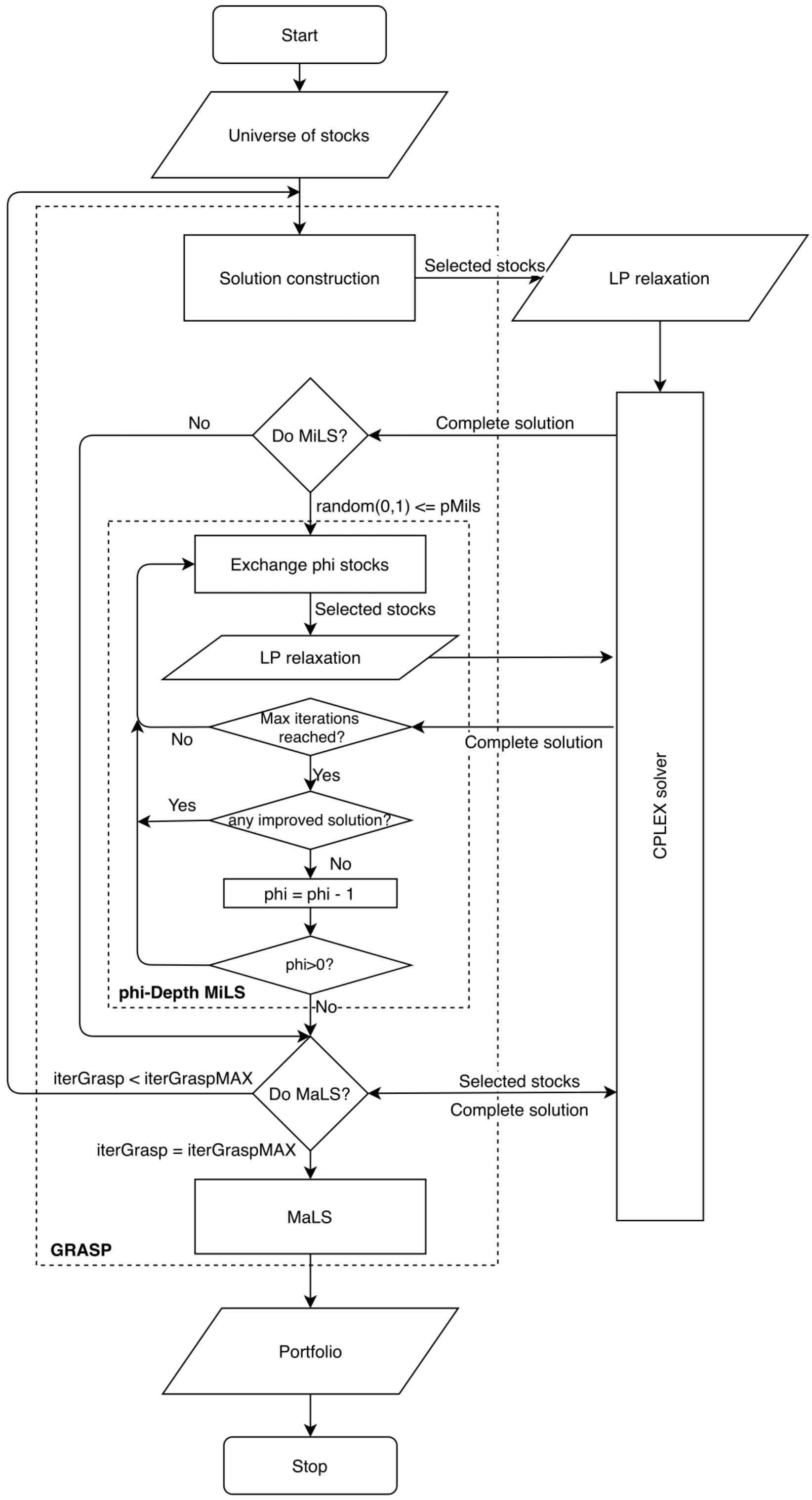

Figure 19 – Modified GQ flowchart.

4.4 CHAPTER CONCLUSION

This chapter presented the proposed GRASP approach and the model it will solve. The adopted GRASP metaheuristic is the one developed by (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015), named GQ. A new greedy function was proposed to adapt to the index tracking problem GQ. Also, a modified local search component was modified. In this modification more than one asset is exchanged and the number of assets exchanged per iteration is maintained if a solution improvement occurs. In the next chapter the proposed GRASP approach will be evaluated against a commercial SOLVER.

5 EXPERIMENTS AND RESULTS

This section exposes the results of experiments over parameter configuration and evaluation of the best parameter setup of the GRASP heuristics applied to the selected index tracking model. Initial experiments concern parameter configuration, since the choice of the parameters of a heuristic will influence in the quality of the final solution (MARTI; PARDALOS; RESENDE, 2018). The final experiments determine the best algorithm for the considered index tracking model. All the performance evaluations are relative to the solutions obtained by a general purpose solver.

5.1 EXPERIMENT DESIGN FOR THE PROPOSED ADAPTATION AND THE INCREMENT OF THE GQ LOCAL SEARCH

The goal of theoretical studies about stochastic algorithms are (MARTI; PARDALOS; RESENDE, 2018):

- Select and apply stochastic algorithms to choose the best for the given problem
- Define the best algorithm parameters
- Modify the design of the algorithms to achieve better ones

The original GQ was adapted and modified. Then, taking the objectives mentioned above into consideration, experiments were run guided by the experiment design presented in Figure 20

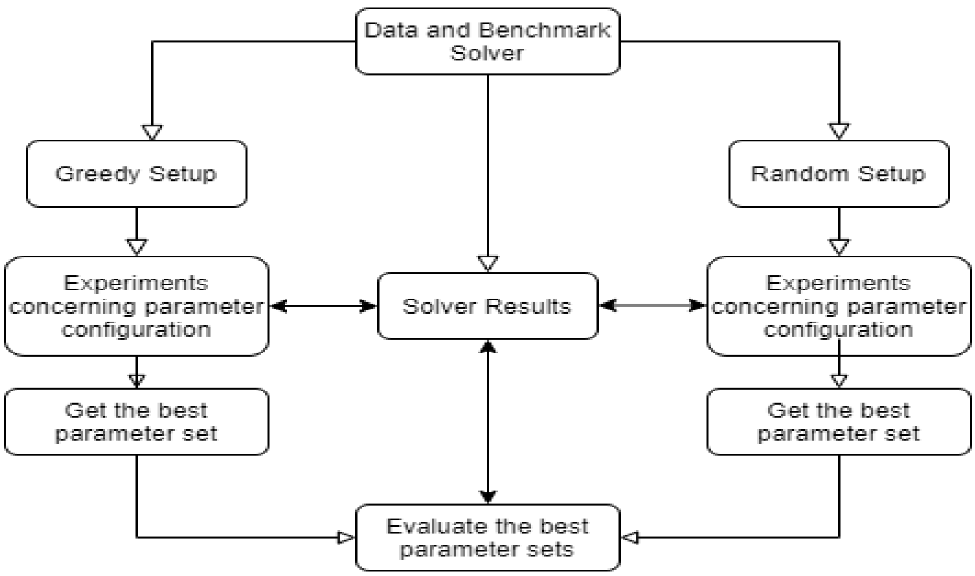

Figure 20 – Experiment design

The first step is the selection of the database and the benchmark solver. Next, solutions are computed for the associated instances of the Wang et al. (2012) model by the selected benchmark solver. Having these solutions in hand, groups of parameters can be estabilished. Those groups are divided in two so that the effectiveness of the greedy function can be measured. The effect of the greedy function will be reflected by a greedy group, which configures a semi-greedy search, balancing greediness and randomness. The group that dilute the effect of the greedy function is the random group. The sets in each group will be compared using the solutions obtained by the solver. The same happens with the random group. After finding the best parameter set for each group, a final evaluation using MaLS is performed to define the best heuristic and to check the effectiveness of the proposed greedy function.

5.2 BENCHMARK SOLVER AND PERFORMANCE EVALUATION

It was decided to choose CPLEX as the benchmark solver since it can solve the chosen model and it is the most used solver for benchmarking against heuristic in index tracking problems, as discussed in Secion 3.2. The solution generated by CPLEX is obtained through the branch-and-cut algorithm (IBM, 2020), which is a combination of the B&B Algorithm, shown in algorithm 4 (PAPADIMITRIOU; STEIGLITZ, 1998), and using cutting planes to reduce the number of branches.

Algorithm 4: Basic branch-and-bound algorithm

1: $activeSet \leftarrow \{0\}$ # *0 is the original problem*
2: $U \leftarrow \infty$
3: $currentBest \leftarrow \{\}$
4: **while** $activeSet \neq \emptyset$ **do**
5: $selectedNode \leftarrow activeSet(k)$ # *choose a node k, where* $k \in activeSet$
6: $activeSet \leftarrow activeSet - \{selectedNode\}$
7: $C \leftarrow getChildren(activeSet)$ # *generate children i and their lower bounds* z_i
8: **for** $i \in C$ **do**
9: **if** $z_i \leq U$ **then**
10: **if** *i is a complete solution* **then**
11: $U \leftarrow z_i$
12: $currentBest \leftarrow i$
13: **else**
14: $activeSet \leftarrow activeSet \cup \{i\}$
15: **end if**
16: **end if**
17: **end for**
18: **end while**

In Algorithm 4, the *activeSet* contains open search nodes, U is the upper bound and $\min_{\forall j \in activeSet} z_j$ is the lower bound. The While loop from line 4 to line 18 holds when there still exists open search nodes. In the beginning of the loop, a search node is selected to be branched. The feasible region of the problem in the selected node is divided into two, therefore generating two children nodes. The optimal solution z_i of the relaxed linear problem is computed in each children node i. If cutting planes are used to tighten the linear programming relaxations, the algorithm becomes branch-and-cut, such as in CPLEX's MIP solvers. If a *complete* solution is found (integer feasible solution) then this branch is bounded and the children becomes a fathomed. If the solution is feasible and $z_i \leq U$ it becomes the upper bound (the best integer solution), otherwise this node is killed. If an incomplete feasible solution is found, this children is stored in *activeSet* and becomes an open search node. In CPLEX MIP solvers, the search proceeds depending on the chosen criteria. For instance, one can set a *GAP* or time tolerance if he/she accepts to stop before finding an optimal integer solution.

The performance of the heuristic was measured in terms of the *GAP* between its solution and the CPLEX solution (5.1). Based on other works that compared commercial solvers and heuristics/metaheuristics (ANDRADE et al., 2015; GONCALVES; RESENDE, 2015; GUASTAROBA; SPERANZA, 2012), a runtime limit of one hour was set for all models solved by CPLEX.

$$GAP = 100 * \left(\frac{f_g - f_w}{f_w} \right) \tag{5.1}$$

Where f_w is the solution obtained by CPLEX and f_g is the solution obtained by GQ

5.3 DATABASE

The index tracking data used in this study is available from Beasley OR library at <http://people.brunel.ac.uk/$\sim$mastjjb/jeb/orlib/indtrackinfo.html>. This database was selected because, according to Section 3.2 analysis, this is the most used database by works that approach index tracking models with heuristics/metaheurstics. Details about the five instances selected from this database are shown in the first two columns of Table 21. These five instances were also used in Baykasoglu, Yunusoglu & Ozsoydan (2015) and works concerning the evaluation of metaheuristics for the index tracking problem (BEASLEY; MEADE; CHANG, 2003; RUIZ-TORRUBIANO; SUAREZ, 2009; GUASTAROBA; SPERANZA, 2012; WANG et al., 2012) to analyse the performance of the metaheuristic when the size of the index to be tracked grows.

5.4 RESULTS.

The experiments were performed using an Intel(R) Core(TM) I7 3.4 GHz with 8GB RAM. Matlab 2017b was used to implement the heuristic and as the main modelling language comprising CPLEX commercial solver API.

Instance	N	K	In-sample (weeks)	$Time_{CPLEX}$ (s)	$GAP_{CPLEX}(\%)$
Hang Seng (indtrack1)	31	10	145	3.73	0
DAX100 (indtrack2)	85	10	145	3619.58	24.69
FTSE100 (indtrack3)	89	10	145	3641.11	41.02
S&P100 (indtrack4)	98	10	145	3606.02	40.20
Nikkei225 (indtrack5)	225	10	145	3604.87	85.48

Table 21 – Instances and model information

5.4.1 CPLEX solutions.

Table 21 summarizes all instances, number of stocks N associated, portfolio cardinality K adopted, total elapsed CPU time in CPLEX solver and GAP_{CPLEX} obtained by each one.

For the index tracking model adopted in this study, only ‘indtrack1’ had its optimal integer solution found by CPLEX in less than 1 hour. GAP_{CPLEX} is an output provided by CPLEX after the end of the optimization. It indicates the percentage difference between the objective value of the best feasible integer solution found and the minimum of the objective values of the relaxed index tracking problem in the active set of nodes. So, if GAP_{CPLEX} is equal to 5%, then CPLEX has found a feasible integer solution which is proved to be within 5% of the integer optimal solution.

5.4.2 Experiments to Define Parameters and to Compare MiLS algorithms.

Solution construction phase and MiLS efficiency depends on how their parameters and greedy functions are set. That efficiency will affect the quality of solutions to be input into MaLS and also the time for it to compute a final solution. Some experiments were performed for both types of solution construction: greedy and random, thereby aiming to examine the effects of the chosen greedy function and local features.

Section 5.4.2.1 presents the greedy construction evaluation and Section 5.4.2.2 presents the random construction evalutaion. Two groups of experiments will be performed in both sections 5.4.2.1 and 5.4.2.2. The first group of experiments consisted in adopting the ‘indtrack1’ instance to define parameters for GQ and at the same time compares both MiLS Algorithms. The second group of experiments evaluated the best parameter configuration and MiLS algorithms against other problem instances (different values of N).

Each experiment inside a group consists of 30 runs of GQ. As discussed in 4.3, the quality of the solution was measured using the mean of the GAP and the precision of the heuristic is indicated by the standard deviation of GAP. If the mean GAP is big and the GAP standard deviation is very small, this can indicate that the search is stuck in a local optimum. To measure the cost of a solution, CPU time mean and standard deviation were observed.

5.4.2.1 Results for Greedy Parameter Sets.

Table 22 summarizes all parameter sets (columns) and the associated parameter values used (lines) and Table 23 shows the results for each parameter set and for both MiLS algorithms used by GQ in instance 1.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
$iter_{MAX}$	50	100	50	50	50	50	50	50
RCL_{size}	0.35N	0.35N	0.25N	0.45N	0.45N	0.45N	0.45N	0.45N
p_{MiLS}	0.4	0.4	0.4	0.4	0.2	0.6	0.4	0.4
UB_{MiLS}	0.15R	0.15R	0.15R	0.15R	0.15R	0.15R	0.25R	0.15R
ϕ	3	3	3	3	3	3	3	2

$*R = RCL_{size}$

Table 22 – Greedy parameter sets

The first two parameters set were used to evaluate how the number of iterations impact the quality and cost of the trial solution, using a base value of 50 iterations, which is not too time consuming, and these are doubled to 100 iterations to verify if a significant impact on the mean GAP occurred. Note from Table 23 that doubling the number of maximum iterations doubles the mean CPU time for both types of MiLS and reduces only 4% of the GAP. This result shows that it is not worth doubling the maximum number of iterations, so, for the following experiments, parameter set 1 will be put to test.

Parameter sets 1, 3 and 4 were compared to evaluate the impacts of a high, medium and low, respectively, greedy RCL_{size} in the trial solution. It was possible to obtain better solutions for both MiLS algorithms using a less greedy RCL_{size}, this being worth the increase in μ(CPUtime) for the ϕ-DMiLS algorithm. Thus, parameter set 4 is the best choice until this point. It is important for the reader to understand that the term 'better solution' used here refers to finding a solution that dominates the reference solution with respect to all the four performance metrics: smallest μ(GAP) and σ(GAP), smallest μ(CPUTime) and σ(CPUTime).

The MiLS parameters were set using the same comparison procedure. The results for parameter sets 4, 5 and 6, containing medium, lower and higher values of p_{MiLS}, respectively, were compared in order to determine which set could offer the best solutions. p_{MiLS} included in parameter set 4 produced the best solutions, since it balances μ(GAP) and μ(CPUtime). Now it will be ascertained if an increase in UB_{MiLS} can produce better solutions. For this verification, results from parameter sets 4 and 7 were compared, where the last set has a higher UB_{MiLS} value. It was concluded that a lower UB_{MiLS} produces the best solutions for both algorithms. Finally, parameter sets 4 and 8 were compared to

Par Set	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
1	GQ	57.981	6.566	7.087	0.294
	ϕ-D	40.086	8.966	12.871	1.379
2	GQ	54.901	5.609	14.226	0.385
	ϕ-D	36.040	11.317	25.241	1.748
3	GQ	58.421	10.389	6.376	0.331
	ϕ-D	44.289	9.185	11.196	0.994
4	GQ	53.766	8.599	7.529	0.518
	ϕ-D	27.468	7.747	19.974	2.869
5	GQ	59.837	8.518	6.224	0.384
	ϕ-D	33.839	11.643	12.109	2.595
6	GQ	53.684	6.613	8.443	0.376
	ϕ-D	21.496	7.244	27.667	2.941
7	GQ	56.781	8.437	8.524	0.709
	ϕ-D	17.847	5.740	30.790	3.735
8	GQ	53.766	8.599	7.529	0.518
	($\phi = 2$)-D	27.468	7.747	19.974	2.869
	($\phi = 3$)-D	29.437	10.386	16.848	2.007

Table 23 – Results for greedy parameter sets and the MiLS algorithm

verify what happens when a lower ϕ value is used. In this case, nothing could be concluded as the solutions produced by both the parameter sets are similar. Then, both ϕ values will be used in the next group of experiments. In Figure 21 we can visualize the greedy parameter configuration results.

Next, parameter set 4 and 8 were tested in different instances. This experiment was performed to evaluate the efficiency of the parameters and MiLS algorithms for larger values of N. The results are shown in Table 24.

Note that no matter the value of N, ϕ-DMiLS will always produce a lower μ(GAP) relative to GQ-MiLS. Although ϕ-DMiLS produces a better μ(GAP), it does so at the cost of a relatively high computational effort. Again, results for the two values of ϕ are similar. Thus, $\phi = 3$ (parameter set 4) was selected for the experiments in section 4.3, because it produces slightly better solutions in relation to $\phi = 2$. Figure 22 summarizes GAP and CPU time results for other instances. The associated MiLS column of parameter set 8 in Figure 22 is equal to the MiLS column of parameter set 4 because this parameter set only changes ϕ, which is a parameter that only exists in the new ϕ-DMiLS.

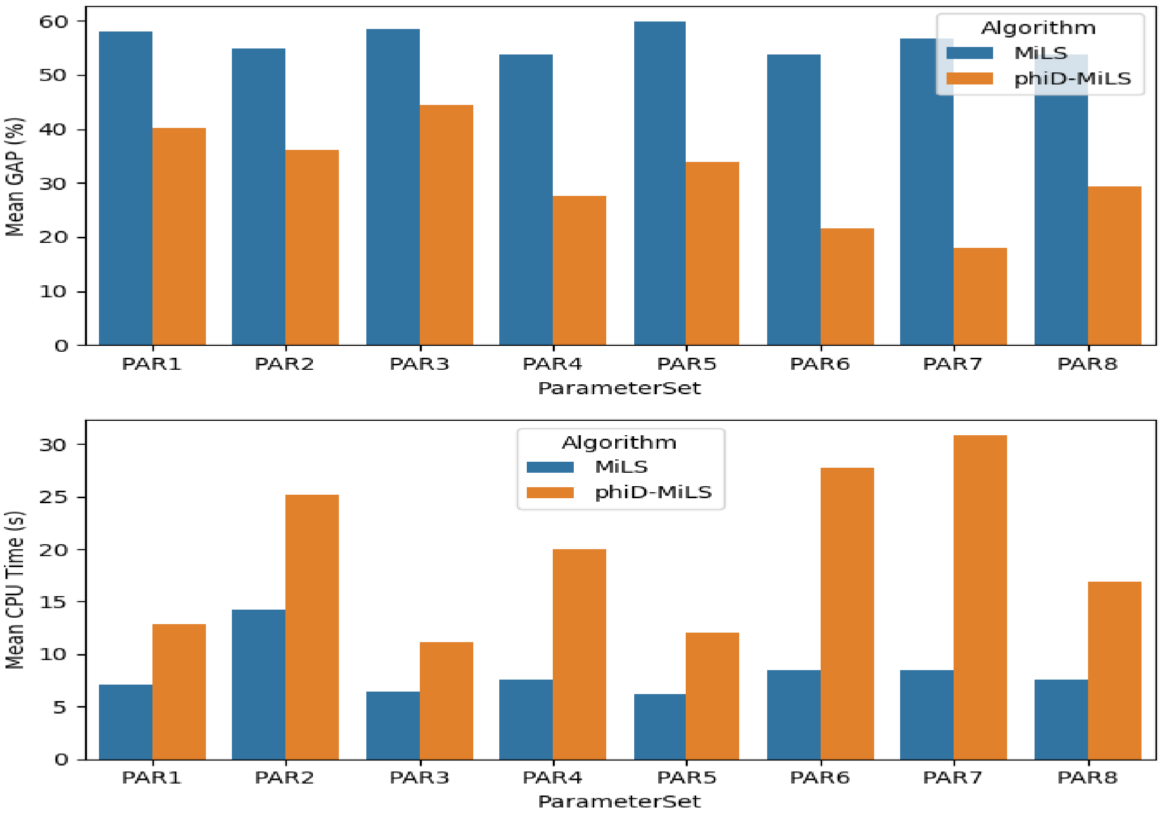

Figure 21 – Semi-greedy configuration results

N	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
85	GQ	84.681	11.669	13.162	0.942
	$\phi = 3$	26.086	5.802	65.560	11.957
	$\phi = 2$	27.651	6.526	62.341	10.022
89	GQ	61.475	7.689	14.231	1.241
	$\phi = 3$	28.948	5.743	61.472	7.737
	$\phi = 2$	30.294	6.049	60.560	8.654
98	GQ	72.641	8.152	15.082	1.359
	$\phi = 3$	30.154	4.411	71.556	10.310
	$\phi = 2$	34.021	6.541	66.052	9.598
225	GQ	61.210	7.614	33.277	2.618
	$\phi = 3$	27.094	4.422	196.010	37.580
	$\phi = 2$	30.059	3.531	173.683	26.571

Table 24 – Results of semi-greedy construction (parameter sets 4 and 8) and MiLS algorithms for different values of N

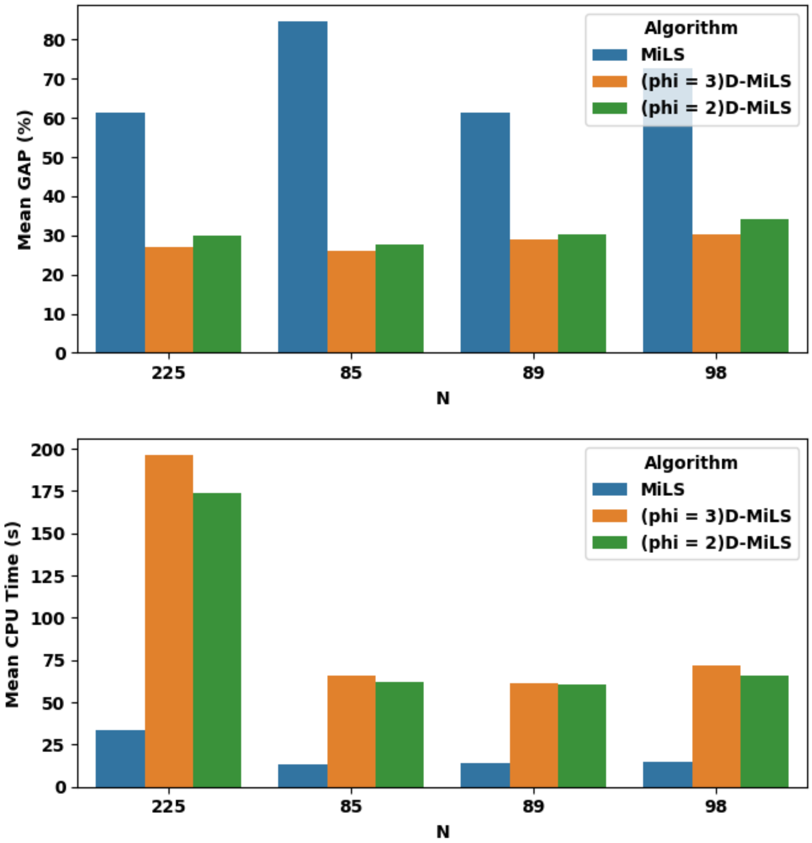

Figure 22 – Semi-greedy configuration results for other instances

5.4.2.2 Results for Random Parameter Sets.

Table 25 summarizes all random parameter sets considered. Only the RCL_{size} value was changed in these experiments, since MiLS parameters have already been defined in Section 5.4.2. Table 26 shows the results for each parameter set and for both MiLS algorithms used by GQ.

	1	**2**	**3**
$iter_{MAX}$	50	50	50
RCL_{size}	N	0,85N	0,75N
p_{MiLS}	0,4	0,4	0,4
UB_{MiLS}	0,15R	0,15R	0,15R
ϕ	3	3	3
MaLS	Off	Off	Off

Table 25 – Random parameter sets

Figure 23 summarizes the results for GAP and CPU time associated with random construction in instance 1. GQ-MiLS combined with random solution construction seems to produce almost the same mean and standard deviation for either GAP and computational

	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
1	GQ	49.189	10.883	10.351	0.872
	ϕ-D	13.729	5.991	47.257	5.898
2	GQ	47.868	15.383	9.573	0.734
	ϕ-D	18.599	6.326	34.706	5.116
3	GQ	48.794	13.492	9.708	0.769
	ϕ-D	16.205	8.049	36.065	5.073

Table 26 – Results for random parameter sets and MiLS algorithm

N	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
85	GQ	71.621	10.091	19.885	2.508
	ϕ-D	15.826	5.077	138.170	25.681
89	GQ	52.279	8.012	21.585	2.780
	ϕ-D	21.605	3.920	129.870	27.312
98	GQ	66.620	7.650	20.782	2.550
	ϕ-D	20.595	3.551	145.937	21.009
225	GQ	61.647	6.366	46.222	7.737
	ϕ-D	23.040	4.151	340.673	47.296

Table 27 – Results of random construction (random parameter set 3) and MiLS algorithms for different values of N

time for any trial solutions independent of the random parameter set. Also, better μ(GAP) values were achieved but the precision was reduced (σ(GAP) increased), obviously, since the selection of stock is more random. On the other hand, now ϕ-DMILS solutions are even more expensive, though they produce lower μ(GAP). Randomness hinders the search and, as a consequence, more effort is made by ϕ-DMILS to improve solutions. Another feature of ϕ-DMILS is that it produces similar precision values for either greedy or random constructions. Parameter set 3 was adopted for the following experiments, so as to reduce the level of randomness, thereby avoiding a completely blind search. Figure 24 summarizes GAP and CPU time results for other instances.

Table 27 presents the results obtained by running GQ with random parameter set 3 for different values of N.

The superiority of ϕ-DMiLS relative to GQ-MiLS in relation to μ(GAP) remains. As expected, the ϕ-DMiLS computational effort increased when compared to results in Table 23. GQ-MiLS μ(CPUtime) also increased for all instances.

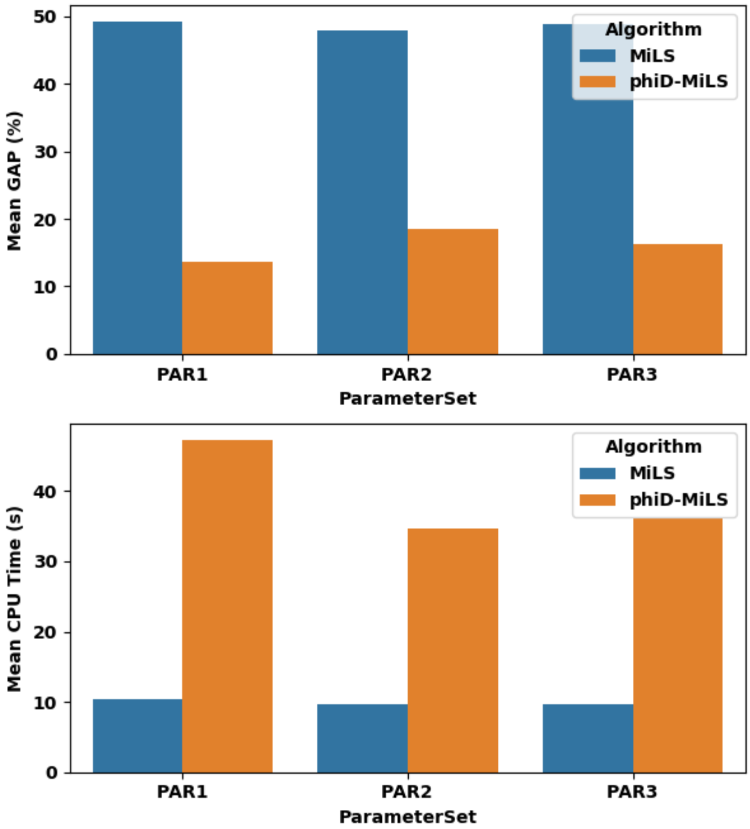

Figure 23 – Random configuration results

5.4.3 Evaluating the Final Solution.

This last group of experiments evaluates how MiLS impacts MaLS. If spending more time on MiLS can be compensated by getting better and faster results in MaLS, then it is worth paying the computational cost required for the ϕ-DMiLS algorithm. Now that parameters were defined for each search mode, the best MiLS algorithm can be obtained combining it with MaLS. The parameter 'random fit', associated with MaLS (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015), was not used in the final experiments due to bad results in preliminary tests.

Because of the expensive runtime required for these experiments, only three instances were used: 'indtrack 1'; 'indtrack 4' also representing 'indtrack 2' and 'indtrack 3', because they have almost the same number of assets that comprise the index, and 'indtrack 5'.Results for the final tests are shown in Table 28 and Table 29.

On comparing results from Table 28 and Table 29, it is inferred that as N gets higher, the importance of the greedy function also gets higher. Greedy construction found similar solutions in terms of quality in less time than random construction did, for both MiLS algorithms when $N = 98$ and $N = 225$. It also can be observed that the use ϕ-Depth MiLS, independent of the construction type, reduces the time interval to compute a final solution and it can still can get a higher quality solution relative to GQ-MiLS. This leads

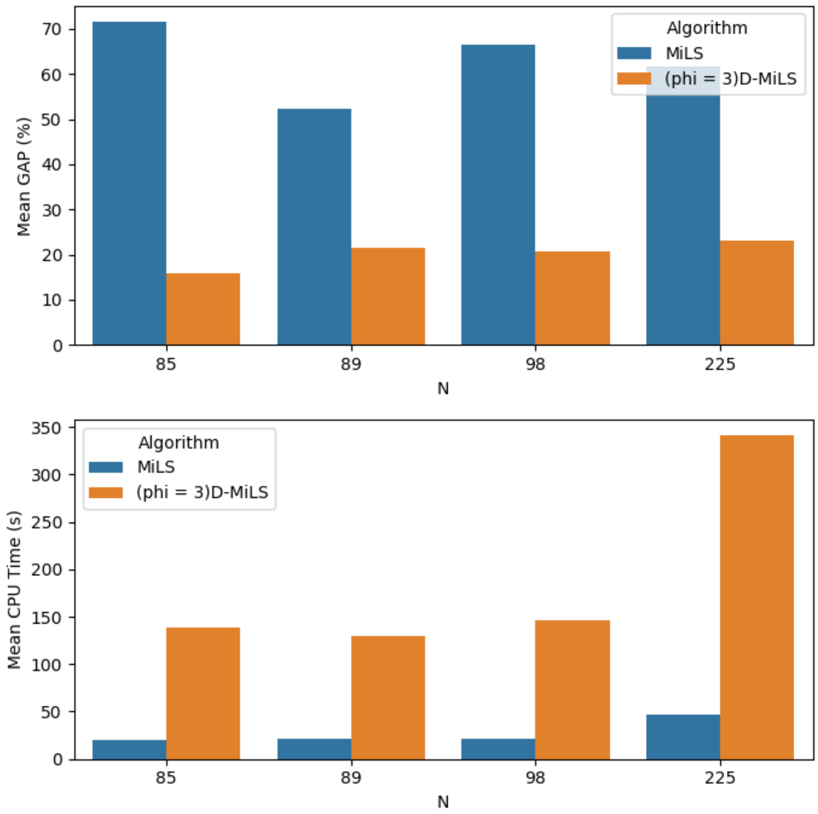

Figure 24 – Random configuration results for other instances

N	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
31	GQ	0.576	1.309	93.07	17.31
	ϕ-Depth	0.648	1.490	86.53	17.09
98	GQ	8.302	6.893	478.00	134.68
	ϕ-Depth	5.712	3.720	369.97	93.88
225	GQ	12.478	4.676	1066.74	249.58
	ϕ-Depth	11.604	5.199	841.37	222.12

Table 28 – Result of MiLS (greedy construction) and MaLS combination

to the conclusion that a greedy ϕ-Depth MiLS is the suitable option for the problem.

The measures $\mu(GAP)$ and $\mu(CPUtime)$ showed that GQ can produce good solutions in less time in comparison to the commercial solver CPLEX. For instance, in Table 28 a $\mu(GAP) = 5.72\%$ was achieved for $N = 98$ in a mean time of 7 minutes. In other words, GQ, in about 7 minutes, could get a solution almost as good as the solution that was found by CPLEX in a time interval of 1 hour. Figures 25 and 26 show the final results for greedy and random configurations, respectively.

N	MiLS	μ(GAP)(%)	σ(GAP)(%)	μ(CPUtime)(s)	σ(CPUtime)(s)
31	GQ	0.230	0.876	93.09	17.03
	ϕ-Depth	0.612	1.403	87.19	11.68
98	GQ	7.294	6.193	488.75	119.02
	ϕ-Depth	4.220	3.779	425.03	108.71
225	GQ	14.590	6.739	1079.76	304.05
	ϕ-Depth	12.917	3.514	933.07	266.83

Table 29 – Result of MiLS (random construction) and MaLS combination

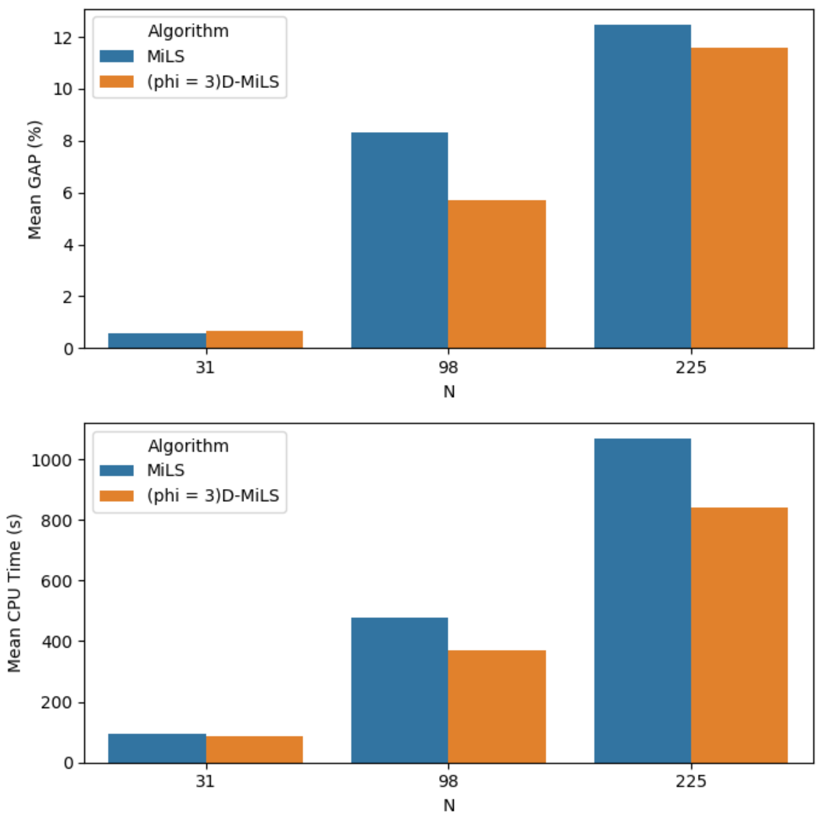

Figure 25 – Final results for semi-greedy configuration

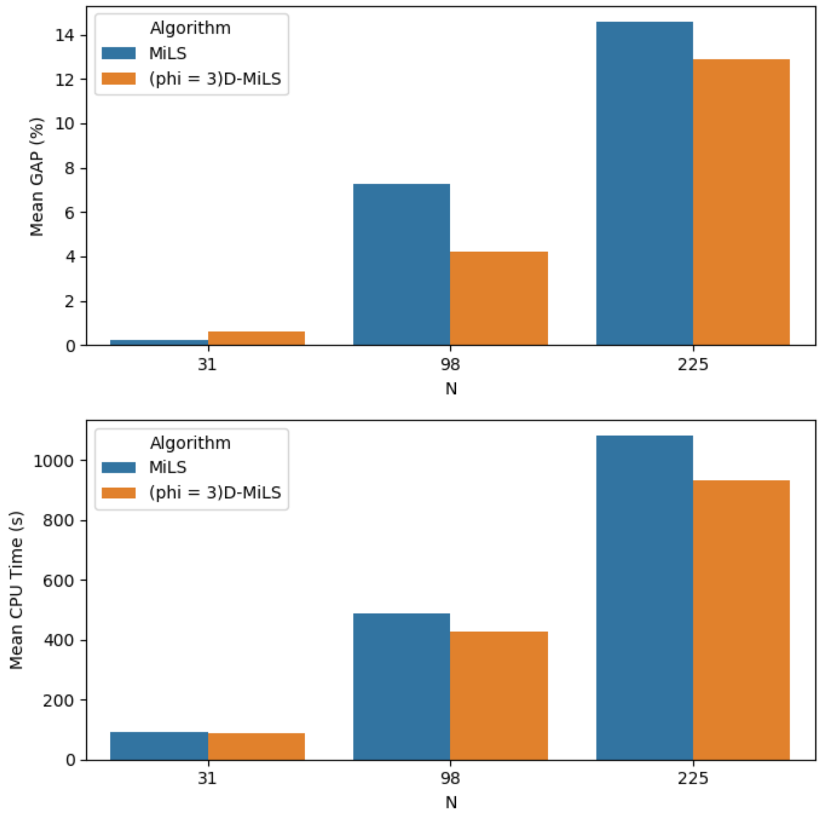

Figure 26 – Final results for random configuration

5.5 CHAPTER CONCLUSION

This chapter presented the results of the proposed GRASP metaheuristic adapted for the index tracking problem. The adopted database was OR-library (BEASLEY; MEADE; CHANG, 2003) and CPLEX was selected as a benchmark against the heuristic. The results showed that the proposed GRASP adaptation outperforms the commercial solver CPLEX. The semi-greedy solution results showed that better final solutions are obtained due to the new greedy function proposed. Besides, since the modified local search component provided better solutions for MaLS it produced better final solutions than that of the original local search component. In the next chapter, conclusions about this work are drawn.

6 CONCLUSIONS AND FUTURE WORK

With the growth of heuristic and non-heuristic solution approaches applied to the index tracking problem in the last decade, journals encompassing operational research and computer science areas were the most promising destinations for research papers concerning this problem. Despite their relatively low production rate in the last decade, heuristics/metaheuristics have more citation impact than non-heuristic solution approaches. Also, it was possible to verify that heuristics/metaheuristics solution approaches are concentrated in the MINLP quantitative modeling framework.

Most of the papers that adopted heuristics/metaheuristics as a solution approach implemented genetic and differential evolution algorithms to their index tracking formulation. Not only pure heuristics were applied to this problem, but also hybridizations using both heuristics and commercial solvers. CPU time and correlation with respect to the index were one of the most used evaluation metrics to indicate heuristics/metaheuristics success. These evaluation metrics were used most often when comparing different heuristics/metaheuristics or when using the CPLEX solver as the main benchmark solution method.

A diversity of objective functions was found among works in index tracking models, such as minimization of tracking error measures, transaction costs minimization and return maximization. A good part of the objective functions used in the formulations solved by heuristics solution approaches consisted of two tracking error measures: Root Mean Squared Error and Mean Squared Error. Although different types of practical constraints were used in index tracking problems solved by heuristic methods in the last decade, such as round-lot and CVaR, there is no prevalence of using any of these. With respect to databases adopted, the most used is the OR-library followed by Datastream. The former database contains public data and the later is a commercial database.

For this work, the selected index tracking model was (WANG et al., 2012) because is best fit for building a base comparison of the GQ metaheuristic, also, there is no loss of generality, since GQ not only can be applied to other models with the same set of constraints but also evaluated and tested in models adopting different objective functions. The adopted database was OR-library because it is widely used in the index tracking literature and is publicly available. The commercial solver CPLEX was selected because it is the most used solver for benchmarking against heuristic solution methods in index tracking problems. The metaheuristic GQ (BAYKASOGLU; YUNUSOGLU; OZSOYDAN, 2015) was adopted because it was the first GRASP adapted for the cardinality constrained financial portfolio optimization. An adaptation of the GQ greedy function for the tracking error criterion was developed, considering that GQ was originally designed for the mean-variance cardinality constrained model. Also, a local search component of GQ was modified and evaluated.

The experiments performed in this work showed that GRASP has great potential in portfolio optimization, more specifically in solving index tracking problems. With a manual parameter tuning procedure it was possible to obtain $\mu(GAP)$ values less than 12% at time intervals of less than 1 hour, using fixed parameters of the GQ for all instances.

Comparing ϕ-DMiLS trial solutions with GQ-MiLs trial solutions it is evident that the better the quality of the solutions obtained before MaLS, the faster it will compute a good quality final solution. The computational time of ϕ-DMiLS before MaLS is compensated because it produces higher quality solutions. These experiments also put the formulation used for the greedy function to test. It was noticed that on using this greedy function formulation, a time-relative gain was achieved as N grows.

6.1 FUTURE WORK

As to future lines of research, there are many other index tracking formulations, containing practical and robust constraints, and other instances which can be explored and experimented with using GRASP. Future works include the comparison with other heuristics in experiments, such as those presented in Amorim et al. (2020), that have not been concluded in time to be presented in this dissertation, although it shall appear in future publications. The reformulation of greedy functions is also important to increase the chances of finding near-optimal solutions in a shorter time interval and to test GRASP adaptations in other objective functions. Furthermore, it is interesting to implement search increments in the proposed metaheuristic. The path-relinking mechanism can provide a long-term memory structure for GRASP (RESENDE; RIBEIRO, 2016; MARTI; PARDALOS; RESENDE, 2018), and can be incremented with restart strategies (RESENDE; RIBEIRO, 2011). Also, instead of using manual parameter adjustments, it is interesting to explore the automated tune-in of the parameters of the GRASP approach applied to the index tracking problem by adopting reactive search procedures (RESENDE; RIBEIRO, 2016; MARTI; PARDALOS; RESENDE, 2018) and Iterated F-Race (IRace) (LóPEZ-IBáñEZ et al., 2016).

REFERENCES

ACOSTA-GONZALEZ, E.; ARMAS-HERRERA, R.; FERNANDEZ-RODRIGUEZ, F. On the index tracking and the statistical arbitrage choosing the stocks by means of cointegration: the role of stock picking. *Quantitative Finance*, 15, n. 6, SI, p. 1075–1091, 2015. ISSN 1469-7688.

AFFOLTER, K.; HANNE, T.; SCHWEIZER, D.; DORNBERGER, R. Invasive weed optimization for solving index tracking problems. *Soft Computing*, 20, n. 9, SI, p. 3393–3401, 2016. ISSN 1432-7643.

AKBAY, M. A.; KALAYCI, C. B.; POLAT, O. A parallel variable neighborhood search algorithm with quadratic programming for cardinality constrained portfolio optimization. *Knowledge-based Systems*, 198, 2020. ISSN 0950-7051.

AMORIM, T. W.; SILVA, J. C. S.; SILVA, D. F. de L.; ALMEIDA-FILHO, A. T. de. Comparação e avaliação de algoritmos genéticos em modelos de index tracking aplicados ao ibovespa. *Anais do Simpósio Brasileiro de Pesquisa Operacional*, 2020.

ANAGNOSTOPOULOS, K.; CHATZOGLOU, P.; KATSAVOUNIS, S. A reactive greedy randomized adaptive search procedure for a mixed integer portfolio optimization problem. *Managerial Finance*, Emerald Group Publishing Ltd., v. 36, n. 12, p. 1057–1065, 2010. ISSN 03074358. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85015678785&doi=10.1108%2f03074351011088450&partnerID=40&md5=a6535dd5b5801a3557835286ae23c65c>.

ANDRADE, C. E. de; TOSO, R. F.; RESENDE, M. G. C.; MIYAZAWA, F. K. Biased Random-Key Genetic Algorithms for the Winner Determination Problem in Combinatorial Auctions. *Evolutionary Computation*, 23, n. 2, p. 279–307, 2015. ISSN 1063-6560.

ANDRIOSOPOULOS, D.; DOUMPOS, M.; PARDALOS, P. M.; ZOPOUNIDIS, C. Computational approaches and data analytics in financial services: A literature review. *Journal Of The Operational Research Society*, 70, n. 10, SI, p. 1581–1599, 2019. ISSN 0160-5682.

ANDRIOSOPOULOS, K.; DOUMPOS, M.; PAPAPOSTOLOU, N. C.; POULIASIS, P. K. Portfolio optimization and index tracking for the shipping stock and freight markets using evolutionary algorithms. *Transportation Research Part E-logistics and Transportation Review*, 52, n. SI, p. 16–34, 2013. ISSN 1366-5545.

ANDRIOSOPOULOS, K.; NOMIKOS, N. 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, n. 2, p. 571–582, 2014. ISSN 0377-2217.

ARAUJO, R. d. A.; NEDJAH, N.; OLIVEIRA I, A. L.; MEIRA, S. R. d. L. A deep increasing-decreasing-linear neural network for financial time series prediction. *Neurocomputing*, 347, p. 59–81, 2019. ISSN 0925-2312.

ARAUJO, R. d. A.; NEDJAH, N.; SEIXAS, J. M. de; OLIVEIRA, A. L. I.; MEIRA, S. R. d. L. Evolutionary-morphological learning machines for high-frequency financial time series prediction. *Swarm and Evolutionary Computation*, 42, p. 1–15, 2018. ISSN 2210-6502.

ARIA, M.; CUCCURULLO, C. bibliometrix: An r-tool for comprehensive science mapping analysis. *Journal of Informetrics*, Elsevier, v. 11, n. 4, p. 959–975, 2017. Disponível em: <https://doi.org/10.1016/j.joi.2017.08.007>.

ARTZNER, P.; DELBAEN, F.; EBER, J.; HEATH, D. Coherent measures of risk. *Mathematical Finance*, 9, n. 3, p. 203–228, 1999. ISSN 0960-1627.

B3. *B3: Brasil, Bolsa, Balcão*. 2019. Disponível em: <http://www.b3.com.br/pt_br>.

BABAZADEH, H.; ESFAHANIPOUR, A. A novel multi period mean-VaR portfolio optimization model considering practical constraints and transaction cost. *Journal of Computational and Applied Mathematics*, 361, p. 313–342, 2019. ISSN 0377-0427.

BACEVIC, A.; VILIMONOVIC, N.; DABIC, I.; PETROVIC, J.; DAMNJANOVIC, D.; DZAMIC, D. Variable neighborhood search heuristic for nonconvex portfolio optimization. *Engineering Economist*, 64, n. 3, SI, p. 254–274, 2019. ISSN 0013-791X.

BARRO, D.; CANESTRELLI, E. Downside risk in multiperiod tracking error models. *Central European Journal of Operations Research*, 22, n. 2, SI, p. 263–283, 2014. ISSN 1435-246X.

BARRO, D.; CANESTRELLI, E.; CONSIGLI, G. Volatility versus downside risk: performance protection in dynamic portfolio strategies. *Computational Management Science*, 16, n. 3, p. 433–479, 2019. ISSN 1619-697X.

BAYKASOGLU, A.; YUNUSOGLU, M. G.; OZSOYDAN, F. B. A GRASP based solution approach to solve cardinality constrained portfolio optimization problems. *Computers & Industrial Engineering*, 90, p. 339–351, 2015. ISSN 0360-8352.

BEASLEY, J.; MEADE, N.; CHANG, T. An evolutionary heuristic for the index tracking problem. *European Journal of Operational Research*, 148, n. 3, p. 621–643, 2003. ISSN 0377-2217.

BEASLEY, J. E. Portfolio optimisation: Models and solution approaches. In: _____. *Theory Driven by Influential Applications*. [s.n.], 2013. cap. Chapter 11, p. 201–221. Disponível em: <https://pubsonline.informs.org/doi/abs/10.1287/educ.2013.0114>.

BENIDIS, K.; FENG, Y.; PALOMAR, D. P. Sparse Portfolios for High-Dimensional Financial Index Tracking. *IEEE Transactions on Signal Processing*, 66, n. 1, p. 155–170, 2018. ISSN 1053-587X.

BERTSEKAS, D. P. *Nonlinear Programming*. Belmont, Massachusetts: Athena Scientific, 1999.

BILBAO-TEROL, A.; ARENAS-PARRA, M.; CANAL-FERNANDEZ, V. A fuzzy multi-objective approach for sustainable investments. *Expert Systems With Applications*, 39, n. 12, p. 10904–10915, 2012. ISSN 0957-4174.

BOUDT, K.; WAN, C. The effect of velocity sparsity on the performance of cardinality constrained particle swarm optimization. *Optimization Letters*, 14, n. 3, p. 747–758, 2020. ISSN 1862-4472.

BURIOL, L. S.; FIGUEIREDO, C.; RESENDE, M. G.; UCHOA, E. The guide to np-completeness is 40 years old: An homage to david s. johnson. *Pesquisa Operacional*, 2020, no prelo.

CACADOR, S.; DIAS, J. M.; GODINHO, P. Global minimum variance portfolios under uncertainty: a robust optimization approach. *Journal of Global Optimization*, 76, n. 2, SI, p. 267–293, 2020. ISSN 0925-5001.

CANAKGOZ, N. A.; BEASLEY, J. E. Mixed-integer programming approaches for index tracking and enhanced indexation. *European Journal of Operational Research*, 196, n. 1, p. 384–399, 2009. ISSN 0377-2217.

CHANG, T.; MEADE, N.; BEASLEY, J.; SHARAIHA, Y. Heuristics for cardinality constrained portfolio optimisation. *Computers & Operations Research*, 27, n. 13, p. 1271–1302, 2000. ISSN 0305-0548.

CHEN, C.; KWON, R. H. Robust portfolio selection for index tracking. *Computers & Operations Research*, 39, n. 4, SI, p. 829–837, 2012. ISSN 0305-0548.

CHEN, W.; LI, D.; LIU, Y.-J. A Novel Hybrid ICA-FA Algorithm for Multiperiod Uncertain Portfolio Optimization Model Based on Multiple Criteria. *IEEE Transactions on Fuzzy Systems*, 27, n. 5, SI, p. 1023–1036, 2019. ISSN 1063-6706.

CHENG, B.; CHEN, Z.; LIU, J. MULTI-STAGE FINANCIAL INDEX TRACKING MODEL UNDER GH DISTRIBUTION. *Pakistan Journal of Statistics*, 29, n. 5, p. 795–810, 2013. ISSN 1012-9367.

CHIAM, S. C.; TAN, K. C.; MAMUN, A. A. Dynamic index tracking via multi-objective evolutionary algorithm. *Applied Soft Computing*, 13, n. 7, p. 3392–3408, 2013. ISSN 1568-4946.

CONSIGLIO, A.; ZENIOS, S. Integrated simulation and optimization models for tracking international fixed income indices. *Mathematical Programming*, v. 89, p. 311–339, 2001.

DJOKO, D. T.; TILLE, Y. Selection of balanced portfolios to track the main properties of a large market. *Quantitative Finance*, 15, n. 2, SI, p. 359–370, 2015. ISSN 1469-7688.

DUTTA, S.; BISWAL, M. P.; ACHARYA, S.; MISHRA, R. Fuzzy stochastic price scenario based portfolio selection and its application to BSE using genetic algorithm. *Applied Soft Computing*, 62, p. 867–891, 2018. ISSN 1568-4946.

EFTEKHARIAN, S. E.; SHOJAFAR, M.; SHAMSHIRBAND, S. 2-Phase NSGA II: An Optimized Reward and Risk Measurements Algorithm in Portfolio Optimization. *Algorithms*, 10, n. 4, 2017. ISSN 1999-4893.

ERTENLICE, O.; KALAYCI, C. B. A survey of swarm intelligence for portfolio optimization: Algorithms and applications. *Swarm and Evolutionary Computation*, 39, p. 36–52, 2018. ISSN 2210-6502.

FABOZZI, F. J.; KOLM, P. N.; PACHAMANOVA, D. A.; FOCARDI, S. M. *Robust Portfolio Optimization and Management*. Hoboken, New Jersey: John Wiley & Sons, 2007.

FAIRBROTHER, A.; TURNER, A.; WALLACE, S. W. Scenario Generation for Single-Period Portfolio Selection Problems with Tail Risk Measures: Coping with High Dimensions and Integer Variables. *Informs Journal on Computing*, 30, n. 3, p. 472–491, 2018. ISSN 1091-9856.

FAMA, E. F. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, v. 25, n. 2, p. 383, 1970. Disponível em: <https://doi.org/10.2307/2325486>.

FASTRICH, B.; PATERLINI, S.; WINKER, P. Cardinality versus q-norm constraints for index tracking. *Quantitative Finance*, 14, n. 11, p. 2019–2032, 2014. ISSN 1469-7688.

FEO, T.; RESENDE, M. GREEDY RANDOMIZED ADAPTIVE SEARCH PROCEDURES. *Journal of Global Optimization*, 6, n. 2, p. 109–133, 1995. ISSN 0925-5001.

FEO, T. A.; RESENDE, M. G. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, v. 8, n. 2, p. 67 – 71, 1989. ISSN 0167-6377. Disponível em: <http://www.sciencedirect.com/science/article/pii/0167637789900023>.

FEO, T. A.; RESENDE, M. G. C.; SMITH, S. H. A greedy randomized adaptive search procedure for maximum independent set. *Operations Research*, INFORMS, v. 42, n. 5, p. 860–878, 1994. ISSN 0030364X, 15265463. Disponível em: <http://www.jstor.org/stable/171545>.

FERREIRA, L.; BORENSTEIN, D.; RIGHI, M. B.; FILHO, A. T. de A. A fuzzy hybrid integrated framework for portfolio optimization in private banking. *Expert Systems With Applications*, 92, p. 350–362, 2018. ISSN 0957-4174.

FESTA, P.; RESENDE, M. G. C. An annotated bibliography of GRASP - Part I: Algorithms. *International Transactions in Operational Research*, 16, n. 1, p. 1–24, 2009. ISSN 0969-6016.

FESTA, P.; RESENDE, M. G. C. An annotated bibliography of GRASP-Part II: Applications. *International Transactions in Operational Research*, 16, n. 2, p. 131–172, 2009. ISSN 0969-6016.

FLOUDAS, C. A. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Princeton, New Jersey: Oxford University Press, 1995.

GAIVORONSKI, A.; KRYLOV, S.; WIJST, N. van der. Optimal portfolio selection and dynamic benchmark tracking. *European Journal of Operational Research*, 163, n. 1, p. 115–131, 2005. ISSN 0377-2217.

GARCIA, F.; GUIJARRO, F.; MOYA, I. The curvature of the tracking frontier: A new criterion for the partial index tracking problem. *Mathematical and Computer Modelling*, 54, n. 7-8, p. 1781–1784, 2011. ISSN 0895-7177.

GARCIA, F.; GUIJARRO, F.; MOYA, I. A Multiobjective Model for Passive Portfolio Management: an Application on the S&P 100 Index. *Journal of Business Economics and Management*, 14, n. 4, p. 758–775, 2013. ISSN 1611-1699.

GARCIA, F.; GUIJARRO, F.; OLIVER, J. Index tracking optimization with cardinality constraint: a performance comparison of genetic algorithms and tabu search heuristics. *Neural Computing & Applications*, 30, n. 8, p. 2625–2641, 2018. ISSN 0941-0643.

GAREY, M. R.; JOHNSON, D. S. *Computers and Intractability: A guide to the theory of NP-Completeness*. New York: W. H. Freeman, 1979.

GENDREAU, M.; POTVIN, J.-Y. (Ed.). *Handbook of Metaheuristics*. [S.l.]: Springer US, 2010.

GIUZIO, M. Genetic algorithm versus classical methods in sparse index tracking. *Decisions in Economics and Finance*, 40, n. 1-2, SI, p. 243–256, 2017. ISSN 1593-8883.

GIUZIO, M.; FERRARI, D.; PATERLINI, S. Sparse and robust normal and t- portfolios by penalized Lq-likelihood minimization. *European Journal of Operational Research*, 250, n. 1, p. 251–261, 2016. ISSN 0377-2217.

GOEL, A.; SHARMA, A.; MEHRA, A. Index tracking and enhanced indexing using mixed conditional value-at-risk. *Journal of Computational and Applied Mathematics*, 335, p. 361–380, 2018. ISSN 0377-0427.

GONCALVES, J. F.; RESENDE, M. G. C. A biased random-key genetic algorithm for the unequal area facility layout problem. *European Journal of Operational Research*, 246, n. 1, p. 86–107, 2015. ISSN 0377-2217.

GONG, C.; XU, C.; ANDO, M.; XI, X. A New Method of Portfolio Optimization Under Cumulative Prospect Theory. *Tsinghua Science and Technology*, 23, n. 1, p. 75–86, 2018. ISSN 1007-0214.

GRISHINA, N.; LUCAS, C. A.; DATE, P. Prospect theory-based portfolio optimization: an empirical study and analysis using intelligent algorithms. *Quantitative Finance*, 17, n. 3, p. 353–367, 2017. ISSN 1469-7688.

GRÖTSCHEL, M.; LOVáSZ, L.; SCHRIJVER, A. *Geometric Algorithms and Combinatorial Optimization*. Berlin: Springer, 1988.

GUASTAROBA, G.; SPERANZA, M. G. Kernel Search: An application to the index tracking problem. *European Journal of Operational Research*, 217, n. 1, p. 54–68, 2012. ISSN 0377-2217.

GUPTA, P.; MEHLAWAT, M. K.; YADAV, S.; KUMAR, A. Intuitionistic fuzzy optimistic and pessimistic multi-period portfolio optimization models. *Soft Computing*, 24, n. 16, p. 11931–11956, 2020. ISSN 1432-7643.

HUANG, J.; LI, Y.; YAO, H. Index tracking model, downside risk and non-parametric kernel estimation. *Journal of Economic Dynamics & Control*, 92, n. SI, p. 103–128, 2018. ISSN 0165-1889.

IBM. *Branch-and-cut in CPLEX*. 2020. Disponível em: <https://www.ibm.com/support/knowledgecenter/SSSA5P_12.10.0/ilog.odms.ide.help/refcppopl/html/branch.html>.

JALOTA, H.; THAKUR, M. Genetic algorithm designed for solving portfolio optimization problems subjected to cardinality constraint. *International Journal of System Assurance Engineering and Management*, 9, n. 1, SI, p. 294–305, 2018. ISSN 0975-6809.

JORION, P. Portfolio optimization with tracking-error constraints. *Financial Analysts Journal*, 59, n. 5, p. 70–82, 2003. ISSN 0015-198X.

KALAYCI, C. B.; ERTENLICE, O.; AKBAY, M. A. A comprehensive review of deterministic models and applications for mean-variance portfolio optimization. *Expert Systems With Applications*, 125, p. 345–368, 2019. ISSN 0957-4174.

KALAYCI, C. B.; POLAT, O.; AKBAY, M. A. An efficient hybrid metaheuristic algorithm for cardinality constrained portfolio optimization. *Swarm and Evolutionary Computation*, 54, 2020. ISSN 2210-6502.

KOLM, P. N.; TUETUENCUE, R.; FABOZZI, F. J. 60 Years of portfolio optimization: Practical challenges and current trends. *European Journal of Operational Research*, 234, n. 2, p. 356–371, 2014. ISSN 0377-2217.

KONNO, H.; WIJAYANAYKE, A. Minimal cost index tracking under nonlinear transaction costs and minimal transaction unit constraints. *International Journal of Theoretical and Applied Finance*, v. 04, n. 06, p. 939–957, 2001. Disponível em: <https://doi.org/10.1142/S0219024901001292>.

KONNO, H.; YAMAZAKI, H. Mean-absolute Deviation Portfolio Optimization Model and its Applications to Tokyo Stock-market. *Management Science*, 37, n. 5, p. 519–531, 1991. ISSN 0025-1909.

KUSIAK, S. Carrier Portfolios. *Journal of Portfolio Management*, 40, n. 1, p. 61+, 2013. ISSN 0095-4918.

LEJEUNE, M. A.; SAMATLI-PAC, G. Construction of Risk-Averse Enhanced Index Funds. *Informs Journal on Computing*, 25, n. 4, p. 701–719, 2013. ISSN 1091-9856.

LI, Q.; BAO, L.; ZHANG, Q. L. Multi-scale tracking dynamics and optimal index replication. *Applied Economics Letters*, 21, n. 4, p. 252–256, 2014. ISSN 1350-4851.

LIAGKOURAS, K. A new three-dimensional encoding multiobjective evolutionary algorithm with application to the portfolio optimization problem. *Knowledge-based Systems*, 163, p. 186–203, 2019. ISSN 0950-7051.

LIAGKOURAS, K.; METAXIOTIS, K. Examining the effect of different configuration issues of the multiobjective evolutionary algorithms on the efficient frontier formulation for the constrained portfolio optimization problem. *Journal Of The Operational Research Society*, 69, n. 3, p. 416–438, 2018. ISSN 0160-5682.

LIAGKOURAS, K.; METAXIOTIS, K. Multi-period mean-variance fuzzy portfolio optimization model with transaction costs. *Engineering Applications of Artificial Intelligence*, 67, p. 260–269, 2018. ISSN 0952-1976.

LIAGKOURAS, K.; METAXIOTIS, K. Improving the performance of evolutionary algorithms: a new approach utilizing information from the evolutionary process and its application to the fuzzy portfolio optimization problem. *Annals of Operations Research*, 272, n. 1-2, SI, p. 119–137, 2019. ISSN 0254-5330.

LING, A.; SUN, J.; YANG, X. Robust tracking error portfolio selection with worst-case downside risk measures. *Journal of Economic Dynamics & Control*, 39, p. 178–207, 2014. ISSN 0165-1889.

LIU, Y. J.; ZHANG, W.-G.; GUPTA, P. International asset allocation optimization with fuzzy return. *Knowledge-based Systems*, 139, p. 189–199, 2018. ISSN 0950-7051.

LIU, Y.-J.; ZHANG, W.-G.; ZHAO, X.-J. Fuzzy multi-period portfolio selection model with discounted transaction costs. *Soft Computing*, 22, n. 1, p. 177–193, 2018. ISSN 1432-7643.

LOTFI, S.; ZENIOS, S. A. Robust VaR and CVaR optimization under joint ambiguity in distributions, means, and covariances. *European Journal of Operational Research*, 269, n. 2, p. 556–576, 2018. ISSN 0377-2217.

LóPEZ-IBáñEZ, M.; DUBOIS-LACOSTE, J.; Pérez Cáceres, L.; BIRATTARI, M.; STüTZLE, T. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, v. 3, p. 43 – 58, 2016. ISSN 2214-7160. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2214716015300270>.

MARKOWITZ, H. Portfolio selection. *The Journal of Finance*, v. 7, n. 1, p. 77–91, 1952.

MARKOWITZ, H. *Portfolio Selection - Efficient Diversification of Investment*. New York: Wiley, 1959.

MARTI, R.; PARDALOS, P. M.; RESENDE, M. G. C. *Handbook of heuristics*. Cham: Springer, 2018.

MEGHWANI, S. S.; THAKUR, M. Multi-objective heuristic algorithms for practical portfolio optimization and rebalancing with transaction cost. *Applied Soft Computing*, 67, p. 865–894, 2018. ISSN 1568-4946.

MEZALI, H.; BEASLEY, J. E. Quantile regression for index tracking and enhanced indexation. *Journal Of The Operational Research Society*, 64, n. 11, p. 1676–1692, 2013. ISSN 0160-5682.

MEZALI, H.; BEASLEY, J. E. Index tracking with fixed and variable transaction costs. *Optimization Letters*, 8, n. 1, p. 61–80, 2014. ISSN 1862-4472.

MUTUNGE, P.; HAUGLAND, D. Minimizing the tracking error of cardinality constrained portfolios. *Computers & Operations Research*, 90, p. 33–41, 2018. ISSN 0305-0548.

NAKAYAMA, J.; YOKOUCHI, D. Applying Time Series Decomposition to Construct Index-Tracking Portfolio. *Asia-Pacific Financial Markets*, 25, n. 4, p. 341–352, 2018. ISSN 1387-2834.

NI, H.; WANG, Y. Stock index tracking by Pareto efficient genetic algorithm. *Applied Soft Computing*, 13, n. 12, p. 4519–4535, 2013. ISSN 1568-4946.

OTKEN, C. N.; ORGAN, Z. B.; YILDIRIM, E. C.; CAMLICA, M.; CANTURK, V. S.; DUMAN, E.; TEKSAN, Z. M.; KAYIS, E. An extension to the classical mean-variance portfolio optimization model. *Engineering Economist*, 64, n. 3, SI, p. 310–321, 2019. ISSN 0013-791X.

OUYANG, H.; ZHANG, X.; YAN, H. Index tracking based on deep neural network. *Cognitive Systems Research*, 57, p. 107–114, 2019. ISSN 1389-0417.

PAPADIMITRIOU, C. H.; STEIGLITZ, K. *Combinatorial Optimization*. Mineola, New York: Dover Publications, 1998.

PAPANTONIS, I. Cointegration-based trading: evidence on index tracking & market-neutral strategies. *Managerial Finance*, 42, n. 5, p. 449–471, 2016. ISSN 0307-4358.

RANGEL-GONZALEZ, J. A.; FRAIRE, H.; SOLIS, J. F.; CRUZ-REYES, L.; GOMEZ-SANTILLAN, C.; RANGEL-VALDEZ, N.; CARPIO-VALADEZ, J. M. Fuzzy Multi-objective Particle Swarm Optimization Solving the Three-Objective Portfolio Optimization Problem. *International Journal of Fuzzy Systems*, 2020. ISSN 1562-2479.

RESENDE, M. G. C.; RIBEIRO, C. C. Restart strategies for GRASP with path-relinking heuristics. *Optimization Letters*, Springer Science and Business Media LLC, v. 5, n. 3, p. 467–478, abr. 2011. Disponível em: <https://doi.org/10.1007/s11590-011-0322-2>.

RESENDE, M. G. C.; RIBEIRO, C. C. *Optimization by GRASP: Greedy Randomized Adaptive Search Procedures*. New York: Springer, 2016.

ROCKAFELLAR, R. T.; URYASEV, S. Optimization of conditional value-at-risk. *Journal of Risk*, v. 2, p. 21–41, 2000.

ROLL, R. A Mean-variance Analysis of Tracking Error - Minimizing the Volatility of Tracking Error Will Not Produce a More Efficient Managed Portfolio. *Journal of Portfolio Management*, 18, n. 4, p. 13–22, 1992. ISSN 0095-4918.

ROY, A. D. Safety First and the Holding of Assets. *ECONOMETRICA*, 20, n. 3, p. A431–449, 1952. ISSN 0012-9682.

RUDOLF, M.; WOLTER, H.; ZIMMERMANN, H. A linear model for tracking error minimization. *Journal of Banking & Finance*.

RUIZ-TORRUBIANO, R.; SUAREZ, A. A hybrid optimization approach to index tracking. *Annals of Operations Research*, 166, n. 1, p. 57–71, 2009. ISSN 0254-5330.

SALEHPOOR, I. B.; MOLLA-ALIZADEH-ZAVARDEHI, S. A constrained portfolio selection model at considering risk-adjusted measure by using hybrid meta-heuristic algorithms. *Applied Soft Computing*, 75, p. 233–253, 2019. ISSN 1568-4946.

SANT'ANA, L. R.; CALDEIRA, J. F.; FILOMENA, T. P. Lasso-based index tracking and statistical arbitrage long-short strategies. *North American Journal of Economics and Finance*, 51, 2020. ISSN 1062-9408.

SANT'ANNA, L. R.; FILOMENA, T. P.; CALDEIRA, J. F. Index tracking and enhanced indexing using cointegration and correlation with endogenous portfolio selection. *Quarterly Review of Economics and Finance*, 65, p. 146–157, 2017. ISSN 1062-9769.

SANT'ANNA, L. R.; FILOMENA, T. P.; CALDEIRA, J. F.; BORENSTEIN, D. Investigating the use of statistical process control charts for index tracking portfolios. *Journal Of The Operational Research Society*, 70, n. 10, SI, p. 1622–1638, 2019. ISSN 0160-5682.

SANT'ANNA, L. R.; FILOMENA, T. P.; GUEDES, P. C.; BORENSTEIN, D. Index tracking with controlled number of assets using a hybrid heuristic combining genetic algorithm and non-linear programming. *Annals of Operations Research*, 258, n. 2, p. 849–867, 2017. ISSN 0254-5330.

SCOZZARI, A.; TARDELLA, F.; PATERLINI, S.; KRINK, T. Exact and heuristic approaches for the index tracking problem with UCITS constraints. *Annals of Operations Research*, 205, n. 1, p. 235–250, 2013. ISSN 0254-5330.

SHARMA, A.; UTZ, S.; MEHRA, A. Omega-CVaR portfolio optimization and its worst case analysis. *OR Spectrum*, 39, n. 2, p. 505–539, 2017. ISSN 0171-6468.

SILVA, Y. L. T. V.; HERTHEL, A. B.; SUBRAMANIAN, A. A multi-objective evolutionary algorithm for a class of mean-variance portfolio selection problems. *Expert Systems With Applications*, 133, p. 225–241, 2019. ISSN 0957-4174.

SOLARES, E.; COELLO, C. A. C.; FERNANDEZ, E.; NAVARRO, J. Handling uncertainty through confidence intervals in portfolio optimization. *Swarm and Evolutionary Computation*, 44, p. 774–787, 2019. ISSN 2210-6502.

S&P. *Standard & Poor's Global Ratings*. 2019. Disponível em: <https://www.standardandpoors.com/pt_LA/web/guest/home>.

STOYAN, S. J.; KWON, R. H. A two-stage stochastic mixed-integer programming approach to the index tracking problem. *Optimization and Engineering*, 11, n. 2, p. 247–275, 2010. ISSN 1389-4420.

STRUB, O.; BAUMANN, P. Optimal construction and rebalancing of index-tracking portfolios. *European Journal of Operational Research*, 264, n. 1, p. 370–387, 2018. ISSN 0377-2217.

STRUB, O.; TRAUTMANN, N. A two-stage approach to the UCITS-constrained index-tracking problem. *Computers & Operations Research*, 103, p. 167–183, 2019. ISSN 0305-0548.

TAS, E.; TURKAN, A. H. Regularized Index-tracking Optimal Portfolio Selection. *Economic Computation and Economic Cybernetics Studies and Research*, 52, n. 3, p. 135–146, 2018. ISSN 0424-267X.

THEOBALD, M. F.; YALLUP, P. J. Liability-driven investment: multiple liabilities and the question of the number of moments. *European Journal of Finance*, 16, n. 5, p. 413–435, 2010. ISSN 1351-847X.

VALLE, C. A.; MEADE, N.; BEASLEY, J. E. An optimisation approach to constructing an exchange-traded fund. *Optimization Letters*, 9, n. 4, p. 635–661, 2015. ISSN 1862-4472.

WANG, M.; XU, C.; XU, F.; XUE, H. A mixed 0-1 LP for index tracking problem with CVaR risk constraints. *Annals of Operations Research*, 196, n. 1, p. 591–609, 2012. ISSN 0254-5330.

WANG, M.; XU, F.; DAI, Y.-H. An index tracking model with stratified sampling and optimal allocation. *Applied Stochastic Models in Business and Industry*, 34, n. 2, p. 144–157, 2018. ISSN 1524-1904.

WU, D.; KWON, R. H.; COSTA, G. A constrained cluster-based approach for tracking the S&P 500 index. *International Journal of Production Economics*, 193, p. 222–243, 2017. ISSN 0925-5273.

WU, D.; WU, D. D. An enhanced decision support approach for learning and tracking derivative index. *Omega- International Journal of Management Science*, 88, p. 63–76, 2019. ISSN 0305-0483.

WU, L.; YANG, Y. Nonnegative Elastic Net and application in index tracking. *Applied Mathematics and Computation*, 227, p. 541–552, 2014. ISSN 0096-3003.

WU, L.; YANG, Y.; LIU, H. Nonnegative-lasso and application in index tracking. *Computational Statistics & Data Analysis*, 70, p. 116–126, 2014. ISSN 0167-9473.

WU, L.-C.; TSAI, I.-C. Three fuzzy goal programming models for index portfolios. *Journal Of The Operational Research Society*, 65, n. 8, p. 1155–1169, 2014. ISSN 0160-5682.

XU, F.; LU, Z.; XU, Z. An efficient optimization approach for a cardinality-constrained index tracking problem. *Optimization Methods & Software*, 31, n. 2, p. 258–271, 2016. ISSN 1055-6788.

YANG, Y.; WU, L. Nonnegative adaptive lasso for ultra-high dimensional regression models and a two-stage method applied in financial modeling. *Journal of Statistical Planning and Inference*, 174, p. 52–67, 2016. ISSN 0378-3758.

YOSHIDA, Y. Portfolio optimization in fuzzy asset management with coherent risk measures derived from risk averse utility. *Neural Computing & Application*, 32, n. 15, p. 10847–10857, 2020. ISSN 0941-0643.

ZHAO, K.; LIAN, H. The Expectation-Maximization approach for Bayesian quantile regression. *Computational Statistics & Data Analysis*, 96, p. 1–11, 2016. ISSN 0167-9473.

ZHOU, X.; WANG, H.; PENG, W.; DING, B.; WANG, R. Solving multi-scenario cardinality constrained optimization problems via multi-objective evolutionary algorithms. *Science China-Information Sciences*, 62, n. 9, 2019. ISSN 1674-733X.