Invited review

Biased random-key genetic algorithms: A review

Mariana A. Londe [a], Luciana S. Pessoa [a], Carlos E. Andrade [b], Mauricio G.C. Resende [c,*]

[a] Department of Industrial Engineering, PUC-Rio, Rua Marquês de São Vicente, 225, Gávea 22453-900 Rio de Janeiro, RJ, Brazil
[b] AT&T Labs Research, 200 South Laurel Avenue, Middletown, NJ 07748, USA
[c] Industrial & Systems Engineering, University of Washington, 3900 E Stevens Way NE, Seattle, WA 98195, USA

ARTICLE INFO

ABSTRACT

This paper is a comprehensive literature review of Biased Random-Key Genetic Algorithms (BRKGA). BRKGA is a metaheuristic that employs random-key-based chromosomes with biased, uniform, and elitist mating strategies in a genetic algorithm framework. The review encompasses over 150 papers with a wide range of applications, including classical combinatorial optimization problems, real-world industrial use cases, and non-orthodox applications such as neural network hyperparameter tuning in machine learning. Scheduling is by far the most prevalent application area in this review, followed by network design and location problems. The most frequent hybridization method employed is local search, and new features aim to increase population diversity. We also detail challenges and future directions for this method. Overall, this survey provides a comprehensive overview of the BRKGA metaheuristic and its applications and highlights important areas for future research.

1. Introduction

Genetic algorithms, first introduced by Holland (1975), have proven to be effective for solving challenging optimization problems, ranging from discrete and combinatorial problems to non-linear and derivative-free optimization problems. This class of methods is a population-based metaheuristic that uses the principles of natural selection and survival of the fittest individuals, providing a powerful tool for searching large solution spaces. As a result, genetic algorithms have been applied in various domains due to their efficiency and versatility.

Biased Random-Key Genetic Algorithms (BRKGA) have become a popular variant of genetic algorithms due to their successful application in several optimization problems, from classical hard combinatorial optimization problems (Resende, Toso, Gonçalves, & Silva, 2012; Rochman, Prasetyo, & Nugroho, 2017) to real-world problems, such as packing (Gonçalves & Resende, 2011b), combinatorial auctions (Andrade, Toso, Resende, & Miyazawa, 2015), scheduling (Andrade, Byers, et al., 2017, 2019; Andrade, Silva, & Pessoa, 2019; Cunha, Pessoa, Vellasco, Tanscheit, & Pacheco, 2018; Pessoa & Andrade, 2018), vehicle routing (Andrade, Miyazawa, & Resende, 2013; Lopes, Andrade, Queiroz, Resende, & Miyazawa, 2016), clustering (Andrade, Resende, Karloff, & Miyazawa, 2014), complex network design (Andrade, Pessoa, & Stawiarski, 2022; Andrade, Resende, et al., 2015), placement of virtual machines in data centers (Stefanello, Aggarwal, Buriol, & Resende, 2019), and machine learning (Caserta & Reiners, 2016; Paliwal et al.,

2020), to cite only a few. BRKGA was formally introduced by Gonçalves and Resende (2011a), although its elements have appeared before in Beirão (1997), Buriol, Resende, Ribeiro, and Thorup (2005), Ericsson, Resende, and Pardalos (2002), Gonçalves and Almeida (2002), Gonçalves and Beirão (1999).

The first prominent feature of BRKGA is its agnosticism in solving the problem. In most (meta) heuristic algorithms, the optimization mechanism is intrinsically tied to the problem structure. Thus, while we apply the general framework, we still need to develop or code details of the framework. In a BRKGA, such constant rework is avoided using a standard representation of the solutions. The population in a BRKGA lives in a half-open unit hypercube of dimension n, and each solution or individual is represented by a point in $(0, 1]^n$, called a *chromosome*. Such representation, proposed in Bean (1994), makes the method independent of the problem it solves and, therefore, allows for code reuse.

The second outstanding feature of a BRKGA is its fast convergence to high-quality solutions. Such achievement is due to the double elitism mechanism embedded in the evolutionary process of BRKGA, characterized by the handing out of the best solutions between generations and the biased mating process towards the best individual. The backdrop of double elitism is the fast convergence to a local optimum. BRKGA deals with such issues by introducing mutants (random solutions) and

---

with other operators. Nevertheless, BRKGA can deliver high-quality solutions in time-critical applications in a reasonably short time, making it suitable for several industrial applications.

Due to such features, there has been an increasing number of applications of BRKGA in the last years. In the same time frame, three studies aimed to organize the literature of BRKGA (Gonçalves & Resende, 2011a; Prasetyo, Fauza, Amer, & Lee, 2015; Resende, 2012) on specific applications, but their attempts are limited in the sample of articles studied. Currently, there are no studies that focus on consolidating the literature of BRKGA. A literature review can be helpful in this, as this type of scholarly research focuses on presenting an overview of performance, knowledge, and significant themes on a field, charting its progress, gaps, and emerging areas, and suggesting opportunities and solutions for future works (Lim, Kumar, & Ali, 2022).

In this paper, we present a detailed survey of the existing literature regarding BRKGA to discern the most studied problems, principal modifications to the framework, and the most frequent hybridization methods explored. With this, we aim to observe the current state of the research with BRKGA, to detail the metaheuristics strengths and weaknesses regarding different problem variants, and to present an extensive overview of the applications of BRKGA. We expect readers to have a concise but broad view of BRKGA applications helping them in future endeavors.

Snyder (2019) details four phases to conduct a literature review: (i) Designing the review; (ii) Conducting the review; (iii) Analysis; and (iv) Writing the review. The first phase, designing the review, comprises the definition of the scope of the review, the characterization of the potential audience, and the specification of the following research questions (RQ):

RQ1: Which are the areas of application of BRKGA and how did they evolve?

RQ2: Which are the types of hybridization involving BRKGA and how did they evolve?

RQ3: How was the BRKGA framework modified since its inception?

RQ4: What are the opportunities for the growth of the BRKGA framework?

The second phase demands the definition and application of the search method. We followed the seven-step approach listed in Thomé, Scavarda, and Scavarda (2016). Those stages are the database selection, followed by the definition of keywords, review of abstracts, definition of criteria for inclusion and exclusion of works, full-text review, and backward and forward search in the selected works.

We selected the Scopus database (Baas, Schotten, Plume, Côté, & Karimi, 2020; Elsevier, 2022) for this study. Scopus is one of the largest curated abstract and citation databases, with over 39,100 journal titles from 5,000 publishers worldwide. It is less restrictive in the selection of titles than its counterpart Web of Science, and has a bias towards the fields of technology and engineering (Singh, Singh, Karmakar, Leta, & Mayr, 2021), thus being adequate for the purpose of this review.

The definition of keywords should be broad so as not to restrict the number of studies and specific enough to bring only works related to the topic (Cooper, 2015). The keywords used were "BRKGA" or "Biased random key genetic algorithm", resulting in 259 works. The limitation to peer-reviewed articles in the English language resulted in 144 works. Then, backward and forward snowball searches were employed. The former is defined as the review of the literature cited in the articles yielded from the keyword search, while the latter is the review of additional articles that cite those retrieved (Webster & Watson, 2002). Both snowball searches specifically looked for articles in which random keys were used with biased mating in genetic algorithms to find possible older titles from before the formal definition of BRKGA in 2011. After applying the keywords to the new registers alongside the

exclusion criteria and eliminating all works with non-related abstracts and duplicated papers, we ended up with 162 articles published before October 1st, 2023.

The third phase of the review, analysis, was fulfilled by obtaining data regarding the used BRKGA methods, results, and hybrids. With this, we categorized the different studies regarding problem characteristics and methodologies and performed a qualitative content analysis with an inductive approach (Seuring & Gold, 2012).

Finally, the fourth phase is the presentation of the results obtained in the previous steps and detailed in the remainder of this paper, which is structured as follows. Section 2 presents the fundamentals of the BRKGA framework in a detailed manner. In Section 3, we present a comprehensive description of BRKGA applications. Section 4 highlights the main hybridizations done with the framework. In Section 5, we detail features that have been developed since the inception of this algorithm. Section 6 comments about under-performing issues of BRKGA and their possible causes. Possible directions for future research are presented in Section 7. Finally, in Section 8, we make concluding remarks.

2. Fundamentals of the BRKGA

This section presents the basics of Biased Random Key Genetic Algorithms, as described in Gonçalves and Resende (2011a). Biased Random Key Genetic Algorithms (BRKGA) fall under the category of genetic algorithms, which are a class of search and optimization algorithms inspired by the process of natural selection. Introduced in the 1970s by Holland (1975), the fundamental concept of Genetic Algorithms (GA) involves treating a problem's solution as an individual within a population. These algorithms operate with a population of potential solutions, where each solution is represented as a chromosome or a string of genes. The population itself comprises a collection of these individuals.

GAs aim to replicate the process of evolution found in nature. Over time, the population evolves by implementing the Darwinian principle of survival of the fittest. Weaker individuals are unable to pass on their characteristics and gradually diminish within the population, while stronger individuals reproduce and transmit their characteristics. The algorithm proceeds through a series of generations, and at the conclusion of these generations, the individual with the best fitness value, often representing the best solution, is the output of the algorithm.

Individuals from one generation are combined to produce offspring that make up the next generation. The concept of passing on genetic information is generally implemented by selecting individuals to form pairs based on a probability proportional to their quality. This means that stronger individuals are more likely to be chosen for reproduction. The process of combining two solutions, the crossover, is the mechanism that intensifies the search, focusing on propagating the best traits of parents to their offspring. Once the two parents are selected, they are combined in some manner to generate offspring. An external process, akin to the mutations that occur in nature, introduces changes to the genetic information of the offspring. When a mutation is beneficial, it can lead to an improvement in the quality of the offspring.

In any search process, two essential components are at play: intensification and diversification. The crossover process contributes primarily to intensification, enhancing the exploration of promising areas in the solution space. On the other hand, diversification, responsible for exploring new regions of the solution space, is chiefly influenced by the process of mutation. The delicate balance between these phases is pivotal to the success of genetic algorithms in effectively optimizing complex problems.

This combination of selection, crossover, and mutation mimics the process of natural selection and drives the evolution of the population towards optimal or near-optimal solutions for the given problem.
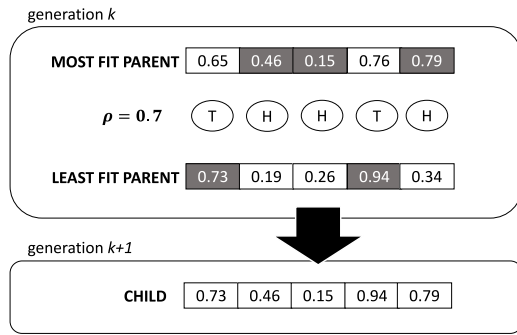
Fig. 1. Mating process in BRKGA.

Fig. 2. Evolutionary process between consecutive generations.

2.1. Genetic algorithms and random-keys

A random key is a randomly generated real number in the continuous interval $[0, 1)$. Specifically, a vector X composed of random-keys, often referred to simply as random keys, consists of an array containing n such random keys.

Optimization problem solutions can be effectively encoded using these vectors of random keys, as proposed by Bean (1994). Once a solution is encoded, a decoding process becomes necessary. The decoder is a deterministic algorithm that takes, as input, a vector of random keys and outputs a solution to the optimization problem and its cost.

An example of a simple decoder applied to the Traveling Salesman Problem (TSP) follows. In this problem, there is a set of n cities that must be visited exactly once, starting and ending in the same city. A solution to this problem is essentially a sequence of cities. Assume a vector of random keys denoted as $X = [0.234, 0.876, 0.321, 0.693, 0.087]$ is given. Each position in the vector corresponds to an index representing a city. To extract a solution for the TSP from this random key vector, one can simply sort this vector in increasing order and retrieve the sequence of indices associated with the sorted sequence. In this example, the sorted sequence yields the solution: 5-1-3-4-2, indicating the order in which the cities are visited by the traveling salesman. By applying the objective function to the decoded sequence, the decoder can determine the overall cost or fitness of the solution.

A critical aspect of a genetic algorithm is the combination of two solutions to generate offspring. In the basic BRKGA, as proposed by Gonçalves and Resende (2011a), each parent is a vector of random keys, and the mating process is performed using parameterized uniform crossover (Spears & De Jong, 1991). For each gene (each position in the random key vector), a biased coin is flipped to determine which parent will contribute its allele (key or gene value) to the offspring. The highest probability is assigned to the most fit parent, prioritizing the selection of genetic material from the parent that represents a higher-quality solution. This bias towards the fitter parent contributes to the algorithm's tendency to preserve and propagate the characteristics of more promising solutions throughout the evolution process. Fig. 1 illustrates this crossover process where two parents with different fitnesses are combined. The most fit of the two will have a higher probability of passing on its genes to the child. A biased coin is flipped to determine which gene is passed on. An outcome of heads (H) corresponds to the most fit parents while tails (T), to the least fit.

2.2. The evolutionary process in the BRKGA

The first generation, also known as the starting or initial population, consists of p individual chromosomes, with each chromosome being composed of n genes. Each gene is assigned a value (allele) that is generated uniformly at random within the interval $[0, 1)$. To transform the random key sequence into a solution and simultaneously evaluate the solution's cost or fit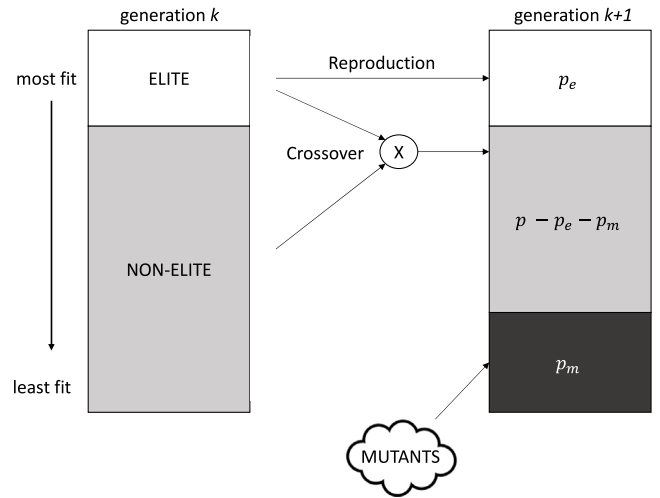ness, each chromosome is processed by the decoder. Usually, decoded solutions are feasible, but infeasibility can be dealt with by penalties.

In generation k, the population is partitioned into two subsets: the elite and the non-elite. The elite subset comprises solutions with superior fitness scores, while the non-elite are the remaining solutions found in the population. Importantly, the size of the elite subset, determined by a parameter p_e is intentionally smaller than that of the non-elite subset, prioritizing the preservation and evolution of high-performing solutions throughout the generations of the genetic algorithm.

To obtain generation $k + 1$, as shown in Fig. 2, three operations are performed on the current population:

Reproduction simply copies the p_e chromosomes (random-key vectors) from the elite set of population k to the successor population. This illustrates the elitist characteristic of the original BRKGA.

Creation of *mutants* inserts p_m new individuals in the successor population. These individual chromosomes are created as those of the initial population.

Crossover performs the mating process as described in Section 2.1 to generate the remaining $p - p_e - p_m$ individuals in population $k + 1$. To form the couple, a parent is randomly chosen from the elite set, while the other parent comes from the non-elite partition. In some papers, however, the second parent is selected from the entire population.

Once a new population is formed, its individuals are decoded and their costs are computed. Finally, the new population can be partitioned into elite and non-elite sets, and the evolutionary process restarts.

3. Main problems studied with BRKGA

To explore the areas of application of BRKGA and how they evolved, as detailed in the first RQ, a thorough reading of 156 papers was performed. Throughout the literature review it was possible to identify ten categories of applications with at least four papers each, and 24 applications studied in up to three papers, for a total of 34 different applications. For illustration purposes, Fig. A.1 in the supplementary material shows the number of studies over the years per type of application domain, while Fig. A.2 presents the total amount of papers in each category. Note that since we only consider indexed peer-reviewed papers in English from the Scopus database, the actual number of papers is probably greater than what we report. For example, we do not consider papers appearing in conferences nor in the arXiv repository.

In the following subsections, we present the applications of BRKGA, detailing the core ideas of these algorithms and how they evolved over time. The papers are presented in chronological order by micro-problems studied.

3.1. Scheduling

BRKGA has been extensively used for scheduling problems. In fact, almost a third of all BRKGA applications focus on this category of problems, which is by far the most studied problem with this metaheuristic. In 2002, Gonçalves and Almeida (2002) proposed a hybrid BRKGA with local search for the *simple assembly line balancing problem*, which managed to outperform several algorithms from the literature on benchmark instances. The study from Moreira, Ritt, Costa, and Chaves (2012) introduces the related problem *assembly line worker assignment and balancing problem*. The proposed BRKGA+LS was compared favorably with a constructive strategy – the same one used in the decoder – and several algorithms from the literature. Later, Araújo, Costa, and Miralles (2015) study the *parallel assembly line worker assignment and balancing problem*, an extension of the previous one. Their BRKGA outperformed a mathematical model but performed worse than a tabu search (TS) strategy.

Several papers study variants of the *single batch processing machine problem* (Kashan, Karimi, & Jolai, 2006; Li & Zhang, 2018; Malve & Uzsoy, 2007; Mönch & Roob, 2018; Wang & Uzsoy, 2002). In Kong, Liu, Pei, Cheng, and Pardalos (2020), the *parallel batch scheduling with deterioration and learning effects on parallel machines problem* is introduced. The authors show that a hybrid BRKGA with a differential evolution algorithm has a better performance than both a particle swarm optimization algorithm and a traditional BRKGA. The same authors, in Kong, Pei, et al. (2020), focus on the *integrated steel production and batch delivery scheduling with uncertain rolling times and deterioration effect problem*. In this case, however, while a BRKGA outperformed dynamic programming and differential evolution approaches, it is outperformed itself by a VNS. Zhang, Shan, and Zeng (2022) studies the *parallel batch processing machine scheduling problem under 2D bin-packing constraints*, which are added to increase applicability. An orthogonal BRKGA approach is developed, in which initial solutions are obtained with orthogonal design. This approach is shown to improve convergence and performance, with robuster and stabler results in comparison with several alternatives. Yu, Qian, Zhang, and Kong (2023) investigates a *two-echelon steel supply chain scheduling problem with parallel batch processing and deterioration effect*. A hybrid BRKGA with flower pollination operators is shown to be effective and efficient in comparison with alternatives such as PSO, FPA, and classic BRKGA.

The *job shop scheduling problem* is another often explored with BRKGA (Gonçalves, Magalhães Mendes, & Resende, 2005; Gonçalves & Resende, 2014; Won, Bae, & Sutrisnowati, 2018). Homayouni, Fontes, and Gonçalves (2020) studies the *flexible job shop scheduling problem*. This work also used a population re-start operator, which acted similarly to a previous reset strategy in the BRKGA literature. This BRKGA outperformed a MILP and a number of approaches from the literature on benchmark instances. Fontes and Homayouni (2023) explores a *bi-objective energy-efficient job shop scheduling problem with transport resources*. The authors introduce a novel multi-objective approach to BRKGA, in which $\Pi + \Omega$ populations are evolved. Each of the Ω populations focuses on one of the objectives, while Π populations consider all objectives and individual contribution to diversity. This alternative is shown to find Pareto Fronts close to the true Pareto Front, and outperforms NSGA-II in solution quality.

Meanwhile, Valente, Gonçalves, and Alves (2006) focuses on the *single-machine scheduling problem with early and tardy penalties*. Their hybrid BRKGA+LS with a warm-start strategy outperformed many approaches from the literature, alongside variants without the warm-start and local search. Valente and Gonçalves (2009) studies a variant of the previous problem, the *single-machine scheduling problem with linear early and quadratic tardy penalties*. The proposed BRKGA with warm-start and local search had a better performance than several algorithms from the literature and variants without the warm-start and/or local search.

Four articles tackle variants of the *project scheduling problem* (Gonçalves, Mendes, & Resende, 2008; Gonçalves, Resende, & Mendes, 2011; Mendes, Gonçalves, & Resende, 2009). In the most recent, Almeida, Correia, and Saldanha-da Gama (2018) study the *resource constrained project scheduling with flexible resources problem*. Their decoder had a variable strategy: a gene indicates which of two algorithms is to be used to obtain the fitness value of a chromosome. This BRKGA outperformed constructive heuristics, and the BRKGA with variable decoding strategy was proved to be better than the ones with fixed strategies.

Gonçalves and Sousa (2011) is the only study that applies BRKGA to the *economic lot scheduling problem*. Their BRKGA was hybridized with an LP model and managed to outperform several approaches from the literature on randomly generated instances. The next article studies the *single-round divisible load scheduling problem*. Brandão, Noronha, Resende, and Ribeiro (2015) propose a BRKGA that has a better performance than a multi-start algorithm combined with the decoding heuristic and some strategies used in the literature. Later, the same authors study the *multi-round divisible load scheduling problem* in Brandão, Noronha, Resende, and Ribeiro (2017). Again in comparison with a multi-start algorithm and literature strategies, the proposed BRKGA was deemed the best approach.

The next two papers study the *multi-user observation scheduling problem*. The first, authored by Tangpattanakul, Jozefowiez, and Lopez (2015a), compares BRKGA multi-objective variants, while the second, also by Tangpattanakul, Jozefowiez, and Lopez (2015b), uses the hybrid decoder formulation suggested in the first. The latter study shows that an indicator-based multi-objective local search performed better than BRKGA for the studied problem. Damm, Resende, and Ronconi (2016) focuses on the *field technician scheduling problem*. Their BRKGA uses a novel elite diversification strategy that only lets individuals be copied in the reproduction step if they are significantly different from the other copied individuals. This was successful in preventing premature convergence and performed better than a MILP model and several constructive heuristics. Later, the work from Cabo, González-Velarde, Possani, and Ríos Solís (2018) explores the *bi-objective p-batch machine scheduling problem*. The BRKGA proposed considered the bi-objective nature of the problem in its decoding process and had competitive results with shorter computational time in comparison with two MILP models.

Two papers deal with the *flowshop scheduling problem* (Andrade, Byers, et al., 2019; Pessoa & Andrade, 2018). A generalization, the *workforce allocation and two-stage flexible flow shop problem*, was studied in Bolsi, Lima, Alves Queiroz, and Iori (2022). BRKGA had the best performance on this lexicographic multi-objective problem in comparison with several heuristics, such as constructive methods, a random multi-start algorithm, a VNS, and a constraint programming model. He, Guijt, Weerdt, Xing, and Yorke-Smith (2019) uses a hybrid BRKGA with an adaptive large neighborhood search algorithm to solve the *order acceptance and scheduling problem*. Their approach outperformed many algorithms and MIP models from the literature on benchmark instances. The authors also introduced new instances for the problem. The work from Andrade, Byers, et al. (2019) introduces the *time- and machine-dependent scheduling problem*. In this work, both real-life-based and synthetic instances were used to prove the effectiveness of the BRKGA. In fact, the BRKGA had a better performance than several other algorithms, including ILS, simulated annealing, tabu search, genetic algorithm, and a MIP model.

Soares and Carvalho (2020) aims to solve the *identical parallel-machines problem with tooling constraints*. The authors use a hybrid BRKGA+VND+LS to solve benchmark instances and prove that their BRKGA performs better than both several approaches from the literature as well as a MIP model. Rocholl and Mönch (2021) studies the *parallel-machine multiple orders per job scheduling problem with a common due date*. The BRKGA was the best among the tested approaches, including a MILP and an ILS. The *resource-constrained parallel-machine scheduling problem with setup times* is explored in Soares and

Carvalho (2022). Their BRKGA is hybridized with a VND with four neighborhoods, is shown to obtain tighter upper bounds for the problem in comparison with state-of-the-art methods. Maecker, Shen, and Mönch (2023) explores the *unrelated parallel-machine scheduling problem with job-machine-dependent delivery times and eligibility constraints*. On randomly generated instances, the proposed BRKGA can outperforms several approaches from the literature, but needs higher computing time than a VNS to reach competitive solutions.

Abreu, Tavares-Neto, and Nagano (2021) focuses on the *open shop scheduling problem with routing by capacitated single vehicle*, in which a scheduling problem is combined with a routing problem. Their hybrid BRKGA+IG approach outperformed a MIP model, a greedy insertion algorithm, an IGS, and a pure BRKGA on benchmark instances. This combination of problems is also explored in Kummer, Araújo, Buriol, and Resende (2022) on the *home health care routing and scheduling problem*. The authors use BRKGA-MP-IPR (Andrade, Toso, Gonçalves, & Resende, 2021) and analyze the impact of each novel feature on solution quality and running times. The experiments indicate that BRKGA-MP-IPR with greedy cheapest-insertion constructive heuristic as decoder has better performance than state-of-the-art methods if longer running times are affordable. Queiroga, Pinheiro, Christ, Subramanian, and Pessoa (2021) presents a hybrid BRKGA with dynamic programming for the *single machine total weighted tardiness batch scheduling problem*. When compared with a traditional ILS and a hybrid ILS+DP, the hybrid BRKGA outperformed its fellow hybrid, but both had worse performances than the classical ILS. Silva-Soto and Ibarra-Rojas (2021) study the *bi-objective optimization approach for frequency setting and timetabling problem*. The authors compare single- and bi-objective BRKGA with a MILP model. The bi-objective BRKGA used a non-dominated sorting algorithm in its decoder and proved to be the most effective approach for this problem.

The *customer order scheduling problem with missing operations* was studied in Abreu, Athayde Prata, Gomes, Braga-Santos, and Nagano (2022). The authors use a parameter-free restart procedure, that is performed when the worst fitness equals the best fitness in the population. A best improvement local search was also introduced. Their BRKGA outperformed several approaches from the literature on generated instances, with the combination of BRKGA+LS+restart being the best overall. Xie, Sheng, Qiu, and Gui (2022) explores the *cloud workflow scheduling problem*. An adaptive decoder is proposed in this paper, where one of three decoding methods is chosen by a random probability. This probability changes during evolution, decreasing chance of the more complex decoders and increasing the chance of a simpler decoding strategy. A warm-start strategy is employed, where some initial chromosomes come from heuristics based on list scheduling and the others are based on a level heuristic. A local search is also customized. The experiments show that adaptive decoding BRKGA has a robust performance and outperforms several approaches from the literature. Fontes and Homayouni (2023) investigates *joint scheduling quay cranes and speed adjustable vehicles in container terminals*. The bi-objective BRKGA uses the same strategy as Fontes, Homayouni, and Fernandes (2023). Experiments shown that the proposed BRKGA obtains diverse and uniformly distributed solutions, which cover, almost always, the whole Pareto Frontier of the instance. Finally, Zhao, Liu, Wang, and Yue (2023) uses a floating point number encoding to solve the *multi-station multi-robot task allocation and sequential planning problem*. In the chromosome, the digit before "." indicates the work station, and the digits after, the robot allocated to the job. Due to problem characteristics, the authors also introduce three novel evolution operators. Double crossover uses two operations, in which the digits before and after "." are analyzed independently. The same is done with double mutation. The elite re-optimization operator diversifies the elite set if the minimum cost of the current and next generations are close by applying classic crossover and mutant operators. Experiments show that the proposed method outperforms a greedy insertion algorithm and a hybrid BRKGA+SA from the literature.

Considering the many studies in this category, it is difficult to identify common trends for scheduling problems. There is approximately the same amount of studies that either use benchmark or randomly generated instances, with the latter being a little more popular, and the slight majority of papers use permutation-based decoders. However, one trend common to most studies is the use of hybridization, generally with local search algorithms, whose positive impact on performance makes BRKGA the best approach to many of the studied problems.

3.2. Network configuration

Several papers focus on the *weight setting problem in OSPF routing*. The first, Ericsson et al. (2002), introduces a BRKGA with warm-start that performed better than several state-of-art algorithms. This BRKGA was later extended by Buriol et al. (2005), who added a local search operator to the previous framework. This addition outperformed both the previous BRKGA and the pure local search. Buriol, Resende, and Thorup (2007) changes the fitness function of their previously used BRKGA and observes which fitness calculation leads to better solution quality. Finally, Reis, Ritt, Buriol, and Resende (2011) extends the BRKGA of Ericsson et al. (2002) and Buriol et al. (2005) for *DEFT routing*. This BRKGA was outperformed by a two-stage approach.

Fontes and Gonçalves (2007) studies the *single source uncapacitated concave minimum cost network flow problem*. Their hybrid BRKGA with local search managed to outperform a pure BRKGA on benchmark instances. A related problem, the *hop-constrained minimum cost flow spanning tree problem*, was studied by Fontes and Gonçalves (2013). Their hybrid BRKGA+LS with a multi-population strategy found the best solutions for all instances and had better results in comparison with a mathematical model.

The *survivable IP/MPLS-over-WSON multilayer network optimization problem* was explored in two papers. Ruiz, Pedrola, Velasco, Careglio, Fernández-Palacios, and Junyent (2011) introduced a BRKGA and a mathematical model for this problem, while Pedrola, Ruiz, Velasco, Careglio, González De Dios, and Comellas (2013) compared a multi-population BRKGA with several GRASP approaches. The authors of the latter showed that a hybrid GRASP+PR approach was the most effective.

Noronha, Resende, and Ribeiro (2011) studies the *routing and wavelength assignment problem*. The proposed BRKGA had a better performance when compared with a state-of-the-art tabu search algorithm and other approaches from the literature. Later, Brandão, Noronha, and Ribeiro (2016) focused on the same problem. The authors used two different algorithms as decoders, and show that the resulting BRKGAs are more effective than many constructive algorithms in the literature. Finally, Pinto, Ribeiro, Rosseti, and Noronha (2020) compared a BRKGA with the best algorithms in the literature and a multi-start version of the decoder algorithm, in which the BRKGA was considered the most effective.

Velasco, Wright, Lord, and Junyent (2013) focuses on three strongly related problems: the *area partitioning problem*, the *IP/MPLS area network design problem*, and the *core network design problem*. The authors introduce three different BRKGAs, each focusing on one of those problems, and observe the quality of the given solutions in close-to-real instances. The focus of Pedrola, Careglio, Klinkowski, Velasco, Bergman, and Solé-Pareta (2013) is the *regenerator placement and dimensioning problem*. The authors hybridize a BRKGA with both VND and path relinking. This hybrid algorithm outperformed several approaches, including a MILP and an evolutionary multi-start GRASP+VND+PR.

Morán-Mirabal, González-Velarde, Resende, and Silva (2013) studies the *handover minimization problem*. The proposed BRKGA+LS hybrid showed better results in comparison with a MIP model but had worse results than an evolutionary GRASP with path relinking. Andrade, Resende, et al. (2015) tackles the *wireless backhaul network design problem*. In real-life inspired instances, the proposed BRKGA outperformed both a MIP model and a multi-start algorithm. Ruiz, Albareda-Sambola,

Fernández, and Resende (2015) focuses on the *capacitated minimum spanning tree problem.* The authors introduce different decoders, alongside a local search strategy, strategic oscillation to explore new areas of the solution space, and a neighborhood reduction strategy to ignore non-feasible moves on the tree. In comparison with other alternatives, the BRKGA had higher solution quality and lower computational effort, especially when combined with the local search algorithm.

Three articles focus on the *distribution network reconfiguration problem* (Cavalheiro, Vergílio, & Lyra, 2018; Faria Jr., Resende, & Ernst, 2017; Raposo, Rodrigues, & da Guia da Silva, 2020). In the most recent paper, Raposo et al. (2020) adapts the BRKGA for multi-objective optimization and compares it favorably with a NSGA-II on benchmark instances. Finally, the paper authored by Mikulski and Tomczewski (2021) uses a multi-objective BRKGA to *reduce transmission losses in power distribution networks.* This multi-objective approach found better results than a multi-objective particle swarm optimization (PSO) and a NSGA-II on four test cases.

For network configuration problems, the use of indicator-based decoders is of note, something done by more than half of the studies. Another characteristic in common is the presence of hybridization, with either local search, warm start, or multi-population strategies. The use of benchmarks is common, as it happens on almost three-fourths of the papers in this category.

3.3. Location

In this subsection, problems that consider the location of one or several facilities are considered. One such problem is the *tollbooth problem,* whose BRKGA was proposed in Buriol, Hirsch, Pardalos, Querido, Resende, and Ritt (2010) and later extended in Stefanello et al. (2017). Those studies show that the approach with local search performs significantly better in time and quality for benchmark instances, in comparison with exact models and a local search-free approach. Another such problem is the *regenerator location problem* studied in Duarte, Martí, Resende, and Silva (2014). Using benchmark instances, the authors show that BRKGA has a better performance than the ones in the literature, but a GRASP+LS hybrid outperforms it.

A version of the *quadratic assignment problem,* the *unequal area facility layout problem,* is studied by Gonçalves and Resende (2015). The proposed approach hybridizes BRKGA with linear programming constraints. The hybridized version outperformed the pure BRKGA and several other algorithms from the literature. In fact, both BRKGA versions improved the best-known solution of several benchmark instances. Lalla-Ruiz, Expósito-Izquierdo, Melián-Batista, and Moreno-Vega (2016) studies the *quadratic assignment problem.* Their BRKGA has an improvement phase, similar to a local search, and performs better than a migrating birds algorithm and a discrete differential evolutionary algorithm with a local search portrayed in literature. Similarly, Stefanello et al. (2019) studies both a generalization of the previous problem and its specific application. This paper also hybridizes a BRKGA with path-relinking and local search. This hybrid BRKGA outperforms mathematical models and a hybrid GRASP-PR-LS for smaller, randomly generated instances.

Both Lopes et al. (2016) and Pessoa, Santos, and Resende (2017) study *hub-location problems.* The approach in Lopes et al. (2016) uses a hybrid BRKGA with local search and warm-start, which performs poorly in comparison with multi-start VND and local search, and an integer programming model with valid inequalities. Meanwhile, Pessoa et al. (2017) compares several BRKGAs and proves this approach's effectiveness and efficiency. A *two-level hub location routing problem with directed tours* is studied in Freitas, Aloise, Da Costa Fontes, Santos, and Da Silva Menezes (2023) with a modified decoding process. Their BRKGA outperformed a MIP and the state-of-the-art variable neighborhood decomposition search and improved the objective value for many of the benchmarks from the literature.

The study of Biajoli, Chaves, and Lorena (2019) focuses on the *two-stage capacitated facility location problem.* Their BRKGA+LS algorithm outperforms the genetic algorithm and clustering search presented in the literature on known benchmark instances, and it is extended to a *multi-product* version in Mauri et al. (2021). This extended algorithm also manages to outperform a clustering search algorithm and several mathematical models.

Johnson et al. (2020) studies two special cases of the *cover-by-pairs optimization problem.* In experiments with synthetic and real-world-based problems, the BRKGA outperforms its only competitor in the *path-disjoint* case, a multi-start greedy algorithm. However, in the *set-disjoint* case, a heuristic-based approximation algorithm is the winner. Londe, Andrade, and Pessoa (2021) uses a BRKGA-MP-IPR (Andrade et al., 2021) with shaking, warm-start, and local search to solve the *p-next center problem.* The authors show that their algorithm has a better performance than the best in the literature, which was a GRASP+VND hybrid. Finally, Villicaña-Cervantes and Ibarra-Rojas (2022) studies the *accessibility location problem for COVID-19 test sites.* The proposed approach has the objective of increasing accessibility. BRKGA obtained solutions with similar quality to a MILP model with a fraction of its computational time.

In general, the adaptations of BRKGA to location problems have some common characteristics. All studies use permutation-based decoders, i.e., the fitness evaluation of the chromosomes involves the sorting of the random keys, except Johnson et al. (2020), which uses an indicator-based decoder, and Freitas et al. (2023), Villicaña-Cervantes and Ibarra-Rojas (2022), whose approaches combine indicator and permutation methods. The great majority of applications also hybridize BRKGA with local search algorithms and use benchmark instances to observe algorithm performance. Lastly, BRKGA tends to outperform or match the best approaches in the literature for problems in this category.

3.4. Cutting and packing

One of the first applications of the defined BRKGA framework was Gonçalves and Resende (2011b), which explores the *container loading problem.* The hybrid BRKGA with multi-populations was shown to perform better than several alternatives from the literature, including GRASP, VNS, genetic algorithms, and tabu search algorithms. Meanwhile, Gonçalves and Resende (2013) studies the *2D and 3D bin packing problems.* Their approach statistically outperforms the ones presented in the literature, which include tabu search, local search, and GRASP+VND approaches. The *3D bin packing problem* was also studied by Zudio, da Silva Costa, Masquio, Coelho, and Pinto (2018), which hybridizes a BRKGA with a VND. This combination was shown to be effective in comparison with pure BRKGA, with better solutions in fewer generations.

Meanwhile, two papers focus on variants of the *cutting and packing problem.* Mundim, Andretta, and Queiroz (2017) first introduces a BRKGA approach, which is then compared favorably with the results of GRASP and MIP models from the literature on benchmark instances. Later, Amaro Júnior, Pinheiro, and Coelho (2017) uses a different strategy in its decoding phase, and either outperforms or matches results found in the literature. Souza Queiroz and Andretta (2020) studies the *2D cutting problem with irregular shaped items.* The authors observe that the BRKGA had solutions of similar quality to the ones from a general VNS. However, the BRKGA was significantly faster than its competitor. Similarly, the *additive manufacturing production planning problem,* which is a version of the 2D irregular packing problem, was explored in Lu, Hu, and Ng (2023). A BRKGA is at the heart of the pixel-based AM packing algorithm (PAMPA). which can be used to check solution feasibility.

Later, Gonçalves and Wäscher (2020) introduces a BRKGA for the *2D non-guillotine cutting problem.* Using benchmarks from the guillotine version, the proposed algorithm is faster and has higher solution quality

than two of the best algorithms in the literature. The same problem was studied in Oliveira and Romero (2022), with an approach based on the one from Gonçalves and Resende (2013). The proposed BRKGA adds a procedure to group plates into blocks. This addition improves performance, with results equal to or better than those previously published. Amaro Júnior, Santos, Carvalho, Araújo, and Pinheiro (2021) uses a BRKGA for the *minimum time cut path problem*. This BRKGA uses a decoder that gives both cut order and direction. The proposed approach outperformed a GA and a commercial software on artificially generated instances. Lastly, Oliveira, Carravilla, and Oliveira (2022) uses a *packing problem with uncertainty* to evaluate performance of a scenario-generating BRKGA. In it, each chromosome represents a possible scenario, which is evaluated regarding diversity and impact on a given first-stage solution.

For packing problems, the BRKGA was mostly used in similar ways. In fact, all proposed BRKGA except Zudio et al. (2018) use permutation-based decoders without any sort of hybridization. Not only that, but almost all articles use benchmark instances, and in all studies, BRKGA was considered the best algorithm with respect to solution quality and computational times.

3.5. Vehicle routing

Grasas, Ramalhinho, Pessoa, Resende, Caballé, and Barba (2014) explores the *blood sample collection problem*. The BRKGA was shown to be an interesting alternative, in comparison with a MIP model, that could be easily used in the practical applications of this study. Three papers focus on *uni- and multi-directional road network problems* (Huang, Santos, & Duhamel, 2018, 2020b). In the most recent one, Huang, Santos, and Duhamel (2020a) adds disruption constraints to the problem. The proposed BRKGA was outperformed by an ILS.

Ruiz, Soto-Mendoza, Ruiz Barbosa, and Reyes (2019) studies the *open vehicle routing problem*. The authors combine a BRKGA with a local search and a strategic oscillation operator. In comparison with several algorithms in the literature and a GRASP, the hybrid BRKGA had the best performance. The work of Ibarra-Rojas and Silva-Soto (2021) focuses on a generalization, the *vehicle routing problem with egalitarian distribution*. The authors compare three decoders and introduce a mathematical model for the problem. Carrabs (2021) studies the *set orienteering problem*. Their BRKGA was hybridized with three simultaneous local searches, a reset operator, and a multi-population strategy. This approach was shown to be highly effective in comparison with a VNS and a matheuristic. Schenekemberg, Chaves, Coelho, Guimarães, and Avelino (2022) focuses on the *dial-a-ride problem with private fleet and common carrier*. To solve this problem, the authors use BRKGA-QL, a hybridization with Q-Learning-based parameter control. They also propose a local search with seven neighborhoods. The hybrid BRKGA-QL+LS dominated a BRKGA-QL and a branch and cut algorithm in performance and computational time on both benchmark instances and a case study.

Schuetz et al. (2022) explores *robot-trajectory planning problems*. A novel BRKGA-based scheme is proposed, where several methods can create solutions for the first generation of the evolution. BRKGA was shown to be faster than a SA applied within a Quadratic Unconstrained Binary Optimization approach but lost in speed to a GA with an SA-based evolution operator. Finally, Marques Jr. et al. (2023) studies the *multi-objective green routing drone grid problem*, where the airspace is divided into horizontal and vertical bands. BRKGA was shown to obtain viable solutions quickly but was outperformed by VND-based methods.

There are some trends common to BRKGA used to solve routing problems. The use of real-life-based instances is a characteristic of this category, with almost half of the studies doing such. Another common trend is the use of permutation-based decoders, done by most articles. Lastly, BRKGA was shown to be the better algorithm on almost all papers, except those in which it is compared with ILS algorithms.

3.6. Traveling salesman problem

Morán-Mirabal, González-Velarde, and Resende (2014) is the first use of BRKGA for any generalization of the TSP. Specifically, the authors study the *family traveling salesman problem*. Their BRKGA approach shows better results than a MIP model and a hybrid GRASP+PR heuristic on smaller instances, but the hybrid GRASP was shown to be better on the larger instances. The same problem is studied in Chaves, Vianna, da Silva, and Schenekemberg (2024), with a chromosome whose last gene indicates which of the five decoding procedures will be used to decode the individual. The BRKGA approach uses Q-Leaning to control its parameters during evolution, alongside a random VND to control a local search with six neighborhoods, and a perturbation component for the crossover between similar individuals. The Q-Learning approach outperformed state-of-the-art algorithms, including the one from Morán-Mirabal et al. (2014), and a parallel branch-and-cut algorithm on larger instances. Later, Bernardino and Paias (2018) addresses the *traveling purchaser problem*. For this, the authors propose two different BRKGA strategies, both hybridized with a local search. In comparison with a hybrid GA+LS, there was no difference in solution quality among the three algorithms, but the hybrid genetic algorithm had considerably smaller computational times.

Silva, Chaves, Yanasse, and Luna (2019) study the *multi-commodity TSP with priority prizes*. The authors propose two hybrid BRKGA with ILS, where the ILS itself is hybridized with a VND. One of the BRKGA approaches is an adaptive BRKGA (A-BRKGA), introduced in Chaves, Gonçalves, and Lorena (2018). In the experiments, the A-BRKGA+ILS was not shown to be statistically better than the BRKGA+ILS, but both had better performances than a MIP model.

The work of Chagas, Toffolo, Souza, and Iori (2020) introduces the *double TSP with partial last-in-first-out loading constraints*. Their BRKGA performs better than two LP models on the proposed instances. Finally, Chagas, Blank, Wagner, Souza, and Deb (2021) focuses on the *bi-objective traveling thief problem*, in which the TSP is combined with a knapsack problem. The bi-objective approach hybridizes a BRKGA with an NSGA-II, alongside warm-start and a local search algorithm. This BRKGA either matched or outperformed several approaches in the literature. Amaro Júnior, Carvalho, Santos, Pinheio, and Celedonio (2023) studies a generalization of the TSP called *laser cutting path planning problem*. In it, the objective is to minimize the time the laser takes between the corners of a cutting form. The authors hybridize A-BRKGA with a perturbation strategy. A Eulerian path heuristic is used to generate initial solutions. The BRKGA approaches outperform a MIP model, though A-BRKGA had worse performance than its classic counterpart.

The studies in this category have great diversity in the choice of instances, with artificially generated, new, and benchmarks being used. This diversity is not extended to BRKGA performance in comparison with other algorithms, which follow a specific trend: namely, BRKGA outperforms most exact approaches, but tends not to be effective with respect to other metaheuristic algorithms. Also in common is the use of permutation-based decoders, as all papers in this category of problems apply.

3.7. Clustering

Festa (2013) uses a BRKGA for clustering biological data. This BRKGA uses an improvement procedure inside its decoder that is similar to a local search approach. Compared with several state-of-the-art algorithms, including variations of hybrid GRASP+PR, the BRKGA was effective and efficient in finding good solutions. This BRKGA was adapted by Oliveira, Chaves, and Lorena (2017) for the *constrained clustering problem*. The authors also add to the algorithm a local search external to the decoding process. In this case, the BRKGA+LS was outperformed by a column generation algorithm hybridized with PR and local search.

Chaves et al. (2018) introduces the adaptive BRKGA with clustering search for the *capacitated centered clustering problem*. The A-BRKGA is a variation that employs on-line parameter tuning. Its combination with clustering search was shown to have the best performance compared to a classic BRKGA and other state-of-the-art algorithms. This problem is also explored with A-BRKGA in Xu, Guo, and Zeng (2022), but with an iterative neighborhood local search. The novel approach is shown to improve known solutions with a small increase in running times.

Martarelli and Nagano (2020) uses BRKGA for *unsupervised feature selection*. The authors introduce two variations of BRKGA. The difference between the two was the use of warm-start. BRKGA was the best approach compared with others in the literature, but the inclusion of warm-start solutions led to better results. The paper by Fadel, Ochi, Brito, and Semaan (2021) studies the *multivariate micro aggregation problem*, in the context of statistical disclosure control. The authors introduce two decoders for this problem. The proposed BRKGA was compared with several other methods and consistently found better results than its competitors. Finally, Brito, Fadel, and Semaan (2020) proposes a BRKGA for the *k-medoids clustering problem*. The authors detail a crossover operator that works as a local search with best improvement strategy. Tests show the efficacy of the approach in comparison with several from the literature.

There are some characteristics in common in the papers that focus on clustering. Most of the article use indicator-based decoders. All articles test their proposed approaches on benchmarks or known datasets, and most of them hybridize BRKGA with local or clustering search algorithms.

3.8. Graph problems

Clique problems are the focus of several articles (Fontes, Gonçalves, & Fontes, 2018; Pinto, Ribeiro, Riveaux, & Rosseti, 2021; Pinto, Ribeiro, Rosseti, & Plastino, 2018). In the most recent one, Melo, Ribeiro, and Riveaux (2023) studies the *minimum quasi-clique partitioning problem*. Their approach was shown to obtain results at least as good as the ones in the literature and to find better results for a fifth of the instances.

Lima, Aquino, Nogueira, and Pinheiro (2022) focuses on the *minimum broadcast time problem*. The proposed approach combines a matheuristic-based BRKGA with two possible decoders. Experiments show that the matheuristic outperformed state-of-the-art methods in quality and computational time. Silva, Ribeiro, and dos Santos Souza (2023) tackles the *chordal completion problem*. The proposed BRKGA uses a permutation-based decoder to eliminate possible orderings of triangulation in the graph. In a comparison with the winner algorithm of the Parameterized Algorithms and Computational Experiments Challenge 2017, BRKGA was shown to find feasible solutions for all instances in which the model failed and to improve solution quality. Finally, Londe, Andrade, and Pessoa (2022) introduces the *root sequence index allocation problem*. This telecommunication problem is tackled as a generalization of the classical vertex coloring problem. The authors customize three decoding approaches, two local searches, a warm-start procedure, and shake and reset operators for two versions of this problem, with differing objectives. BRKGA was shown to obtain better solutions than an ILS and VNS on one of the objectives and to be comparable to VNS on the other.

For this category of problem, all BRKGAs use permutation-based decoders on benchmark instances except Londe et al. (2022), which proposes an indicator-based decoder and introduces new instances. Also noteworthy is the quality of the obtained solutions, with BRKGA consistently outperforming several state-of-the-art algorithms.

3.9. Parameter optimization

In Caserta and Reiners (2016), the BRKGA is used to tune a cross entropy-based scheme, itself used to solve a binary classification problem. Cicek and Ozturk (2021) uses a BRKGA to *optimize ANN parameters*

with time series forecasting. The algorithm was successful with respect to several classical approaches, such as SARIMA, SVR, ARIMA, and classical GA, but had a clear disadvantage in computational times.

The work of Sun and Zhou (2022) focuses on *Bayesian network structure learning*. The authors use the algorithm NOTEARS as both a decoder and local optimizer to solve the non-convex problem. The proposed BRKGA can achieve a good performance on benchmark networks and on a real data set in comparison with pure NOTEARS, classical GA, and hybrid PSO+GA. Falls, Bernardello, Castrillo, Acosta, Llort, and Galí (2022) uses BRKGA to *estimate the ideal set of parameters* to simulate the behavior of the biogeochemical component of an ocean model. In experiments, the authors observe the ability of BRKGA to create simulations and compare those with real data. Finally, Japa et al. (2023) optimizes *hyperparameters for neural networks*. BRKGA+Bayesian Walk exploitation procedure was shown to produce better results in a more consistent manner than several approaches from the literature.

Some conclusions may be observed for the papers in this category of application, such as that all proposed BRKGA approaches use indicator-based decoders. One point that is always observed is the speed of the algorithm in comparison with others from the literature to solve both benchmarks and real-life datasets. It is also of note that BRKGA tends to outperform other metaheuristic approaches for problems in this category.

3.10. Container loading

Gonçalves and Resende (2012) introduces a multi-population BRKGA for the *3D container loading problem*. This algorithm had the best overall performance in comparison with several other approaches, including genetic algorithm, branch-and-bound, tabu search, and GRASP. Later, Zheng, Chien, and Gen (2015) studies the same problem, also with a multi-population BRKGA, but with a bi-objective spin. The bi-objective approach outperformed several variants on benchmark instances.

Ramos, Oliveira, Gonçalves, and Lopes (2016) hybridizes a multi-population BRKGA with a constructive heuristic, an approach shown to be effective in comparison with several other algorithms. These algorithms include the one in Gonçalves and Resende (2012). This multi-population BRKGA was later extended by Ramos, Silva, and Oliveira (2018) to consider other necessary constraints, leading to improvements in solution quality and efficiency.

Even though the container loading problems are very similar, in concept, to packing problems, the authors of the former problem use different strategies than those of the latter. In this category, it is of note the extension of already known and well-performing BRKGA approaches to consider new problem characteristics. In fact, the comparison of performance among proposed algorithms is frequent in this category, alongside the use of known benchmarks, permutation-based decoders, and multi-population strategies.

3.11. Other problems

This subsection details studies that focus on problems that are not included in any of the previous categories. The first article, Gonçalves and Resende (2004), uses a hybrid BRKGA+LS for *manufacturing cell formation*. Resende et al. (2012) studies the *Steiner triple covering problem*. Another application of BRKGA presented by Silva, Resende, and Pardalos (2014) is to solve *non-linear systems of equations with multiple roots*. Two studies focus on *berth allocation problems*. The first, Lalla-Ruiz, González-Velarde, Melián-Batista, and Moreno-Vega (2014), outperformed both a tabu search and several branch-and-price approaches. Later, Correcher and Alvarez-Valdes (2017) hybridizes a BRKGA with two distinct local searches and a matheuristic.

Similarly, the *unit commitment problem* is studied in two papers. Roque, Fontes, and Fontes (2014) introduces a BRKGA+LS hybrid heuristic. Roque, Fontes, and Fontes (2017) studies a multi-objective

version of the previously mentioned problem. The authors use the same decoder as in the previous study, but in a multi-objective BRKGA. Andrade, Toso, Resende, and Miyazawa (2015) focuses on the *winner determination problem in combinatorial auctions*. The authors use a BRKGA with warm-start solutions originating from a LP model, alongside three decoding strategies. Chan, Tibrewal, Prakash, and Tiwari (2015) tackles the *multi-item capacitated lot sizing problem*. The *cloud resource management problem* is the focus of Heilig, Lalla-Ruiz, and Voß (2016).

Gonçalves, Resende, and Costa (2016) studies the *minimization of open stacks problem*. Chaves, Lorena, Senne, and Resende (2016) tackles the *minimization of tool switches problem*. The proposed approach hybridizes BRKGA with a clustering search algorithm. Hottung and Tierney (2016) focuses on the *container pre-marshalling problem*. Andrade, Ahmed, Nemhauser, and Shao (2017) hybridizes a BRKGA with a feasibility pump to find *feasible solutions to MIP*. The *protein–ligand flexible molecular docking problem* is studied by Leonhart, Spieler, Ligabue-Braun, and Dorn (2019).

Oliveira, Carravilla, and Oliveira (2018) study the *car rental problem*. The proposed methodology includes a warm-start with constraint programming and integer non-linear programming models, linear programming constraints included in the decoding process, and a MIP model applied to the best solution found by BRKGA. Oliveira, Carravilla, Oliveira, and Costa (2019) explores the *car rental capacity-pricing stochastic problem*. The authors combine a BRKGA with a matheuristic to generate solutions for a two-stage stochastic problem and show the efficiency of the proposed approach in several instances. The study of Brito et al. (2020) focuses on *optimal allocation in stratified sampling*. Pinacho-Davidson and Blum (2020) tackles the *minimum capacitated dominating set problem*. The proposed BRKGA uses an ILP solver inside its decoder and outperforms a state-of-the-art local search heuristic and a MIP model.

Pan, Zhang, Yan, Lwin, and Skibniewski (2021) uses BRKGA in the decision-making phase of a framework to *mitigate Covid-19 spread*. Ochoa, Malan, and Blum (2021) compares the *search trajectory network* of several metaheuristics. BRKGA is used in the combinatorial optimization case study, in which a *p*-median problem was explored. The authors observe that BRKGA quickly gets trapped in different areas of the solution space and visits a higher number of solutions compared to other metaheuristics. The work of Pastore, Menna, and Asprone (2022) studies *topology optimization for stress-constrained structures*. This problem is in the context of additive manufacturing, and the authors use Bézier curves to obtain the best paths. To test the results, the authors made a reduced-scale print of a piece and observed its structural capabilities. Silva, Leite, et al. (2023) explores the *maximum diversity problem*. Experimental results in benchmarks indicate that a hybrid BRKGA+LS outperforms other approaches from the literature in terms of solution quality. Finally, Morgan, McGrath, and Weck (2023) searches for *multi-spacecraft maneuvers for mobile target tracking*. Two case studies point out that a BRKGA-based method outperforms its competitors but with diminishing returns for greater spacecraft numbers.

3.12. Summary

The studies detailed in this section highlight several points in regard to BRKGA use and characteristics. First, the decoders are frequently permutation-based, i.e., use sorted random-keys as a part of the decoding process. Second, the use of hybridization is noted to improve BRKGA performance, especially when a local search or matheuristics are used. Third, the majority of papers use benchmark instances, and compare the proposed algorithms with state-of-the-art approaches. Fourth, BRKGA almost always outperforms MIP models and tends to obtain similar or better results than other algorithms. Fifth and last, authors do not frequently build on already existing BRKGA approaches, instead introducing new decoding processes or operators.

The scenario shown in this section can be summarized as such: BRKGA is a versatile algorithm, used successfully in several problems, and that improves significantly when hybridized. The use of hybridization is, thus, crucial for an efficient and well-performing BRKGA, and is the focus of the next section.

4. Main hybridizations with BRKGA

To observe the evolution of hybridization with BRKGA, as pointed out by the second RQ, another detailed reading of the selected papers was performed. In this case, the objective was to observe and detail other algorithms or frameworks simultaneously used with BRKGA to solve different optimization problems. We explore 94 papers that use diverse hybrid strategies. For illustrative purposes, the number of studies over the years can be seen in Fig. A.3, while Fig. A.4 shows the total amount of papers in each category. Please note that papers can be included in more than one category of hybrid.

4.1. Local search heuristics

The complementary use of two or several heuristics to explore all available knowledge is the basis of memetic algorithms (Moscato et al., 1989). The use of local search (LS) algorithms as the intensification phase of an evolutionary algorithm is a well-known application of the memetic framework, and tries to mimic the idea of Lamarck evolution, that individuals may pass characteristics obtained during their lives to their descendants (Cotta, Mathieson, & Moscato, 2018). Several of the papers that apply local search alongside BRKGA do not give extensive details about LS neighborhood or stopping criteria, or when it is applied in the main loop of the algorithm. Details on the complexity of the LS are also, in general, scarce. Thus, in the first part of this section, the use of simpler local searches – not based on specific heuristic frameworks – is addressed. They are divided by which module of the BRKGA the LS is embedded in and by the neighborhood explored. The second part of this section explores the application of known heuristics as the LS of BRKGA.

In BRKGA, the use of local search algorithms may be divided into two categories, indicating in which phase of the GA it is used. The most frequent approach is the use inside the decoding phase, either before, e.g., Festa (2013) Fontes and Gonçalves (2007, 2013), Gonçalves and Resende (2014), Gonçalves et al. (2011), Heilig et al. (2016), Morán-Mirabal et al. (2013), Ruiz et al. (2015), Silva et al. (2014), Stefanello et al. (2017), Valente and Gonçalves (2009), Londe et al. (2022), and Maecker et al. (2023) or after, e.g., Gonçalves and Almeida (2002), Gonçalves et al. (2005), Gonçalves and Resende (2004), Lopes et al. (2016), Malve and Uzsoy (2007), Ruiz et al. (2019), and Valente et al. (2006) the calculation of the fitness of the chromosome.

Outside of the decoding phase, the local search is usually applied to some of the best solutions on a specific number of generations. This is due to the fact that the application of local search on all individuals may be too computationally expensive for specific problems, especially if the mentioned local search has a large neighborhood. In this category, several papers apply local search on all offspring generated by the crossover process (Brito, Semaan, & Fadel, 2022; Buriol et al., 2010, 2005, 2007; Oliveira et al., 2017; Ruiz et al., 2011; Silva, Leite, et al., 2023). Other papers only use local search after the evolutionary process is completed, often on the best incumbent solutions (Bernardino & Paias, 2018; Biajoli et al., 2019; Chagas et al., 2021; Mauri et al., 2021; Roque et al., 2014). Finally, some papers select special chromosomes to apply local search, usually only on selected generations (Abreu et al., 2022; Carrabs, 2021; Japa et al., 2023; Kashan et al., 2006; Lalla-Ruiz et al., 2016; Londe et al., 2021).

One can also note that there is some variation among the explored neighborhoods. In fact, one can observe three options of moves among the used local searches. The first is the exchange between items of different categories, which is, by far, the most common approach.

This includes the exchange of items in different categories, e.g., Biajoli et al. (2019), Buriol et al. (2010, 2005, 2007); Gonçalves and Almeida (2002); Fontes and Gonçalves (2007, 2013); Gonçalves and Resende (2004); Gonçalves et al. (2011); Carrabs (2021); Kashan et al. (2006); Londe et al. (2021); Malve and Uzsoy (2007); Mauri et al. (2021); Oliveira et al. (2017); Reis et al. (2011); Roque et al. (2017); Ruiz et al. (2015) and Stefanello et al. (2017), and creation or modification of the categories themselves, e.g.,Festa (2013); Gonçalves and Almeida (2002), and Ruiz et al. (2015).

The second possible approach is comprised of the swap of items in the same category. This move is more frequently used on problems whose solution is permutation-based, i.e., based on the ordering of items such as jobs in a schedule (Abreu et al., 2022; Maecker et al., 2023; Valente & Gonçalves, 2009; Valente et al., 2006) or locations (Bernardino & Paias, 2018; Chagas et al., 2021; Lalla-Ruiz et al., 2016; Lopes et al., 2016), often with adjacent swap moves.

The third commonly used neighborhood is composed of a greedy insertion or removal of the items of a solution. That is, the most advantageous item, in regards to the fitness value of the solution, is modified (Bernardino & Paias, 2018; Carrabs, 2021; Gonçalves & Resende, 2014; Heilig et al., 2016; Londe et al., 2022; Lopes et al., 2016; Silva et al., 2014).

Several other search strategies used in BRKGA are derived from other well-known heuristics. The tabu search (TS) strategy first appeared in Glover (1977) and was first named as such in Glover (1986). This method considers adaptive memory and responsive exploration, by forbidding certain moves in a neighborhood with either short- or long-term memory. The study of Gonçalves and Resende (2014) combines a tabu search with a local search based on Akers' graphical method (Akers, 1956).

A variable neighborhood descent (VND) is a search strategy that explores different neighborhoods in a deterministic way (Duarte, Sánchez-Oro, Mladenović, & Todosijević, 2018). It is a variant of the variable neighborhood search (VNS) framework introduced in Hansen and Mladenović (2005). As a search method, VND has been frequently hybridized with BRKGA. Pedrola, Careglio, et al. (2013) uses a VND after the application of a path relinking heuristic that explores, on some high-quality solutions, every N-move neighborhoods after a set number of generations. Stefanello et al. (2019) similarly applies the VND every generation, after the decoding step and the use of path relinking. The authors use four neighborhoods: shift, swap, chain2L, and chain3L, with the last two being chained shift moves.

Pessoa and Andrade (2018) applies a two-neighborhood VND inside the decoding process for a flowshop scheduling problem. Those neighborhoods are of the insert type, which moves jobs to new positions, and of the interchange type, which swaps two jobs. Andrade, Silva, and Pessoa (2019) uses a similar strategy but with a caveat: as the insert neighborhood acutely increases computational effort, it is restricted to only one interaction. The study of Soares and Carvalho (2020) also inserts a VND into the decoder, with three search moves: insertion, interchange, and one-block grouping. The last neighborhood is specifically tailored to the problem characteristics.

Another hybridization of BRKGA and VND is shown by Zudio et al. (2018) for a bin packing problem. The authors combine the VND with the crossover operator. The combination of BRKGA and VND is also observed in Soares and Carvalho (2022) for a scheduling problem. This heuristic is applied to all individuals of the elite set, with four neighborhoods of insertion, exchange, relocation, and block grouping. In the same work, a simple local search is applied to all unexplored individuals of the non-elite set.

A random VNS (RVNS) is used by Schenekemberg et al. (2022). After a set amount of generations, a label propagation method is used to identify communities of similar individuals. The RVNS is applied to the unexplored best solutions of each community, by randomly selecting one of the seven neighborhoods to explore. This approach is used with six neighborhoods by Chaves et al. (2024). In this case, the application

of the RNVS only occurs when the current generation is a multiple of $n/5$, with n being the number of nodes in a TSP problem. A VNS is used alongside the PAMPA framework proposed in Lu et al. (2023). To test the effectiveness of the BRKGA-based framework, it was applied to the solutions obtained by a VNS algorithm.

The clustering search approach (Oliveira, Chaves, & Lorena, 2013) identifies interesting individuals in the search space using clustering strategies and applies a local search to those individuals. The studies of Chaves et al. (2016) and Silva et al. (2019) use a label propagation algorithm (Raghavan, Albert, & Kumara, 2007) to find interesting solutions. This method constructs and iteratively labels a graph of the solution. The local search applied by Chaves et al. (2016) considers two moves: the removal and insertion of items in the clusters, and the swap of items in different clusters. Meanwhile, Silva et al. (2019) uses an iterated local search (ILS) (Lourenço, Martin, & Stützle, 2003), whose perturbation phase is based on swapping one-on-one items from different categories, and whose local search phase is a variable neighborhood descent (Hansen & Mladenović, 2005) with three neighborhoods: the two from Chaves et al. (2016), and a two-opt in which adjacent items are exchanged. In both papers, the use of the clustering search improves solution quality without significantly increasing computational times.

An adaptive large neighborhood search (ALNS) (Pisinger & Ropke, 2007) is hybridized with BRKGA in He et al. (2019) for a scheduling problem with setup times. The ALNS is used after the decoding process, once for every solution that reaches a minimum fitness threshold. The ALNS uses five removal operators: random, minimal revenue, minimal unit revenue, maximal setup time, and worst sequence removal. The insertion operators are based on maximal revenue and maximal unit revenue.

An iterated greedy algorithm (IGA) (Ruiz & Stützle, 2007) is used in the intensification phase of the BRKGA proposed by Abreu et al. (2021). This algorithm is applied after a set number of iterations to the best solution of that generation. It is based on the removal of a percentage of the operations of a solution, followed by the reconstruction of the solution. The hybrid approach had better performance than a pure BRKGA.

In Xu et al. (2022), two Iterative Neighborhood Local Search Algorithms are introduced, one inexact and the other, exact. The inexact INLS is used on Individuals selected by a label propagation method after a set amount of generations. After the end of the evolutionary process, the exact INLS is used on the incumbent solution. The authors observe that this addition improves quality and performance in comparison with state-of-the-art methods.

As this section shows, there are several ways of combining BRKGA and local search heuristics. Generally, this hybridization manages to improve solution quality at an increase in computational effort.

4.2. Warm-start

The classical BRKGA framework creates the initial population with randomly generated individuals. Nonetheless, the introduction of good solutions in the initial population has been noted to improve the performance of the algorithm. Studies in this category create one or several initial high-quality solutions and introduce them to the initially generated population as individuals.

Generally, this approach uses known heuristics from the literature to generate the initial solutions. Ericsson et al. (2002) does this, using known heuristics to generate two solutions, while Buriol et al. (2005) generates one initial solution using a simple heuristic. It is also done by Valente et al. (2006), but, in this case, three solutions are created. The authors show that the use of good individuals improves the performance of the algorithm. Pessoa and Andrade (2018) introduce in the initial population one solution from one effective heuristic from the literature. Meanwhile, Andrade, Silva, and Pessoa (2019) use diverse solutions of a known heuristic from the literature. A similar approach is used by Lopes et al. (2016). Martarelli and Nagano (2020) creates

half of the initial population randomly. In the other half, the first and second quartiles and the first solution from the third quartile receive solutions from three known heuristics. This strategy is shown to improve algorithm performance and convergence rate. The study of Abreu et al. (2021) uses a greedy insertion algorithm to create two solutions for the initial population. Chagas et al. (2021) use known effective algorithms to generate solutions to both sub-problems (knapsack and TSP). Then, those solutions are combined to generate complete solutions for the traveling thief problem. In Xie et al. (2022), three solutions come from algorithms from the literature, while all other initial individuals are obtained from a level heuristic. Londe et al. (2022) uses a customized heuristic to introduce one individual in the first population. Amaro Júnior et al. (2023) uses an Eulerian heuristic to generate initial individuals for a TSP problem. Lastly, the insertion of solutions from two constructive heuristics in Silva, Ribeiro, and dos Santos Souza (2023) is shown to increase algorithm performance.

An alternative to the use of heuristics to generate initial solutions is the use of mathematical models. Andrade, Toso, Resende, and Miyazawa (2015) introduces solutions from a linear relaxation of the MIP formulation of the problem. The model interactively fixes genes to create several solutions, and its use leads to better performance on all proposed decoders. Oliveira et al. (2018) use three strategies to generate warm-start solutions: decomposing the complex integer non-linear programming model to a simpler one and solving it to optimality, relaxing the model's integrality constraints and solving it to optimality, and constructing naive solutions. The work of Chagas et al. (2020) uses the model of a less constrained version of the problem to create a possible solution. This solution is the optimal solution of the relaxed model. The paper from Schuetz et al. (2022) uses solutions from several alternative algorithms in their initial population. The methods include linear programming, quantum annealing, and greedy algorithms, among others.

An additional use of warm-start is when the proposed strategy demands specific characteristics improbable to occur in random individuals. The paper of Kashan et al. (2006) on single batch processing illustrates this situation, as the proposed BRKGA uses a decoding strategy that needs specific batch sizes to be effective, something that is improbable in random individuals. This means that, for that decoder, the entire initial population is generated by a robust heuristic. This heuristic aims to reduce the chance of BRKGA converging to bad solutions, by modifying the number of batches associated with them. Zhang et al. (2022), meanwhile, uses orthogonal design to create the initial population. This is shown to increase convergence rate and solution quality in comparison with random-generation of initial solutions.

One may observe that warm-start solutions have been shown to improve performance and convergence rate, independently of which methods are used to create the solutions. More than half of the studies with this hybridization use known algorithms from the literature, while the use of mathematical programming is also frequent.

4.3. Other heuristics

Path Relinking (PR) (Glover, 1997) is an intensification strategy that explores the neighborhood in the path between two distinct solutions (Ribeiro & Resende, 2012). Pedrola, Careglio, et al. (2013) uses this strategy alongside a VND. The evolutionary process of the BRKGA is stopped after a set number of generations when all solutions in the elite population have a chance of being selected for the so-called elite set. In this set, path-relinking and the VND can be applied to all solutions, with probability proportional to the distance between them and the guide solution. The PR used has a back-and-forth strategy, in which guide and base solutions change places after the first path is completed. Stefanello et al. (2019) also uses PR alongside VND. In this case, both are applied after the decoding step on all generations. If a better solution is found, then the corresponding chromosome

is rewritten. Both studies point out that the hybridization of path-relinking, VND, and BRKGA leads to better results without a severe increase in computational effort. Later, Schuetz et al. (2022) uses PR to refine high-quality solutions of the last population of BRKGA.

The differential evolution (DE) algorithm is inspired by geometry instead of by nature (Storn & Price, 1997). Kong, Liu, et al. (2020) uses the JADE (Zhang & Sanderson, 2009) framework as the crossover operator of a BRKGA. This crossover is also of note as it considers three possible parents: one from the elite set, one from the non-elite set, and one randomly chosen among all individuals. The crossover probability comes from a normal distribution, while the mutation chance is derived from a Cauchy distribution. This addition causes longer running times but also improves solution quality and convergence rate in comparison with a pure BRKGA.

Queiroga et al. (2021) uses a dynamic programming (DP) approach (Bellman, 1966) as the decoding process of a proposed BRKGA. Specifically, the authors use the approach proposed in Chou and Wang (2008). This hybrid is shown to have worse performance than other approaches for the given problem.

The Flower Pollination algorithm (Yang, 2012) is used in the crossover operator of Yu et al. (2023). In it, there are three possible crossover combinations in each application. First, the global pollination strategy considers the combination of the current solution with the best known solution. Second, the local pollination scheme combines the current solution with one elite and one non-elite individuals. Third and last, a parameterized uniform crossover can be used. This approach had success in comparison with pure BRKGA and FPA.

4.4. Multi-objective

Multi-objective optimization problems bring challenges to the BRKGA framework, as the competition between objectives demands special adaptations to items such as decoding processes and chromosome ordering and selection. One such adaptation is the work of Zheng et al. (2015). In the chromosome ordering phase, their bi-objective algorithm uses a Pareto ranking process with the average of the weighted objectives. The weights used change at each generation, as they depend on their proportion among the current values of the complete population.

Tangpattanakul et al. (2015a, 2015b) also propose a bi-objective approach with modifications inside the chromosome ordering phase. The authors experiment with methods inspired by successful multi-objective algorithms, such as NSGA-II (Deb, Pratap, Agarwal, & Meyarivan, 2002), SMS-EMOA (Beume, Naujoks, & Emmerich, 2007), and IBEA (Zitzler & Künzli, 2004), with results pointing to better computational times of the one inspired by IBEA. The authors also conclude that a hybrid decoder – in which several algorithms are used as part of the decoding process, and the best solution among all is used as the chromosome fitness value – is more effective than simpler decoding heuristics. Meanwhile, Roque et al. (2017) indicates that a selection method based on NSGA-II can also be effective for a bi-objective BRKGA.

The bi-objective BRKGA of Cabo et al. (2018) also uses the hybrid decoder strategy of Tangpattanakul et al. (2015a, 2015b), but with a twist. In this case, one of the objective functions refers is the number of batches in the scheduling problem. After the decoding process is performed, this batch number is inserted into the last genes of the chromosome, while the other objective is used by the algorithm as the fitness of the chromosome.

Both papers of Huang et al. (2018, 2020a) focus on the same bi-objective problem, but with different approaches. Huang et al. (2018) uses an aggregated fitness function, with weights α and $1 - \alpha$. This algorithm also updates the Pareto front every time a novel non-dominated solution is found. The other paper only has a modification regarding

the aggregated fitness value of the chromosomes inside the decoding process.

The studies by Raposo et al. (2020), Mikulski and Tomczewski (2021), Bolsi et al. (2022), and Marques Jr. et al. (2023) show the only BRKGAs adapted to more than two objective functions. Raposo et al. (2020) consider a three-stage problem, that may be solved in two ways: the first stage is solved independently, then the two others are solved together, or all three stages are solved concurrently. The authors use the concept of Pareto dominance to select solutions and construct the Pareto front. The work by Mikulski and Tomczewski (2021), meanwhile, modifies the elite set of the BRKGA for a multi-objective problem. In this case, the elite individuals are non-dominated solutions, while the non-elite chromosomes are dominated. This adaptation of the framework means that the selection and crossover operators are modified. The three objectives of Bolsi et al. (2022) are considered in a lexicographic manner, with three decreasing weights indicating preference. In Marques Jr. et al. (2023), an aggregated fitness function is used for their three objectives. This means that a single-objective VNS could be used in their experiments with little modification.

Silva-Soto and Ibarra-Rojas (2021) is the only study in which a bi-objective BRKGA approach is compared with a single-objective BRKGA. For the bi-objective algorithm, the authors combine BRKGA with a hierarchical non-dominated sorting algorithm for the chromosome sorting phase. For the comparison, the BRKGA uses an aggregated fitness function with weighted sums, in which $\alpha = 0.5$. The hybrid approach was proven to be more efficient in finding the Pareto front than its competitor.

Chagas et al. (2021) adapts the BRKGA to a bi-objective problem by hybridizing it with NSGA-II. This is done in the chromosome selection phase to join the strengths of both approaches for solving combinatorial optimization problems.

A novel multi-objective framework for BRKGA is introduced by Fontes et al. (2023) and Fontes and Homayouni (2023). The multi-population BRKGA (mp-BRKGA) evolves $\Omega + \Pi$ populations independently and in parallel. Each Ω population considers only one of the $|\Omega|$ objectives, while the remaining Π populations observe all objectives simultaneously. This algorithm borrows ideas from NSGA-II to create and rank Pareto fronts, but unlike NSGA-II, mp-BRKGA always allows offspring and mutant solutions to pass tho the next generations. The Ω populations use the classical BRKGA evolutionary process. Meanwhile, for each the $\pi \in \Pi$ populations the elite set is chosen from a pool of solutions, which consists of the best solutions of the previous Ω populations and the best individual of the given π population. After a set number of generations, all Π populations contribute with their best elite solutions to the others' pools, from which repeated chromosomes are removed. In tests, this approach was proven effective in comparison with NSGA-II and is able to cover the whole Pareto fronts of the instances successfully.

For this type of hybrid, some trends may be observed. Generally, BRKGA is adapted to bi-objective problems, with modifications to its chromosome ordering and selection phase and, more rarely, to the decoding process. The multi-objective algorithm is rarely compared with its single-objective counterpart and frequently uses concepts from effective multi-objective algorithms to improve performance. The use of aggregated fitness functions with weighted sums is also present in a third of the studies.

4.5. Matheuristic

There are several ways to combine mathematical formulations and heuristics to improve solution quality. A matheuristic is defined as the hybridization of metaheuristics and mathematical programming techniques (Boschetti, Maniezzo, Roffilli, & Röhler, 2009). These methods use exact approaches as tools within the heuristic framework (Fischetti & Fischetti, 2018). In BRKGA, mathematical programming is frequently applied to the decoding process. One such work was authored by Gonçalves and Sousa (2011), which uses a mathematical model as part of the decoder. After an individual is decoded, it goes to a surrogate model, which is then solved by an LP solver. The surrogate's objective value is used as chromosome fitness. A similar process is used by Gonçalves and Resende (2015), with a slight difference. In this case, the LP model is only applied if the solution given by the chromosome is promising. The same idea of using a chromosome solution to obtain an easier surrogate model is also applied by Oliveira et al. (2018). This proposed algorithm applies a MIP solver to the best solution found by the BRKGA. Oliveira et al. (2019) uses a model inside the decoder to find the objective value of the second-stage problem, given the first-stage solution and possible scenarios. Lastly, Sun and Zhou (2022) uses the NOTEARS algorithm inside its decoder for Bayesian network structure learning. This algorithm considers the optimization of the dual version of the equivalent problem, which is also non-convex.

An alternative use of mathematical models inside BRKGA is in the form of exact local searches. An example of this is Correcher and Alvarez-Valdes (2017). The exact local search is applied to all individuals at the end of each generation for a berth allocation problem, as long as the time limit is not exceeded. This local search removes a cluster, then uses a MILP model to reallocate its items, while considering items in other clusters as fixed. This exact local search outperforms other methods and the matheuristic has better solution quality than a pure BRKGA. A similar strategy is used by Mönch and Roob (2018). In this case, a model-based heuristic is applied if a solution is worse than a percentage of a reference value. Lastly, Pinto et al. (2021) uses an exact local search. In this case, it is internal to the decoder, with a given probability of occurring in each decoding process. In comparison with a non-exact decoder, the matheuristic leads to consistently better solutions.

Andrade, Ahmed, et al. (2017) combines BRKGA with a feasibility pump heuristic to find feasible solutions to mixed integer programs. The proposed framework has three phases. BRKGA is used with a feasibility pump decoder in the evolutionary phase. After a few generations, the local MIP search phase occurs. It is applied to a pool of the best solutions, where solution quality is evaluated in regard to the number of non-integer variables. Meanwhile, the fixing phase reduces the problem's size by fixing variables that have the same values between relaxation and rounding. The use of the evolutionary phase is noted by the authors to lead to better results.

Lastly, Pinacho-Davidson and Blum (2020) uses mathematical models in a novel way. In each generation, some of the solutions are selected to be merged. This selection may be elitist, with only elite solutions picked, or random. Afterwards, a MIP model is applied to the merged solution. The optimized solutions are then inserted back to the algorithm as a new chromosome, and the worst individuals in the population are eliminated. A solution-merging strategy is also used in Lima et al. (2022). After some generations without improvement, solutions are merged and to create a sub-graph with the original vertices and the solutions' edges. An ILP model is used to optimize the problem induced by this sub-graph, and the obtained solution is added to a pool of best solutions.

Considering BRKGA-based matheuristics, there are some characteristics in common. One is using models inside the decoding process, which occurs in more than half of the studies. Moreover, comparisons between exact and non-exact heuristics are rare and only occur when the hybrid uses exact local searches.

4.6. Machine learning

Machine learning techniques may be used in conjunction with metaheuristics to improve performance by extracting useful knowledge

within the search process (Karimi-Mamaghan, Mohammadi, Meyer, Karimi-Mamaghan, & Talbi, 2022). One of its uses is for online parameter control, something that Chaves et al. (2018) applies to BRKGA. The proposed approach, called Adaptive BRKGA (A-BRKGA), updates parameters for population size, elite proportion, mutant proportion, crossover probability, and maximum number of generations at each iteration. It also uses two self-adaptive parameters, named α and β. The former indicates how many individuals in the elite restricted control list (RCL) evolve, while the latter is used as a probability of perturbing similar individuals. Parameters β and crossover probability are included in the chromosome, thus evolving alongside the iterations. The crossover probability used is always the one belonging to the non-RCL parent.

Among the parameters, population size, mutant proportion, and α decrease with each iteration, while elite proportion increases. Maximum number of generations proportional to user-set parameter γ and the minimum and maximum population sizes. In comparison to a pure BRKGA, A-BRKGA outperformed its competitor.

With no changes, the same framework is used by Silva et al. (2019). Again, the A-BRKGA performed better than a pure BRKGA, but the difference in results was not statistically significant. Later, it is also successfully applied in Amaro Júnior et al. (2023), Xu et al. (2022).

Schenekemberg et al. (2022) introduces BRKGA-QL, a novel online parameter control scheme. In it, a Q-Learning algorithm (Watkins & Dayan, 1992) is used to control the same parameters as A-BRKGA with a Markov Decision Process in an environment in terms of actions, states, and rewards. The Q function is used to map state–action pairs to possible rewards, with an exponential decreasing η-greedy policy to chose following actions. This policy gives $1 - \eta$ chance of choosing the action with highest reward to increase intensification, and η probability of increasing diversification by choosing randomly. In this case, η gradually decays during the evolutionary process. This framework is also used in Chaves et al. (2024), and in both cases was shown to be an interesting alternative in comparison with state-of-the-art methods.

Another approach using machine learning methods is Pan et al. (2021). The proposed framework has a predictor creation phase, using a random forest algorithm. The results from this phase are then used with a BRKGA to formulate the ideal mitigation strategy for COVID-19 spread.

Xie et al. (2022) proposes an adaptive decoding procedure. With three possible decoders, this procedure increases chance of a simpler decoding process being used by late evolution. This means that, in early generations, more complex decoding processes tend to be used and thus increase the chance of converging to local optima, while in late generations there is a higher chance of doing a global search in the solution space.

4.7. Stochastic

The first work in this category is Oliveira et al. (2019). This work solves a two-stage stochastic optimization problem using two co-evolving populations. The first population evolves solutions for the first-stage problem. In the decoding process, the first-stage solutions proposed by the chromosome are fixed as part of an approximate LP model. This model is solved to optimality. The other population evolves scenarios, with a focus on diversification. The fitness function of the second population, thus, is a measure of how much the decoded scenario differs from the others. The link between populations is the calculation of total cost between first-stage solutions and scenarios. The authors note that this approach does not consider probability distributions among the scenarios, but also that it provides the decision-maker with several solutions to a multi-stage problem with uncertainty.

Later, the same authors introduce a method of scenario generation in Oliveira et al. (2022). In it, individuals represent possible scenarios and their fitness is measured by the variation of their impact on the first-stage solution, as calculated by crowding distance. There are three possible cases to create the scenarios: one for solution with perfect

information, in which one first-stage solution is developed for each scenario; one for average scenario, in which one solution is generated and used to observe scenario impact on all generations; and one for moving average scenario, where a new first-stage solution is created for each new generation.

4.8. Summary

As shown in this section, the use of hybridization alongside BRKGA is a widespread practice that frequently improves both solution quality and convergence rate, but at the cost of higher computational effort. The explored papers also show that it is a versatile metaheuristic, capable of being combined in several ways with other frameworks. Of note is the use of local search heuristics as intensification strategies, done by a third of all studies and shown to improve performance.

There are some limitations observed in the studies presented in this section. The use of hybridization encumbers an already computationally heavy algorithm. This preoccupation with computational effort may be the reason for the limited number of papers in the machine learning and stochastic categories, as the techniques associated with both hybrids also have a high computational burden. Different functionalities inside the BRKGA framework might be a solution to mitigate this impact, as shown in the next section.

5. New features added to BRKGA

Several new features were added to the BRKGA framework in the ten years since its formal introduction. This is the focus of the third research question. The difference between a hybridization and a feature is that the former defines items added from outside the BRKGA framework that demand modifications of the algorithm, while the latter is comprised of strategies adapted specifically for the framework. The most important features are described in this section.

Island model. The island model is a tool to prevent premature convergence of the algorithm (Pandey, Chaudhary, & Mehrotra, 2014; Whitley, Rana, & Heckendorn, 1999). It is based on a facet of Darwinian evolution, in which isolated populations evolve in different ways. Thus, this model involves the evolution of parallel populations, which exchange elite individuals after a set number of iterations. This procedure also improves individual variability (Andrade et al., 2021).

Reset operator. The application of a full population reset may be demanded if the algorithm cannot escape a local optimum for a high number of generations (Toso & Resende, 2015; Whitley et al., 1999). This procedure re-initializes all chromosomes with randomly generated genes, destroying the benefits of convergence but potentially encountering new locations in the solution space. As this operator increases population diversity, it also prevents premature convergence (Pandey et al., 2014).

Shake operator. The shaking operator introduced in Andrade, Silva, and Pessoa (2019) aims to increase the diversity of the population while partially preserving some of the genes obtained in the evolutionary phase. This is done by partially re-initiating the population, applying random modifications to the elite individuals and re-initializing non-elite chromosomes with random keys. Thus, the structure of the elite chromosomes is partially preserved, and the diversity of the non-elite set is guaranteed.

Online parameter tuning. The Adaptive BRKGA (Chaves et al., 2018) introduces online parameter tuning on the BRKGA framework. In this approach, the parameters relative to crossover probability, population size, elite and mutant proportions, and maximum number of generations are updated at each iteration. This is

done to increase exploration in earlier generations while increasing exploitation in the later ones. The parameters are modified in one direction, i.e., the trajectory of the parameters is either increasing or decreasing over time. The same parameters are modified in the BRKGA-QL framework (Schenekemberg et al., 2022). In this approach, the Q-Leaning reinforced learning algorithm is used inside the BRKGA framework with possible increases and decreases. The chance of diversification decays exponentially during evolution in this approach.

Multi-parent crossover. In the original BRKGA framework (Gonçalves & Resende, 2011a), the crossover operator uses two randomly chosen individuals as parents, one from the elite set and another from the non-elite set. The mating of the two parents is biased in favor of the genes belonging to the elite parent. Andrade et al. (2021) introduces the multi-parent crossover. The algorithm uses π_e elite parents and $\pi_t - \pi_e$ parents from the non-elite set, and the genes are selected with a fitness ranking-based bias function (Bresina, 1996).

Implicit path-relinking. The Implicit Path-Relinking procedure (IPR) is also introduced in Andrade et al. (2021). As an intensification strategy, the classical path-relinking explores the neighborhood obtained in the path between two distinct solutions (Glover, 1997; Ribeiro & Resende, 2012). The use of path-relinking is typically problem-dependent, but the implicit variant creates the procedure inside of the existing BRKGA solution space. The proposed method is applied alongside the island model so that base and guide solutions come from different populations. Those individuals may belong to either the elite set or be randomly selected among all chromosomes. Andrade et al. (2021) introduces two types of this procedure, the permutation- and indicator-based IPR, whose application depends on the type of decoder.

Multi-objective evolution. Fontes and Homayouni (2023), Fontes et al. (2023) introduce the multi-population BRKGA (mp-BRKGA) for multi-objective combinatorial problems. In this approach, the Ω populations focuses on only one of the problem objectives, while the Π populations concentrate on all existing objectives. The elite set of each of the $\pi \in \Pi$ populations are stored in a pool of solutions, comprised by the best current chromosome of π and of all Ω populations. After a set amount of generations, solutions from all pools are mixed and evaluated.

Application programming interfaces. Several papers introduce application programming interfaces for BRKGA. Toso and Resende (2015) presents a C++ application that handles most of the evolutionary processes. The user only needs to define the decoding function. This implementation may use the OpenMPI API to parallelize the decoding phase into multiple threads and can be downloaded at http://github.com/rfrancotoso/brkgaAPI. This API was used to develop a GNU-style dynamic shared Python/C++ library of BRKGA. This library was introduced in Silva, Resende, and Pardalos (2015), where an extensive example of its utilization is detailed, alongside download and user instructions. Andrade et al. (2021) presents the BRKGA-MP-IPR framework and corresponding API. This API, available in the C++, Python, and Julia languages, implements the multi-parent and IPR strategies discussed in this section, alongside the multi-population strategy. Oliveira, Carravilla, Oliveira, and Resende (2021) presents an API similar to the approach used in Oliveira et al. (2019). This implementation co-evolves one population of scenarios and another of solutions for a two-stage stochastic problem. Finally, Oliveira et al. (2022) introduces an API for

scenario generation on two-stage problems, with three possible strategies for scenario evaluation.

One can note that the introduced features mainly focus on increasing population diversity, especially among elite solutions. Genetic algorithms tend to suffer from premature convergence, i.e., convergence to a sub-optimal solution (Pandey et al., 2014). In BRKGA, this is related to a lack of diversity in the elite set (Damm et al., 2016).

6. Possible under-performing issues

There are some possible challenges for this method in terms of applications, which are the focus of the fourth RQ. Those tend to be derived from BRKGA's characteristics, such as its ability to extract and optimize hidden structures in problems. This is possible due to the double elitist mechanism carried out during the evolutionary process. Notably, the standard BRKGA biased uniform crossover, when set correctly, can keep blocks of "good" genes (from the elite parents), possibly carrying optimal substructures from the problem. This mechanism is ideal for real-world problems that usually carry many hidden structures. However, vanilla BRKGA may lose performance on some classical problems, e.g., the simplest versions of set covering and routing problems, such as the Traveling Salesman Problem (TSP).

This situation was previously noted in Resende et al. (2012) which applied BRKGA to solve the Steiner triple covering problem. Since the solution landscape is relatively flat for this problem, the authors reduced the crossover bias (ρ) and increased the mutant ratio (p_m/p) to inject more variability into the population. Though this led to long computational times, it enabled the BRKGA to explore a larger portion of the solution space than would be possible with a vanilla BRKGA using the usual parameter settings. We can observe a similar effect in Andrade, Toso, Resende, and Miyazawa (2015), where the winner determination problem in combinatorial auctions was modeled as a multidimensional knapsack problem, a relatively traditional problem with a flat response surface. While the BRKGA still produced better results than the other algorithms, the difference was not very large. This was also observed on the 2D and 3D bin packing problem (Gonçalves & Resende, 2013) where the authors modified the fitness function to include not only the number of bins but also the remaining available space in the bins thus unflattening the landscape. Lastly, we can note that the BRKGA does not perform well on the traditional Traveling Salesman Problem. Such a fact is due to the uniform crossover, which is inadequate for the problem. Indeed, several papers have developed special-purpose crossover methods such as Al-Omeer and Ahmed (2019) and Ahmed (2020) for the TSP, showing that such methods outperform standard ones.

Also, the double elitism comes with a price: premature convergence. This situation may benefit scenarios where we need to quickly respond to the user, such as real-time planning systems. However, premature convergence may lead to sub-optimal solutions when we can afford additional optimization time. Indeed, the performance of BRKGA is greatly affected by its auxiliary perturbation procedures, such as reset and shaking. Although many papers do not emphasize the number of calls to methods, such as reset and shaking, this number of calls is frequent, and as Section 5 shows, novel features that affect population diversity are frequent in the literature. In Andrade, Silva, and Pessoa (2019), we can see the effect of the shaking procedure on the solution quality compared to a vanilla BRKGA.

Therefore, properly tuning the parameters that control the population diversity as well as the thresholds to call these perturbation auxiliary methods is essential. The framework proposed here shows some important counters, such as the number of iterations between solution improvements, large offsets, and others that can be used to check the convergence.

7. Possible directions for future research

The development of BRKGA since its inception has encompassed several modifications and additions to the framework. It also sheds light on diverse strengths of the metaheuristic and details methods to increase performance and versatility. The literature survey also points out to several possible research avenues for BRKGA. This section details those worthwhile areas for future research efforts, which are also the focus of the fourth research question.

An area where those efforts can thrive is the development of features and generalizations of the framework for multi-objective problems and problems with uncertainty. The mp-BRKGA framework (Fontes & Homayouni, 2023; Fontes et al., 2023) is a potential starting point for multi-objective approaches, as it is shown to have success in the coverage of the whole Pareto frontier of the instances. However, while the mp-BRKGA framework was developed for multi-objective applications, both papers in which it is used study bi-objective problems. In fact, the use of BRKGA for problems with more than two objectives is scarce, and points out to a possible research avenue. Meanwhile, for problems with uncertainty, solely two-stage problems were studied. In both cases, BRKGA was used to generate scenarios for the second-stage, either in parallel to the first-stage solution (Oliveira et al., 2019) or using the first-stage solution to compute scenario quality (Oliveira et al., 2022). Problems with uncertainty have several real-life applications, and the development of BRKGA-based frameworks to solve them would be noteworthy additions to the current literature.

Another possible research avenue is the application of hybridization, specially with matheuristics, machine learning (ML) methods, and other algorithms. As indicated in Section 4, presently there is no unified framework for the application of matheuristics in BRKGA. A possible reason is the fact that the most common use of mathematical models is inside the decoding process, which must be customized for each application. The use of exact constraints as part of the genetic operators (reproduction, crossover, mutation) has not been yet applied to BRKGA either. Meanwhile, Karimi-Mamaghan et al. (2022) defines eight possible uses of ML in metaheuristics: algorithm selection, fitness evaluation, initialization, operator selection, learnable evolution model, neighbor generation, parameter setting, and cooperation. The authors observe that those methods have already been used in, at least, one GA. However, currently only online parameter tuning (Chaves et al., 2018; Schenekemberg et al., 2022) and operator selection (Xie et al., 2022) were explored with BRKGA. Hybrids with machine learning methods are, thus, a promising area for study. Lastly, the hybridization of BRKGA and other single/multi-solution metaheuristic frameworks is shown to improve solution quality and convergence, e.g. the use of VND as local search or NSGA-II ranking methods in multi-objective approaches. This practice can be extended to other algorithms in the literature, such as GRASP, ILS, PSO, and others.

The exploitation of novel generalizations and hybrids with BRKGA implies in a demand for new APIs. The current APIs must be updated to reflect the new hybridization possibilities and state-of-the-art features, while others must be developed in popular programming languages. As Section 5 points out, the most recent complete API for BRKGA is the BRKGA-MP-IPR framework presented by Andrade et al. (2021) in C++, Julia, and Python. However, only the C++ version is updated to reflect recent innovations. The Julia API has an older but functional version of the BRKGA-MP-IPR framework, while the Python version is incomplete. One possible future work is the further development of the Python API to observe the additions to the framework. Possibly a Python-based library could be developed for BRKGA. One can find implementations of BRKGA in Java, Rust, F#, Matlab, and R, but those are not stable versions and/or do not consider present-day inclusions to the framework. The definition of a committee to oversee API development for BRKGA is a possibility for the future of this metaheuristic.

The previous topics imply in novel applications for the framework. Section 3 indicates that BRKGA versatility is a key feature to its performance, something that facilitates its use to diverse problems. One should note that novel applications can lead to new research avenues and novel hybridizations that, in turn, result in generalizations and new APIs. Thus, the development of BRKGA has the potential to turn into a virtuous cycle, where the algorithm evolves alongside the new research areas.

8. Conclusion

This paper surveyed over 150 articles on the metaheuristic Biased Random-Key Genetic Algorithm (BRKGA). The first applications of this methodology were in the early 2000s (Ericsson et al., 2002; Gonçalves & Almeida, 2002), but the framework was only formally defined in 2011 (Gonçalves & Resende, 2011a). This metaheuristic has been extensively used in several optimization problems since its conception, proving to be reliable, efficient, and versatile.

This paper was guided by four research questions (RQ). The first RQ, *Which are the areas of application of BRKGA and how did they evolve?*, was answered with the discussion of the 34 categories of problems that were studied with BRKGA since its first application. The prevalence of scheduling, network configuration, and location problems is of note, as those were some of the first studied with this metaheuristic (Buriol et al., 2010; Ericsson et al., 2002; Gonçalves & Almeida, 2002). Those are also problems in which BRKGA frequently performs the same or better with respect to solution quality than other approaches.

The second research question, *Which are the types of hybridization involving BRKGA and how did they evolve?*, was the focus of the second part of this survey, which indicates that hybrid BRKGA strategies are common, especially alongside local search heuristics. The use of hybridization is frequently concerned with increases in computational effort, as BRKGA is already a computational-heavy heuristic. The third part of this review was guided by the third RQ, *How was the BRKGA framework modified since its inception?*, and points out that the new features added to the framework usually focus on increasing population diversity to prevent premature convergence.

The focus of the fourth RQ, *What are the opportunities for the growth of the BRKGA framework?*, are the possible challenges and growth opportunities for this method. We observe that populational diversity and perturbation procedures are essential to increase performance and that there are opportunities in the areas of multi-objective problems, problems with uncertainty, hybridizations, and the development of APIs.

To sum up, this study presents an extensive overview of the current corpus of literature regarding BRKGA that may be useful for any researcher aiming to study this metaheuristic. Possible avenues to broaden the scope of this study are the addition of other databases, such as Web of Science and arXiv, and the inclusion of thesis, dissertations, conference papers, and non-peer-reviewed material such as technical reports and preprints.

Appendix A. Supplementary material
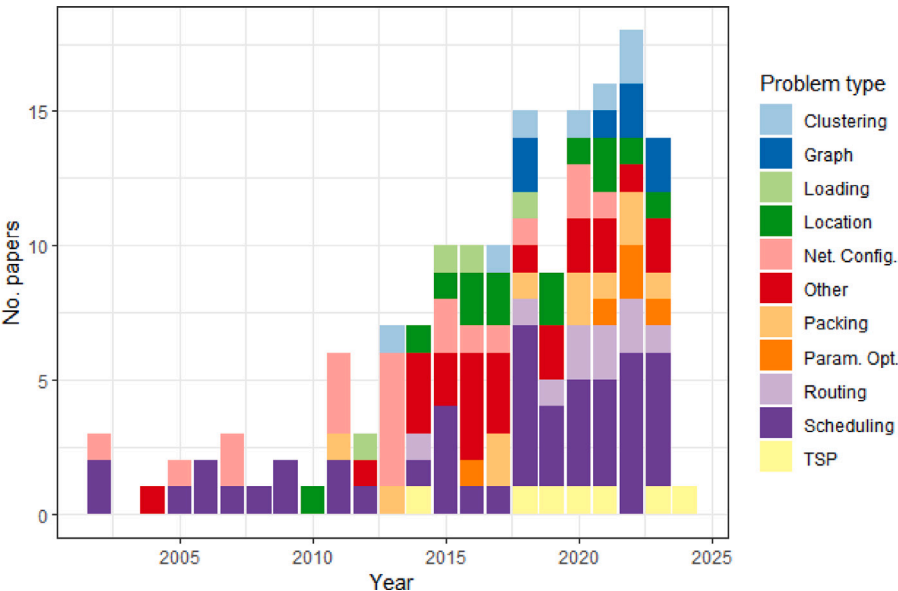
See Figs. A.1–A.4.

Fig. A.1. Number of papers per year and problem. Problem type "other" refers to problems studied in two or less articles.
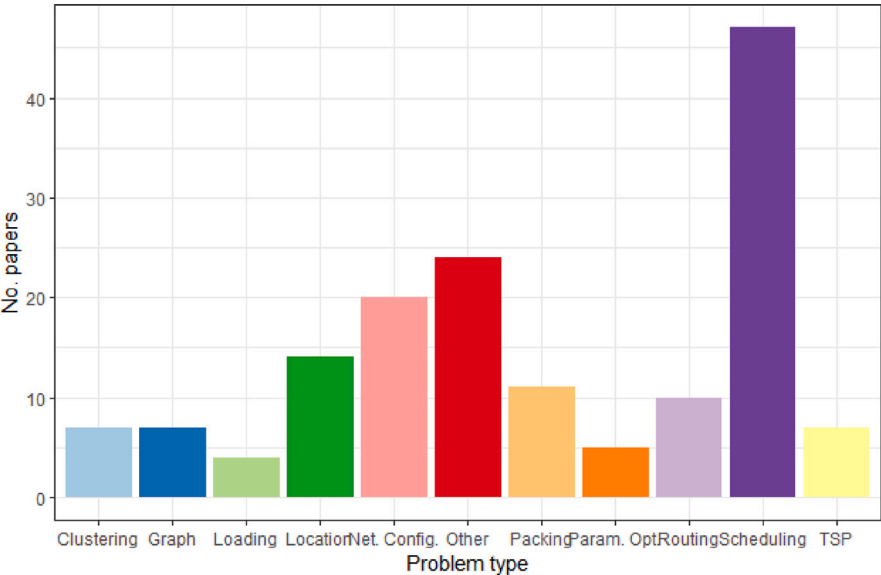

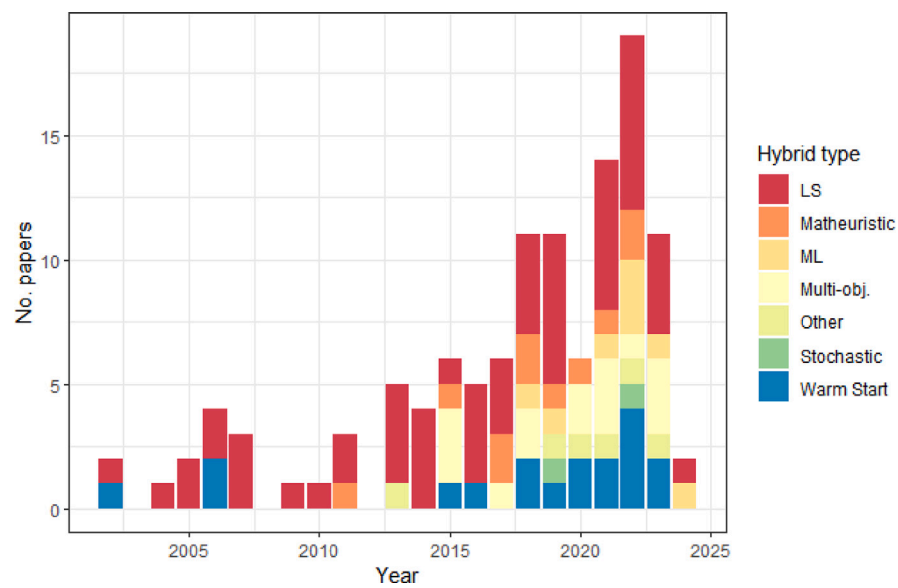**Fig. A.2.** Number of papers per problem. Problem type "other" refers to problems studied in two or less articles.

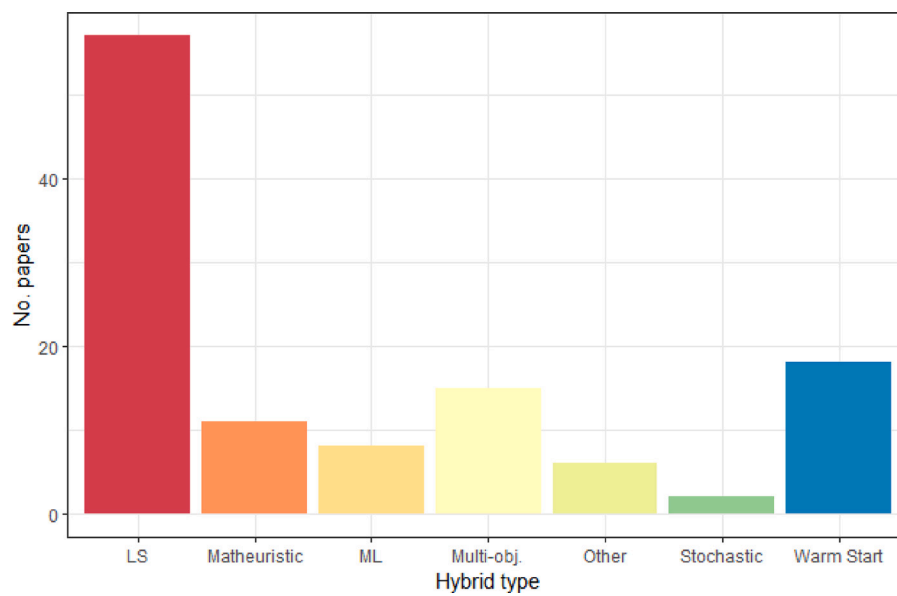**Fig. A.3.** Number of papers per year and hybrid.

Fig. A.4. Number of papers per hybrid.

References

Abreu, L. R., Athayde Prata, B., Gomes, A. C., Braga-Santos, S. A., & Nagano, M. S. (2022). A novel BRKGA for the customer order scheduling with missing operations to minimize total tardiness. *Swarm and Evolutionary Computation*, *75*, Article 101149. http://dx.doi.org/10.1016/j.swevo.2022.101149.

Abreu, L. R., Tavares-Neto, R. F., & Nagano, M. S. (2021). A new efficient biased random-key genetic algorithm for open shop scheduling with routing by capacitated single vehicle and makespan minimization. *Engineering Applications of Artificial Intelligence*, *104*, Article 104373. http://dx.doi.org/10.1016/j.engappai.2021.104373.

Ahmed, Z. H. (2020). A comparative study of eight crossover operators for the maximum scatter travelling salesman problem. *International Journal of Advanced Computer Science and Applications*, *11*, URL https://api.semanticscholar.org/CorpusID:226735188.

Akers, S. B. (1956). A graphical approach to production scheduling problems. *Operations Research*, *4*(2), 244–245. http://dx.doi.org/10.1287/opre.4.2.244.

Al-Omeer, M. A., & Ahmed, Z. H. (2019). Comparative study of crossover operators for the MTSP. In *2019 international conference on computer and information sciences* (pp. 1–6). http://dx.doi.org/10.1109/ICCISci.2019.8716483.

Almeida, B. F., Correia, I., & Saldanha-da Gama, F. (2018). A biased random-key genetic algorithm for the project scheduling problem with flexible resources. *TOP*, *26*(2), 283–308. http://dx.doi.org/10.1007/s11750-018-0472-9, [ISSN: 11345764 (ISSN)].

Amaro Júnior, B., Carvalho, G. N., Santos, M. C., Pinheiro, P. R., & Celedonio, J. W. L. (2023). Evolutionary algorithms for optimization sequence of cut in the laser cutting path problem. *Applied Sciences*, *13*(18), 10133. http://dx.doi.org/10.3390/app131810133.

Amaro Júnior, B., Pinheiro, P. R., & Coelho, P. V. (2017). A parallel biased random-key genetic algorithm with multiple populations applied to irregular strip packing problems. *Mathematical Problems in Engineering*, *2017*, Article 1670709. http://dx.doi.org/10.1155/2017/1670709, [ISSN: 1024123X (ISSN)].

Amaro Júnior, B., Santos, M. C., Carvalho, G. N., Araújo, L. J. P., & Pinheiro, P. R. (2021). Metaheuristics for the minimum time cut path problem with different cutting and sliding speeds. *Algorithms*, [ISSN: 19994893] *14*(11), 305. http://dx.doi.org/10.3390/a14110305.

Andrade, C. E., Ahmed, S., Nemhauser, G. L., & Shao, Y. (2017). A hybrid primal heuristic for finding feasible solutions to mixed integer programs. *European Journal of Operational Research*, *263*(1), 62–71. http://dx.doi.org/10.1016/j.ejor.2017.05.003, [ISSN: 03772217 (ISSN)].

Andrade, C. E., Byers, S. D., Gopalakrishnan, V., Halepovic, E., Majmundar, M., Poole, D. J., et al. (2017). Managing massive firmware-over-the-air updates for connected cars in cellular networks. In *Proceedings of the 2nd ACM international workshop on smart, autonomous, and connected vehicular systems and services* (pp. 65–72). ACM, ISBN: 978-1-4503-5146-1, http://dx.doi.org/10.1145/3131944.3131953.

Andrade, C. E., Byers, S. D., Gopalakrishnan, V., Halepovic, E., Poole, D. J., Tran, L. K., et al. (2019). Scheduling software updates for connected cars with limited availability. *Applied Soft Computing*, *82*, Article 105575. http://dx.doi.org/10.1016/j.asoc.2019.105575, [ISSN: 15684946 (ISSN)].

Andrade, C. E., Miyazawa, F. K., & Resende, M. G. C. (2013). Evolutionary algorithm for the *k*-interconnected multi-depot multi-traveling salesmen problem. In *Proceedings of the 15th annual conference on genetic and evolutionary computation* (pp. 463–470). New York, NY, USA: ACM, ISBN: 978-1-4503-1963-8, http://dx.doi.org/10.1145/2463372.2463434.

Andrade, C. E., Pessoa, L. S., & Stawiarski, S. (2022). The physical cell identity assignment problem: a practical optimization approach. *IEEE Transactions on Evolutionary Computation*, [ISSN: 1089-778X] 1. http://dx.doi.org/10.1109/TEVC.2022.3185927, To appear.

Andrade, C. E., Resende, M. G. C., Karloff, H. J., & Miyazawa, F. K. (2014). Evolutionary algorithms for overlapping correlation clustering. In *Proceedings of the 16th conference on genetic and evolutionary computation* (pp. 405–412). New York, NY, USA: ACM, ISBN: 978-1-4503-2662-9, http://dx.doi.org/10.1145/2576768.2598284.

Andrade, C. E., Resende, M. G. C., Zhang, W., Sinha, R. K., Reichmann, K. C., Doverspike, R. D., et al. (2015). A biased random-key genetic algorithm for wireless backhaul network design. *Applied Soft Computing*, *33*, 150–169. http://dx.doi.org/10.1016/j.asoc.2015.04.016, [ISSN: 15684946 (ISSN)].

Andrade, C. E., Silva, T., & Pessoa, L. S. (2019). Minimizing flowtime in a flowshop scheduling problem with a biased random-key genetic algorithm. *Expert Systems with Applications*, *128*, 67–80. http://dx.doi.org/10.1016/j.eswa.2019.03.007, [ISSN: 09574174 (ISSN)].

Andrade, C. E., Toso, R. F., Gonçalves, J. F., & Resende, M. G. C. (2021). The multiparent biased random-key genetic algorithm with implicit path-relinking and its real-world applications. *European Journal of Operational Research*, *289*(1), 17–30. http://dx.doi.org/10.1016/j.ejor.2019.11.037, [ISSN: 03772217 (ISSN)].

Andrade, C. E., Toso, R. F., Resende, M. G. C., & Miyazawa, F. K. (2015). Biased random-key genetic algorithms for the winner determination problem in combinatorial auctions. *Evolutionary Computation*, *23*(2), 279–307. http://dx.doi.org/10.1162/EVCO_a_00138, [ISSN: 10636560 (ISSN)].

Araújo, F. F. B., Costa, A. M., & Miralles, C. (2015). Balancing parallel assembly lines with disabled workers. *European Journal of Industrial Engineering*, *9*(3), 344–365. http://dx.doi.org/10.1504/EJIE.2015.069343, [ISSN: 17515254 (ISSN)].

Baas, J., Schotten, M., Plume, A., Côté, G., & Karimi, R. (2020). Scopus as a curated, high-quality bibliometric data source for academic research in quantitative science studies. *Quantitative Science Studies*, *1*(1), 377–386. http://dx.doi.org/10.1162/qss_a_00019.

Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal on Computing*, *6*(2), 154–160. http://dx.doi.org/10.1287/ijoc.6.2.154.

Beirão, N. C. L. F. (1997). *Sistema de apoio à decisão para sequenciamento de operações em ambientes Job Shop* (Ph.D. thesis), University of Porto, URL https://repositorio-aberto.up.pt/handle/10216/12242.

Bellman, R. (1966). Dynamic programming. *Science*, *153*(3731), 34–37. http://dx.doi.org/10.1126/science.153.3731.34.

Bernardino, R., & Paias, A. (2018). Metaheuristics based on decision hierarchies for the traveling purchaser problem. *International Transactions in Operational Research*, *25*(4), 1269–1295. http://dx.doi.org/10.1111/itor.12330, [ISSN 09696016 (ISSN)].

Beume, N., Naujoks, B., & Emmerich, M. (2007). SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, *181*(3), 1653–1669. http://dx.doi.org/10.1016/j.ejor.2006.08.008.

Biajoli, F. L., Chaves, A. A., & Lorena, L. A. N. (2019). A biased random-key genetic algorithm for the two-stage capacitated facility location problem. *Expert Systems with Applications*, *115*, 418–426. http://dx.doi.org/10.1016/j.eswa.2018.08.024, [ISSN: 09574174 (ISSN)].

Bolsi, B., Lima, V. L., Alves Queiroz, T. A., & Iori, M. (2022). Heuristic algorithms for integrated workforce allocation and scheduling of perishable products. *International Journal of Production Research*, 1–16. http://dx.doi.org/10.1080/00207543.2022.2144525.

Boschetti, M. A., Maniezzo, V., Roffilli, M., & Röhler, A. B. (2009). Matheuristics: Optimization, simulation and control. In *International workshop on hybrid metaheuristics* (pp. 171–177). Springer, http://dx.doi.org/10.1007/978-3-642-04918-7_13.

Brandão, J. S., Noronha, T. F., Resende, M. G. C., & Ribeiro, C. C. (2015). A biased random-key genetic algorithm for single-round divisible load scheduling. *International Transactions in Operational Research*, *22*(5), 823–839. http://dx.doi.org/10.1111/itor.12178, [ISSN: 09696016 (ISSN)].

Brandão, J. S., Noronha, T. F., Resende, M. G. C., & Ribeiro, C. C. (2017). A biased random-key genetic algorithm for scheduling heterogeneous multi-round systems. *International Transactions in Operational Research*, *24*(5), 1061–1077. http://dx.doi.org/10.1111/itor.12429, [ISSN: 09696016 (ISSN)].

Brandão, J. S., Noronha, T. F., & Ribeiro, C. C. (2016). A biased random-key genetic algorithm to maximize the number of accepted lightpaths in WDM optical networks. *Journal of Global Optimization*, *65*(4), 813–835. http://dx.doi.org/10.1007/s10898-015-0389-x, [ISSN: 09255001 (ISSN)].

Bresina, J. L. (1996). Heuristic-biased stochastic sampling. In *Proceedings of the thirteenth national conference on artificial intelligence-volume 1* (pp. 271–278).

Brito, J., Fadel, A., & Semaan, G. (2020). A genetic algorithm applied to optimal allocation in stratified sampling. *Communications in Statistics. Simulation and Computation*, 3714–3732. http://dx.doi.org/10.1080/03610918.2020.1722832, [ISSN: 03610918 (ISSN)].

Brito, J. A., Semaan, G., & Fadel, A. (2022). The effective BRKGA algorithm for the k-medoids clustering problem. *RAIRO-Operations Research*, *56*(4), 3137–3153. http://dx.doi.org/10.1051/ro/2022141.

Buriol, L. S., Hirsch, M. J., Pardalos, P. M., Querido, T., Resende, M. G. C., & Ritt, M. (2010). A biased random-key genetic algorithm for road congestion minimization. *Optimization Letters*, *4*(4), 619–633. http://dx.doi.org/10.1007/s11590-010-0226-6, [ISSN: 18624472 (ISSN)].

Buriol, L. S., Resende, M. G. C., Ribeiro, C. C., & Thorup, M. (2005). A hybrid genetic algorithm for the weight setting problem in OSPF/IS-IS routing. *Networks*, *46*(1), 36–56. http://dx.doi.org/10.1002/net.20070.

Buriol, L. S., Resende, M. G. C., & Thorup, M. (2007). Survivable IP network design with OSPF routing. *Networks: An International Journal*, *49*(1), 51–64. http://dx.doi.org/10.1002/net.20141.

Cabo, M., González-Velarde, J. L., Possani, E., & Ríos Solís, Y. Á. (2018). Bi-objective scheduling on a restricted batching machine. *Computers & Operations Research*, *100*, 201–210. http://dx.doi.org/10.1016/j.cor.2018.07.004, [ISSN: 03050548 (ISSN)].

Carrabs, F. (2021). A biased random-key genetic algorithm for the set orienteering problem. *European Journal of Operational Research*, *292*(3), 830–854. http://dx.doi.org/10.1016/j.ejor.2020.11.043, [ISSN: 03772217 (ISSN)].

Caserta, M., & Reiners, T. (2016). A pool-based pattern generation algorithm for logical analysis of data with automatic fine-tuning. *European Journal of Operational Research*, *248*(2), 593–606. http://dx.doi.org/10.1016/j.ejor.2015.05.078, [ISSN: 03772217 (ISSN)].

Cavalheiro, E. M. B., Vergílio, A. H. B., & Lyra, C. (2018). Optimal configuration of power distribution networks with variable renewable energy resources. *Computers & Operations Research*, *96*, 272–280. http://dx.doi.org/10.1016/j.cor.2017.09.021, [ISSN: 03050548 (ISSN)].

Chagas, J. B. C., Blank, J., Wagner, M., Souza, M. J. F., & Deb, K. (2021). A non-dominated sorting based customized random-key genetic algorithm for the bi-objective traveling thief problem. *Journal of Heuristics*, *27*(3), 267–301. http://dx.doi.org/10.1007/s10732-020-09457-7, [ISSN: 13811231 (ISSN)].

Chagas, J. B. C., Toffolo, T. A. M., Souza, M. J. F., & Iori, M. (2020). The double traveling salesman problem with partial last-in-first-out loading constraints. *International Transactions in Operational Research*, *29*(4), 2346–2373. http://dx.doi.org/10.1111/itor.12876, [ISSN: 09696016 (ISSN)].

Chan, F. T. S., Tibrewal, R. K., Prakash, A., & Tiwari, M. K. (2015). A biased random key genetic algorithm approach for inventory-based multi-item lot-sizing problem. *Proceedings of the Institution of Mechanical Engineers, Part B (Management and Engineering Manufacture)*, *229*(1), 157–171. http://dx.doi.org/10.1177/0954405414523594, [ISSN: 09544054 (ISSN)].

Chaves, A. A., Gonçalves, J. F., & Lorena, L. A. N. (2018). Adaptive biased random-key genetic algorithm with local search for the capacitated centered clustering problem. *Computers & Industrial Engineering*, *124*, 331–346. http://dx.doi.org/10.1016/j.cie.2018.07.031, [ISSN: 03608352 (ISSN)].

Chaves, A. A., Lorena, L. A. N., Senne, E. L. F., & Resende, M. G. C. (2016). Hybrid method with CS and BRKGA applied to the minimization of tool switches problem. *Computers & Operations Research*, *67*, 174–183. http://dx.doi.org/10.1016/j.cor.2015.10.009, [ISSN: 03050548 (ISSN)].

Chaves, A. A., Vianna, B. L., da Silva, T. T., & Schenekemberg, C. M. (2024). A parallel branch-and-cut and an adaptive metaheuristic to solve the Family Traveling Salesman Problem. *Expert Systems with Applications*, *238*, Article 121735. http://dx.doi.org/10.1016/j.eswa.2023.121735.

Chou, F. D., & Wang, H. M. (2008). Scheduling for a single semiconductor batch-processing machine to minimize total weighted tardiness. *Journal of the Chinese Institute of Industrial Engineers*, *25*(2), 136–147. http://dx.doi.org/10.1080/10170660809509079.

Cicek, Z. I. E., & Ozturk, Z. K. (2021). Optimizing the artificial neural network parameters using a biased random key genetic algorithm for time series forecasting. *Applied Soft Computing*, *102*, Article 107091. http://dx.doi.org/10.1016/j.asoc.2021.107091, [ISSN: 15684946 (ISSN)].

Cooper, H. (2015). *Vol. 2, Research synthesis and meta-analysis: A step-by-step approach.* Sage Publications.

Correcher, J. F., & Alvarez-Valdes, R. (2017). A biased random-Key genetic algorithm for the time-invariant berth allocation and quay crane assignment problem. *Expert Systems with Applications*, *89*, 112–128. http://dx.doi.org/10.1016/j.eswa.2017.07.028, [ISSN: 09574174 (ISSN)].

Cotta, C., Mathieson, L., & Moscato, P. (2018). Memetic algorithms. In R. Martí, P. M. Pardalos, & M. G. C. Resende (Eds.), *Handbook of heuristics* (pp. 607–638). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-07124-4_29.

Cunha, V., Pessoa, L., Vellasco, M., Tanscheit, R., & Pacheco, M. A. (2018). A biased random-key genetic algorithm for the rescue unit allocation and scheduling problem. In *2018 IEEE congress on evolutionary computation* (pp. 1–6). http://dx.doi.org/10.1109/CEC.2018.8477819.

Damm, R. B., Resende, M. G. C., & Ronconi, D. P. (2016). A biased random key genetic algorithm for the field technician scheduling problem. *Computers & Operations Research*, *75*, 49–63. http://dx.doi.org/10.1016/j.cor.2016.05.003, [ISSN: 03050548 (ISSN)].

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. http://dx.doi.org/10.1109/4235.996017.

Duarte, A., Martí, R., Resende, M. G. C., & Silva, R. M. A. (2014). Improved heuristics for the regenerator location problem. *International Transactions in Operational Research*, *21*(4), 541–558. http://dx.doi.org/10.1111/itor.12085, [ISSN: 09696016 (ISSN)].

Duarte, A., Sánchez-Oro, J., Mladenović, N., & Todosijević, R. (2018). Variable neighborhood descent. In R. Martí, P. M. Pardalos, & M. G. C. Resende (Eds.), *Handbook of heuristics* (pp. 341–367). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-07124-4_9.

Elsevier (2022). Scopus. https://www.scopus.com. (Accessed 24 August 2022).

Ericsson, M., Resende, M. G. C., & Pardalos, P. M. (2002). A genetic algorithm for the weight setting problem in OSPF routing. *Journal of Combinatorial Optimization*, [ISSN: 1382-6905] *6*(3), 299–333. http://dx.doi.org/10.1023/A:1014862026591.

Fadel, A. C., Ochi, L. S., Brito, J. A. D. M., & Semaan, G. S. (2021). Microaggregation heuristic applied to statistical disclosure control. *Information Sciences*, *548*, 37–55. http://dx.doi.org/10.1016/j.ins.2020.09.069, [ISSN: 00200255 (ISSN)].

Falls, M., Bernardello, R., Castrillo, M., Acosta, M., Llort, J., & Galí, M. (2022). Use of genetic algorithms for ocean model parameter optimisation: a case study using PISCES-v2_RC for North Atlantic particulate organic carbon. *Geoscientific Model Development*, *15*(14), 5713–5737. http://dx.doi.org/10.5194/gmd-15-5713-2022.

Faria Jr., H., Resende, M. G. C., & Ernst, D. (2017). A biased random key genetic algorithm applied to the electric distribution network reconfiguration problem. *Journal of Heuristics*, *23*(6), 533–550. http://dx.doi.org/10.1007/s10732-017-9355-8, [ISSN: 13811231 (ISSN)].

Festa, P. (2013). A biased random-key genetic algorithm for data clustering. *Mathematical Biosciences*, *245*(1), 76–85. http://dx.doi.org/10.1016/j.mbs.2013.07.011, [ISSN: 00255564 (ISSN)].

Fischetti, M., & Fischetti, M. (2018). Matheuristics. In R. Martí, P. M. Pardalos, & M. G. C. Resende (Eds.), *Handbook of heuristics* (pp. 121–153). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-07124-4_14.

Fontes, D. B. M. M., & Gonçalves, J. F. (2007). Heuristic solutions for general concave minimum cost network flow problems. *Networks*, *50*(1), 67–76. http://dx.doi.org/10.1002/net.20167, [ISSN: 00283045 (ISSN)].

Fontes, D. B. M. M., & Gonçalves, J. F. (2013). A multi-population hybrid biased random key genetic algorithm for hop-constrained trees in nonlinear cost flow networks. *Optimization Letters*, *7*(6), 1303–1324. http://dx.doi.org/10.1007/s11590-012-0505-5, [ISSN: 18624472 (ISSN)].

Fontes, D. B. M. M., Gonçalves, J. F., & Fontes, F. A. C. C. (2018). An evolutionary approach to the maximum edge weight clique problem. *Recent Advances in Electrical and Electronic Engineering*, *11*(3), 260–266. http://dx.doi.org/10.2174/2352096511666180214105643, [ISSN: 23520965 (ISSN)].

Fontes, D. B. M. M., & Homayouni, S. M. (2023). A bi-objective multi-population biased random key genetic algorithm for joint scheduling quay cranes and speed adjustable vehicles in container terminals. *Flexible Services and Manufacturing Journal*, *35*(1), 241–268. http://dx.doi.org/10.1007/s10696-022-09467-6.

Fontes, D. B. M. M., Homayouni, S. M., & Fernandes, J. C. (2023). Energy-efficient job shop scheduling problem with transport resources considering speed adjustable resources. *International Journal of Production Research*, 1–24. http://dx.doi.org/10.1080/00207543.2023.2175172.

Freitas, C. C., Aloise, D. J., Da Costa Fontes, F. F., Santos, A. C., & Da Silva Menezes, M. (2023). A biased random-key genetic algorithm for the two-level hub location routing problem with directed tours. *OR Spectrum*, 1–22. http://dx.doi.org/10.1007/s00291-023-00718-y.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences*, *8*(1), 156–166. http://dx.doi.org/10.1111/j.1540-5915.1977.tb01074.x.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, *13*(5), 533–549. http://dx.doi.org/10.1016/0305-0548(86)90048-1.

Glover, F. (1997). Tabu search and adaptive memory programming – Advances, Applications and Challenges. In *Interfaces in computer science and operations research* (pp. 1–75). Springer, http://dx.doi.org/10.1007/978-1-4615-4102-8_1.

Gonçalves, J. F., & Almeida, J. R. (2002). A hybrid genetic algorithm for assembly line balancing. *Journal of Heuristics*, *8*(6), 629–642. http://dx.doi.org/10.1023/A:1020377910258, [ISSN: 13811231 (ISSN)].

Gonçalves, J. F., & Beirão, N. C. L. F. (1999). Um algoritmo genético baseado em chave aleatórias para sequenciamento de opearções. *Investigação Operacional, 19*, 123–137.

Gonçalves, J. F., Magalhães Mendes, J. J., & Resende, M. G. C. (2005). A hybrid genetic algorithm for the job shop scheduling problem. *European Journal of Operational Research*, *167*(1), 77–95. http://dx.doi.org/10.1016/j.ejor.2004.03.012, [ISSN: 03772217 (ISSN)].

Gonçalves, J. F., Mendes, J. J. M., & Resende, M. G. C. (2008). A genetic algorithm for the resource constrained multi-project scheduling problem. *European Journal of Operational Research*, *189*(3), 1171–1190. http://dx.doi.org/10.1016/j.ejor.2006.06.074, [ISSN: 03772217 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers & Industrial Engineering*, *47*(2–3), 247–273. http://dx.doi.org/10.1016/j.cie.2004.07.003, [ISSN: 03608352 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2011a). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, *17*(5), 487–525. http://dx.doi.org/10.1007/s10732-010-9143-1, [ISSN: 13811231 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2011b). A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem. *Journal of Combinatorial Optimization*, *22*(2), 180–201. http://dx.doi.org/10.1007/s10878-009-9282-1, [ISSN: 13826905 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2012). A parallel multi-population biased random-key genetic algorithm for a container loading problem. *Computers & Operations Research*, *39*(2), 179–190. http://dx.doi.org/10.1016/j.cor.2011.03.009, [ISSN: 03050548 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2013). A biased random key genetic algorithm for 2D and 3D bin packing problems. *International Journal of Production Economics*, *145*(2), 500–510. http://dx.doi.org/10.1016/j.ijpe.2013.04.019, [ISSN: 09255273 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2014). An extended Akers graphical method with a biased random-key genetic algorithm for job-shop scheduling. *International Transactions in Operational Research*, *21*(2), 215–246. http://dx.doi.org/10.1111/itor.12044, [ISSN: 09696016 (ISSN)].

Gonçalves, J. F., & Resende, M. G. C. (2015). A biased random-key genetic algorithm for the unequal area facility layout problem. *European Journal of Operational Research*, *246*(1), 86–107. http://dx.doi.org/10.1016/j.ejor.2015.04.029, [ISSN: 03772217 (ISSN)].

Gonçalves, J. F., Resende, M. G. C., & Costa, M. D. (2016). A biased random-key genetic algorithm for the minimization of open stacks problem. *International Transactions in Operational Research*, *23*(1–2), 25–46. http://dx.doi.org/10.1111/itor.12109, [ISSN: 09696016 (ISSN)].

Gonçalves, J. F., Resende, M. G. C., & Mendes, J. J. M. (2011). A biased random-key genetic algorithm with forward-backward improvement for the resource constrained project scheduling problem. *Journal of Heuristics*, *17*(5), 467–486. http://dx.doi.org/10.1007/s10732-010-9142-2, [ISSN: 13811231 (ISSN)].

Gonçalves, J. F., & Sousa, P. S. A. (2011). A genetic algorithm for lot sizing and scheduling under capacity constraints and allowing backorders. *International Journal of Production Research*, *49*(9), 2683–2703. http://dx.doi.org/10.1080/00207543.2010.532936, [ISSN: 00207543 (ISSN)].

Gonçalves, J. F., & Wäscher, G. (2020). A MIP model and a biased random-key genetic algorithm based approach for a two-dimensional cutting problem with defects. *European Journal of Operational Research*, *286*(3), 867–882. http://dx.doi.org/10.1016/j.ejor.2020.04.028, [ISSN: 03772217 (ISSN)].

Grasas, A., Ramalhinho, H., Pessoa, L. S., Resende, M. G., Caballé, I., & Barba, N. (2014). On the improvement of blood sample collection at clinical laboratories. *BMC Health Services Research*, *14*, 12. http://dx.doi.org/10.1186/1472-6963-14-12, [ISSN: 14726963 (ISSN)].

Hansen, P., & Mladenović, N. (2005). Variable neighborhood search. In *Search methodologies* (pp. 211–238). Springer, http://dx.doi.org/10.1007/0-387-28356-0_8.

He, L., Guijt, A., Weerdt, M., Xing, L., & Yorke-Smith, N. (2019). Order acceptance and scheduling with sequence-dependent setup times: A new memetic algorithm and benchmark of the state of the art. *Computers & Industrial Engineering*, *138*, Article 106102. http://dx.doi.org/10.1016/j.cie.2019.106102, [ISSN: 03608352 (ISSN)].

Heilig, L., Lalla-Ruiz, E., & Voß, S. (2016). A cloud brokerage approach for solving the resource management problem in multi-cloud environments. *Computers & Industrial Engineering*, *95*, 16–26. http://dx.doi.org/10.1016/j.cie.2016.02.015, [ISSN: 03608352 (ISSN)].

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press.

Homayouni, S. M., Fontes, D. B. M. M., & Gonçalves, J. F. (2020). A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation. *International Transactions in Operational Research*, 688–716. http://dx.doi.org/10.1111/itor.12878, [ISSN: 09696016 (ISSN)].

Hottung, A., & Tierney, K. (2016). A biased random-key genetic algorithm for the container pre-marshalling problem. *Computers & Operations Research*, *75*, 83–102. http://dx.doi.org/10.1016/j.cor.2016.05.011, [ISSN: 03050548 (ISSN)].

Huang, Y., Santos, A. C., & Duhamel, C. (2018). Methods for solving road network problems with disruptions. *Electronic Notes in Discrete Mathematics*, *64*, 175–184. http://dx.doi.org/10.1016/j.endm.2018.01.019, [ISSN: 15710653 (ISSN)].

Huang, Y., Santos, A. C., & Duhamel, C. (2020a). Bi-objective methods for road network problems with disruptions and connecting requirements. *Journal of the Operational Research Society*, *71*(12), 1959–1971. http://dx.doi.org/10.1080/01605682.2019.1639479, [ISSN: 01605682 (ISSN)].

Huang, Y., Santos, A. C., & Duhamel, C. (2020b). Model and methods to address urban road network problems with disruptions. *International Transactions in Operational Research*, *27*(6), 2715–2739. http://dx.doi.org/10.1111/itor.12641, [ISSN: 09696016 (ISSN)].

Ibarra-Rojas, O. J., & Silva-Soto, Y. (2021). Vehicle routing problem considering equity of demand satisfaction. *Optimization Letters*, *15*(6), 2275–2297. http://dx.doi.org/10.1007/s11590-021-01704-5, [ISSN: 18624472 (ISSN)].

Japa, L., Serqueira, M., Mendonça, I., Aritsugi, M., Bezerra, E., & González, P. H. (2023). A population-based hybrid approach for hyperparameter optimization of neural networks. *IEEE Access*, 50752. http://dx.doi.org/10.1109/ACCESS.2023.3277310.

Johnson, D. S., Breslau, L., Diakonikolas, I., Duffield, N., Gu, Y., Hajiaghayi, M., et al. (2020). Near-optimal disjoint-path facility location through set cover by pairs. *Operations Research*, *68*(3), 896–926. http://dx.doi.org/10.1287/opre.2019.1956.

Karimi-Mamaghan, M., Mohammadi, M., Meyer, P., Karimi-Mamaghan, A. M., & Talbi, E. G. (2022). Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art. *European Journal of Operational Research*, *296*(2), 393–422. http://dx.doi.org/10.1016/j.ejor.2021.04.032.

Kashan, A. H., Karimi, B., & Jolai, F. (2006). Effective hybrid genetic algorithm for minimizing makespan on a single-batch-processing machine with non-identical job sizes. *International Journal of Production Research*, *44*(12), 2337–2360. http://dx.doi.org/10.1080/00207540500525254, [ISSN: 00207543 (ISSN)].

Kong, M., Liu, X., Pei, J., Cheng, H., & Pardalos, P. M. (2020). A BRKGA-DE algorithm for parallel-batching scheduling with deterioration and learning effects on parallel machines under preventive maintenance consideration. *Annals of Mathematics and Artificial Intelligence*, *88*(1–3), 237–267. http://dx.doi.org/10.1007/s10472-018-9602-1, [ISSN: 10122443 (ISSN)].

Kong, M., Pei, J., Xu, J., Liu, X., Yu, X., & Pardalos, P. M. (2020). A robust optimization approach for integrated steel production and batch delivery scheduling with uncertain rolling times and deterioration effect. *International Journal of Production Research*, *58*(17), 5132–5154. http://dx.doi.org/10.1080/00207543.2019.1693659, [ISSN: 00207543 (ISSN)].

Kummer, A. F., Araújo, O. C. B., Buriol, L. S., & Resende, M. G. C. (2022). A biased random-key genetic algorithm for the home health care problem. *International Transactions in Operational Research*, 1–31. http://dx.doi.org/10.1111/itor.13221.

Lalla-Ruiz, E., Expósito-Izquierdo, C., Melián-Batista, B., & Moreno-Vega, J. M. (2016). A hybrid biased random key genetic algorithm for the quadratic assignment problem. *Information Processing Letters*, *116*(8), 513–520. http://dx.doi.org/10.1016/j.ipl.2016.03.002, [ISSN: 00200190 (ISSN)].

Lalla-Ruiz, E., González-Velarde, J. L., Melián-Batista, B., & Moreno-Vega, J. M. (2014). Biased random key genetic algorithm for the Tactical Berth Allocation Problem. *Applied Soft Computing*, *22*, 60–76. http://dx.doi.org/10.1016/j.asoc.2014.04.035, [ISSN: 15684946 (ISSN)].

Leonhart, P. F., Spieler, E., Ligabue-Braun, R., & Dorn, M. (2019). A biased random key genetic algorithm for the protein–ligand docking problem. *Soft Computing*, *23*(12), 4155–4176. http://dx.doi.org/10.1007/s00500-018-3065-5, [ISSN: 14327643 (ISSN)].

Li, X., & Zhang, Y. (2018). Single batch processing machine scheduling with two-dimensional bin packing constraints. *International Journal of Production Economics*, *196*, 113–121. http://dx.doi.org/10.1016/j.ijpe.2017.11.015, [ISSN: 09255273 (ISSN)].

Lim, W. M., Kumar, S., & Ali, F. (2022). Advancing knowledge through literature reviews:'what','why', and 'how to contribute'. *The Service Industries Journal*, *42*(7–8), 481–513. http://dx.doi.org/10.1080/02642069.2022.2047941.

Lima, A., Aquino, A. L. L., Nogueira, B., & Pinheiro, R. G. S. (2022). A matheuristic approach for the minimum broadcast time problem using a biased random-key genetic algorithm. *International Transactions in Operational Research*, 246–273. http://dx.doi.org/10.1111/itor.13146.

Londe, M. A., Andrade, C. E., & Pessoa, L. S. (2021). An evolutionary approach for the p-next center problem. *Expert Systems with Applications*, *175*, Article 114728. http://dx.doi.org/10.1016/j.eswa.2021.114728, [ISSN: 09574174 (ISSN)].

Londe, M. A., Andrade, C. E., & Pessoa, L. S. (2022). Exact and heuristic approaches for the root sequence index allocation problem. *Applied Soft Computing*, *130*, Article 109634. http://dx.doi.org/10.1016/j.asoc.2022.109634.

Lopes, M. C., Andrade, C. E., Queiroz, T. A., Resende, M. G. C., & Miyazawa, F. K. (2016). Heuristics for a hub location-routing problem. *Networks*, *68*(1), 54–90. http://dx.doi.org/10.1002/net.21685, [ISSN: 00283045 (ISSN)].

Lourenço, H. R., Martin, O. C. M., & Stützle, T. (2003). Iterated local search. In *Handbook of metaheuristics* (pp. 320–353). Springer, http://dx.doi.org/10.1007/0-306-48056-5_11.

Lu, Z., Hu, K., & Ng, T. S. (2023). Improving Additive Manufacturing production planning: A sub-second pixel-based packing algorithm. *Computers & Industrial Engineering*, *181*, Article 109318. http://dx.doi.org/10.1016/j.cie.2023.109318.

Maecker, S., Shen, L., & Mönch, L. (2023). Unrelated parallel machine scheduling with eligibility constraints and delivery times to minimize total weighted tardiness. *Computers & Operations Research*, *149*, Article 105999. http://dx.doi.org/10.1016/j.cor.2022.105999.

Malve, S., & Uzsoy, R. (2007). A genetic algorithm for minimizing maximum lateness on parallel identical batch processing machines with dynamic job arrivals and incompatible job families. *Computers & Operations Research*, *34*(10), 3016–3028. http://dx.doi.org/10.1016/j.cor.2005.11.011, [ISSN: 03050548 (ISSN)].

Marques Jr., E. L., Coelho, V. N., Coelho, I. M., Satoru Ochi, L., Maculan, N., Mladenović, N., et al. (2023). A two-phase multi-objective metaheuristic for a green UAV grid routing problem. *Optimization Letters*, 1–24. http://dx.doi.org/10.1007/s11590-023-02013-9.

Martarelli, N. J., & Nagano, M. S. (2020). Unsupervised feature selection based on bio-inspired approaches. *Swarm and Evolutionary Computation*, *52*, Article 100618. http://dx.doi.org/10.1016/j.swevo.2019.100618, [ISSN: 22106502 (ISSN)].

Mauri, G. R., Biajoli, F. L., Rabello, R. L., Chaves, A. A., Ribeiro, G. M., & Lorena, L. A. N. (2021). Hybrid metaheuristics to solve a multiproduct two-stage capacitated facility location problem. *International Transactions in Operational Research*, *28*(6), 3069–3093. http://dx.doi.org/10.1111/itor.12930, [ISSN: 09696016 (ISSN)].

Melo, R. A., Ribeiro, C. C., & Riveaux, J. A. (2023). A biased random-key genetic algorithm for the minimum quasi-clique partitioning problem. *Annals of Operations Research*, 1–33. http://dx.doi.org/10.1007/s10479-023-05609-7.

Mendes, J. J. M., Gonçalves, J. F., & Resende, M. G. C. (2009). A random key based genetic algorithm for the resource constrained project scheduling problem. *Computers & Operations Research*, *36*(1), 92–109. http://dx.doi.org/10.1016/j.cor.2007.07.001, [ISSN: 03050548 (ISSN)].

Mikulski, S., & Tomczewski, A. (2021). Use of energy storage to reduce transmission losses in meshed power distribution networks. *Energies*, [ISSN: 19961073] *14*(21), 7304. http://dx.doi.org/10.3390/en14217304.

Mönch, L., & Roob, S. (2018). A matheuristic framework for batch machine scheduling problems with incompatible job families and regular sum objective. *Applied Soft Computing*, *68*, 835–846. http://dx.doi.org/10.1016/j.asoc.2017.10.028, [ISSN: 15684946 (ISSN)].

Morán-Mirabal, L. F., González-Velarde, J. L., & Resende, M. G. C. (2014). Randomized heuristics for the family traveling salesperson problem. *International Transactions in Operational Research*, *21*(1), 41–57. http://dx.doi.org/10.1111/itor.12026, [ISSN: 09696016 (ISSN)].

Morán-Mirabal, L. F., González-Velarde, J. L., Resende, M. G. C., & Silva, R. M. A. (2013). Randomized heuristics for handover minimization in mobility networks. *Journal of Heuristics*, *19*(6), 845–880. http://dx.doi.org/10.1007/s10732-013-9223-0, [ISSN: 13811231 (ISSN)].

Moreira, M. C. O., Ritt, M., Costa, A. M., & Chaves, A. A. (2012). Simple heuristics for the assembly line worker assignment and balancing problem. *Journal of Heuristics*, *18*(3), 505–524. http://dx.doi.org/10.1007/s10732-012-9195-5, [ISSN: 13811231 (ISSN)].

Morgan, S. J., McGrath, C. N., & Weck, O. L. (2023). Optimization of multispacecraft maneuvers for mobile target tracking from low earth orbit. *Journal of Spacecraft and Rockets*, *60*(2), 581–590. http://dx.doi.org/10.2514/1.A35457.

Moscato, P., et al. (1989). On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. *Caltech Concurrent Computation Program, C3P Report, 826*, 1989.

Mundim, L. R., Andretta, M., & Queiroz, T. A. (2017). A biased random key genetic algorithm for open dimension nesting problems using no-fit raster. *Expert Systems with Applications*, *81*, 358–371. http://dx.doi.org/10.1016/j.eswa.2017.03.059, [ISSN: 09574174 (ISSN)].

Noronha, T. F., Resende, M. G. C., & Ribeiro, C. C. (2011). A biased random-key genetic algorithm for routing and wavelength assignment. *Journal of Global Optimization*, *50*(3), 503–518. http://dx.doi.org/10.1007/s10898-010-9608-7, [ISSN: 09255001 (ISSN)].

Ochoa, G., Malan, K., & Blum, C. (2021). Search trajectory networks: A tool for analysing and visualising the behaviour of metaheuristics. *Applied Soft Computing*, [ISSN: 15684946] *109*, Article 107492. http://dx.doi.org/10.1016/j.asoc.2021.107492.

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2018). Integrating pricing and capacity decisions in car rental: A matheuristic approach. *Operations Research Perspectives*, *5*, 334–356. http://dx.doi.org/10.1016/j.orp.2018.10.002, [ISSN: 22147160 (ISSN)].

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2022). A diversity-based genetic algorithm for scenario generation. *European Journal of Operational Research*, *299*(3), 1128–1141. http://dx.doi.org/10.1016/j.ejor.2021.09.047.

Oliveira, B. B., Carravilla, M. A., Oliveira, J. F., & Costa, A. M. (2019). A co-evolutionary matheuristic for the car rental capacity-pricing stochastic problem. *European Journal of Operational Research*, *276*(2), 637–655. http://dx.doi.org/10.1016/j.ejor.2019.01.015, [ISSN: 03772217 (ISSN)].

Oliveira, B. B., Carravilla, M. A., Oliveira, J. F., & Resende, M. G. C. (2021). A C++ application programming interface for co-evolutionary biased random-key genetic algorithms for solution and scenario generation. *Optimization Methods & Software*, *37*(3), 1065–1086. http://dx.doi.org/10.1080/10556788.2021.1884250, [ISSN: 10556788 (ISSN)].

Oliveira, A. C. M., Chaves, A. A., & Lorena, L. A. N. (2013). Clustering search. *Pesquisa Operacional*, *33*(1), 105–121. http://dx.doi.org/10.1590/S0101-74382013000100007.

Oliveira, R. M. D., Chaves, A. A., & Lorena, L. A. N. (2017). A comparison of two hybrid methods for constrained clustering problems. *Applied Soft Computing*, *54*, 256–266. http://dx.doi.org/10.1016/j.asoc.2017.01.023, [ISSN: 15684946 (ISSN)].

Oliveira, E. V., & Romero, R. (2022). Improved Biased Random Key Genetic Algorithm for the two-dimensional non-guillotine cutting problem. *Pesquisa Operacional*, *42*, 1–24. http://dx.doi.org/10.1590/0101-7438.2022.042.00260406.

Paliwal, A., Li, Y., Lubin, M., Oriol, V., Gimeno, F., Nair, V., et al. (2020). Reinforced genetic algorithm learning for optimizing computation graphs. In *ICLR' 20, Proceedings of the 8th international conference on learning representations* (pp. 1–13). URL https://openreview.net/pdf?id=rkxDoJBYPB.

Pan, Y., Zhang, L., Yan, Z., Lwin, M., & Skibniewski, M. (2021). Discovering optimal strategies for mitigating COVID-19 spread using machine learning: Experience from Asia. *Sustainable Cities and Society*, [ISSN: 22106707] *75*, Article 103254. http://dx.doi.org/10.1016/j.scs.2021.103254.

Pandey, H. M., Chaudhary, A., & Mehrotra, D. (2014). A comparative review of approaches to prevent premature convergence in GA. *Applied Soft Computing*, *24*, 1047–1077. http://dx.doi.org/10.1016/j.asoc.2014.08.025.

Pastore, T., Menna, C., & Asprone, D. (2022). Bézier-based biased random-key genetic algorithm to address printability constraints in the topology optimization of concrete structures. *Structural and Multidisciplinary Optimization*, *65*(2), 64. http://dx.doi.org/10.1007/s00158-021-03119-3.

Pedrola, O., Careglio, D., Klinkowski, M., Velasco, L., Bergman, K., & Solé-Pareta, J. (2013). Metaheuristic hybridizations for the regenerator placement and dimensioning problem in sub-wavelength switching optical networks. *European Journal of Operational Research*, *224*(3), 614–624. http://dx.doi.org/10.1016/j.ejor.2012.08.011, [ISSN: 03772217 (ISSN)].

Pedrola, O., Ruiz, M., Velasco, L., Careglio, D., González De Dios, O., & Comellas, J. (2013). A GRASP with path-relinking heuristic for the survivable IP/MPLS-over-WSON multi-layer network optimization problem. *Computers & Operations Research*, *40*(12), 3174–3187. http://dx.doi.org/10.1016/j.cor.2011.10.026, [ISSN: 03050548 (ISSN)].

Pessoa, L. S., & Andrade, C. E. (2018). Heuristics for a flowshop scheduling problem with stepwise job objective function. *European Journal of Operational Research*, *266*(3), 950–962. http://dx.doi.org/10.1016/j.ejor.2017.10.045, [ISSN: 03772217 (ISSN)].

Pessoa, L. S., Santos, A. C., & Resende, M. G. C. (2017). A biased random-key genetic algorithm for the tree of hubs location problem. *Optimization Letters*, *11*(7), 1371–1384. http://dx.doi.org/10.1007/s11590-016-1082-9, [ISSN: 18624472 (ISSN)].

Pinacho-Davidson, P., & Blum, C. (2020). Barrakuda: A hybrid evolutionary algorithm for minimum capacitated dominating set problem. *Mathematics*, *8*(11), 1–26. http://dx.doi.org/10.3390/math8111858, [ISSN: 22277390 (ISSN)].

Pinto, B. Q., Ribeiro, C. C., Riveaux, J. A., & Rosseti, I. (2021). A BRKGA-based matheuristic for the maximum quasi-clique problem with an exact local search strategy. *RAIRO - Operations Research*, *55*, S741–S763. http://dx.doi.org/10.1051/ro/2020003, [ISSN: 03990559 (ISSN)].

Pinto, B. Q., Ribeiro, C. C., Rosseti, I., & Noronha, T. F. (2020). A biased random-key genetic algorithm for routing and wavelength assignment under a sliding scheduled traffic model. *Journal of Global Optimization*, *77*(4), 949–973. http://dx.doi.org/10.1007/s10898-020-00877-0, [ISSN: 09255001 (ISSN)].

Pinto, B. Q., Ribeiro, C. C., Rosseti, I., & Plastino, A. (2018). A biased random-key genetic algorithm for the maximum quasi-clique problem. *European Journal of Operational Research*, *271*(3), 849–865. http://dx.doi.org/10.1016/j.ejor.2018.05.071, [ISSN: 03772217 (ISSN)].

Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, *34*(8), 2403–2435. http://dx.doi.org/10.1016/j.cor.2005.09.012.

Prasetyo, H., Fauza, G., Amer, Y., & Lee, S.-H. (2015). Survey on applications of biased-random key genetic algorithms for solving optimization problems. In *2015 IEEE international conference on industrial engineering and engineering management* (pp. 863–870). IEEE, http://dx.doi.org/10.1109/IEEM.2015.7385771.

Queiroga, E., Pinheiro, R. G. S., Christ, Q., Subramanian, A., & Pessoa, A. A. (2021). Iterated local search for single machine total weighted tardiness batch scheduling. *Journal of Heuristics*, *27*(3), 353–438. http://dx.doi.org/10.1007/s10732-020-09461-x, [ISSN: 13811231 (ISSN)].

Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, *76*(3), Article 036106. http://dx.doi.org/10.1103/PhysRevE.76.036106.

Ramos, A., Oliveira, J. F., Gonçalves, J. F., & Lopes, M. P. (2016). A container loading algorithm with static mechanical equilibrium stability constraints. *Transportation Research, Part B (Methodological)*, *91*, 565–581. http://dx.doi.org/10.1016/j.trb.2016.06.003, [ISSN: 01912615 (ISSN)].

Ramos, A. G., Silva, E., & Oliveira, J. F. (2018). A new load balance methodology for container loading problem in road transportation. *European Journal of Operational Research*, *266*(3), 1140–1152. http://dx.doi.org/10.1016/j.ejor.2017.10.050, [ISSN: 03772217 (ISSN)].

Raposo, A. A. M., Rodrigues, A. B., & da Guia da Silva, M. (2020). Robust meter placement for state estimation considering Distribution Network Reconfiguration for annual energy loss reduction. *Electric Power Systems Research*, *182*, Article 106233. http://dx.doi.org/10.1016/j.epsr.2020.106233, [ISSN: 03787796 (ISSN)].

Reis, R., Ritt, M., Buriol, L. S., & Resende, M. G. C. (2011). A biased random-key genetic algorithm for OSPF and DEFT routing to minimize network congestion. *International Transactions in Operational Research*, *18*(3), 401–423. http://dx.doi.org/10.1111/j.1475-3995.2010.00771.x, [ISSN: 09696016 (ISSN)].

Resende, M. G. C. (2012). Biased random-key genetic algorithms with applications in telecommunications. *Top*, *20*(1), 130–153. http://dx.doi.org/10.1007/s11750-011-0176-x.

Resende, M. G. C., Toso, R. F., Gonçalves, J. F., & Silva, R. M. A. (2012). A biased random-key genetic algorithm for the Steiner triple covering problem. *Optimization Letters*, *6*(4), 605–619. http://dx.doi.org/10.1007/s11590-011-0285-3, [ISSN: 18624472 (ISSN)].

Ribeiro, C. C., & Resende, M. G. C. (2012). Path-relinking intensification methods for stochastic local search algorithms. *Journal of Heuristics*, *18*(2), 193–214. http://dx.doi.org/10.1007/s10732-011-9167-1.

Rochman, A. N., Prasetyo, H., & Nugroho, M. T. (2017). Biased random key genetic algorithm with insertion and gender selection for capacitated vehicle routing problem with time windows. *AIP Conference Proceedings*, *1855*(1), Article 020025. http://dx.doi.org/10.1063/1.4985470.

Rocholl, J., & Mönch, L. (2021). Decomposition heuristics for parallel-machine multiple orders per job scheduling problems with a common due date. *Journal of the Operational Research Society*, *72*(8), 1737–1753. http://dx.doi.org/10.1080/01605682.2019.1640589, [ISSN: 01605682 (ISSN)].

Roque, L. A. C., Fontes, D. B. M. M., & Fontes, F. A. C. C. (2014). A hybrid biased random key genetic algorithm approach for the unit commitment problem. *Journal of Combinatorial Optimization*, *28*(1), 140–166. http://dx.doi.org/10.1007/s10878-014-9710-8, [ISSN: 13826905 (ISSN)].

Roque, L. A. C., Fontes, D. B. M. M., & Fontes, F. A. C. C. (2017). A metaheuristic approach to the multi-objective unit commitment problem combining economic and environmental criteria. *Energies*, *10*(12), 2029. http://dx.doi.org/10.3390/en10122029, [ISSN: 19961073 (ISSN)].

Ruiz, E., Albareda-Sambola, M., Fernández, E., & Resende, M. G. C. (2015). A biased random-key genetic algorithm for the capacitated minimum spanning tree problem. *Computers & Operations Research*, *57*, 95–108. http://dx.doi.org/10.1016/j.cor.2014.11.011, [ISSN: 03050548 (ISSN)].

Ruiz, M., Pedrola, O., Velasco, L., Careglio, D., Fernández-Palacios, J., & Junyent, G. (2011). Survivable IP/MPLS-over-WSON multilayer network optimization. *Journal of Optical Communications and Networking*, *3*(8), 629–640. http://dx.doi.org/10.1364/JOCN.3.000629, [ISSN: 19430620 (ISSN)].

Ruiz, E., Soto-Mendoza, V., Ruiz Barbosa, A. E., & Reyes, R. (2019). Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm. *Computers & Industrial Engineering*, *133*, 207–219. http://dx.doi.org/10.1016/j.cie.2019.05.002, [ISSN: 03608352 (ISSN)].

Ruiz, R., & Stützle, T. (2007). A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, *177*(3), 2033–2049. http://dx.doi.org/10.1016/j.ejor.2005.12.009.

Schenekemberg, C. M., Chaves, A. A., Coelho, L. C., Guimarães, T. A., & Avelino, G. G. (2022). The dial-a-ride problem with private fleet and common carrier. *Computers & Operations Research*, *147*, Article 105933. http://dx.doi.org/10.1016/j.cor.2022.105933.

Schuetz, M. J. A., Brubaker, J. K., Montagu, H., van Dijk, Y., Klepsch, J., Ross, P., et al. (2022). Optimization of robot-trajectory planning with nature-inspired and hybrid quantum algorithms. *Physical Review A*, *18*(5), Article 054045. http://dx.doi.org/10.1103/PhysRevApplied.18.054045.

Seuring, S., & Gold, S. (2012). Conducting content-analysis based literature reviews in supply chain management. *Supply Chain Management: An International Journal*, http://dx.doi.org/10.1108/13598541211258609.

Silva, T. T., Chaves, A. A., Yanasse, H. H., & Luna, H. P. L. (2019). The multicommodity traveling salesman problem with priority prizes: a mathematical model and metaheuristics. *Computational & Applied Mathematics*, *38*(4), 188. http://dx.doi.org/10.1007/s40314-019-0976-4, [ISSN: 22383603 (ISSN)].

Silva, G., Leite, A., Ospina, R., Leiva, V., Figueroa-Zúñiga, J., & Castro, C. (2023). Biased random-key genetic algorithm with local search applied to the maximum diversity problem. *Mathematics*, *11*(14), 3072. http://dx.doi.org/10.3390/math11143072.

Silva, R. M. A., Resende, M. G. C., & Pardalos, P. M. (2014). Finding multiple roots of a box-constrained system of nonlinear equations with a biased random-key genetic algorithm. *Journal of Global Optimization*, *60*(2), 289–306. http://dx.doi.org/10.1007/s10898-013-0105-7, [ISSN: 09255001 (ISSN)].

Silva, R. M. A., Resende, M. G. C., & Pardalos, P. M. (2015). A Python/C++ library for bound-constrained global optimization using a biased random-key genetic algorithm. *Journal of Combinatorial Optimization*, *30*(3), 710–728. http://dx.doi.org/10.1007/s10878-013-9659-z, [ISSN: 13826905 (ISSN)].

Silva, S. E., Ribeiro, C. C., & dos Santos Souza, U. (2023). A biased random-key genetic algorithm for the chordal completion problem. *RAIRO-Operations Research*, *57*(3), 1559–1578. http://dx.doi.org/10.1051/ro/2023081.

Silva-Soto, Y. I., & Ibarra-Rojas, O. J. (2021). Timetabling with flexible frequencies to synchronise groups of bus lines at common stops. *Transportmetrica A: Transport Science*, *17*(4), 978–1001. http://dx.doi.org/10.1080/23249935.2020.1822952, [ISSN: 23249935 (ISSN)].

Singh, V. K., Singh, P., Karmakar, M., Leta, J., & Mayr, P. (2021). The journal coverage of Web of Science, Scopus and Dimensions: A comparative analysis. *Scientometrics*, *126*(6), 5113–5142. http://dx.doi.org/10.1007/s11192-021-03948-5.

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, *104*, 333–339. http://dx.doi.org/10.1016/j.jbusres.2019.07.039.

Soares, L. C. R., & Carvalho, M. A. M. (2020). Biased random-key genetic algorithm for scheduling identical parallel machines with tooling constraints. *European Journal of Operational Research*, *285*(3), 955–964. http://dx.doi.org/10.1016/j.ejor.2020.02.047, [ISSN: 03772217 (ISSN)].

Soares, L. C., & Carvalho, M. A. (2022). Application of a hybrid evolutionary algorithm to resource-constrained parallel machine scheduling with setup times. *Computers & Operations Research*, *139*, Article 105637. http://dx.doi.org/10.1016/j.cor.2021.105637.

Souza Queiroz, L. R. D., & Andretta, M. (2020). Two effective methods for the irregular knapsack problem. *Applied Soft Computing*, *95*, Article 106485. http://dx.doi.org/10.1016/j.asoc.2020.106485, [ISSN: 15684946 (ISSN)].

Spears, W. M., & De Jong, K. A. (1991). An analysis of multi-point crossover. *Vol. 1*, In *Foundations of genetic algorithms* (pp. 301–315). Elsevier, http://dx.doi.org/10.1016/B978-0-08-050684-5.50022-7.

Stefanello, F., Aggarwal, V., Buriol, L. S., & Resende, M. G. C. (2019). Hybrid algorithms for placement of virtual machines across geo-separated data centers. *Journal of Combinatorial Optimization*, *38*(3), 748–793. http://dx.doi.org/10.1007/s10878-019-00411-3, [ISSN: 13826905 (ISSN)].

Stefanello, F., Buriol, L. S., Hirsch, M. J., Pardalos, P. M., Querido, T., Resende, M. G. C., et al. (2017). On the minimization of traffic congestion in road networks with tolls. *Annals of Operations Research*, *249*(1–2), 119–139. http://dx.doi.org/10.1007/s10479-015-1800-1, [ISSN: 02545330 (ISSN)].

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*(4), 341–359. http://dx.doi.org/10.1023/A:1008202821328.

Sun, B., & Zhou, Y. (2022). Bayesian network structure learning with improved genetic algorithm. *International Journal of Intelligent Systems*, *37*(9), 6023–6047. http://dx.doi.org/10.1002/int.22833.

Tangpattanakul, P., Jozefowiez, N., & Lopez, P. (2015a). Biased random key genetic algorithm for multi-user earth observation scheduling. *Recent Advances in Computational Optimization*, *580*, 143–160. http://dx.doi.org/10.1007/978-3-319-12631-9_9, [ISSN: 1860949X (ISSN)].

Tangpattanakul, P., Jozefowiez, N., & Lopez, P. (2015b). A multi-objective local search heuristic for scheduling Earth observations taken by an agile satellite. *European Journal of Operational Research*, *245*(2), 542–554. http://dx.doi.org/10.1016/j.ejor.2015.03.011, [ISSN: 03772217 (ISSN)].

Thomé, A. M. T., Scavarda, L. F., & Scavarda, A. J. (2016). Conducting systematic literature review in operations management. *Production Planning and Control*, *27*(5), 408–420. http://dx.doi.org/10.1080/09537287.2015.1129464.

Toso, R. F., & Resende, M. G. C. (2015). A C++ application programming interface for biased random-key genetic algorithms. *Optimization Methods & Software*, *30*(1), 81–93. http://dx.doi.org/10.1080/10556788.2014.890197, [ISSN: 10556788 (ISSN)].

Valente, J. M. S., & Gonçalves, J. F. (2009). A genetic algorithm approach for the single machine scheduling problem with linear earliness and quadratic tardiness penalties. *Computers & Operations Research*, *36*(10), 2707–2715. http://dx.doi.org/10.1016/j.cor.2008.11.016, [ISSN: 03050548 (ISSN)].

Valente, J. M. S., Gonçalves, J. F., & Alves, R. A. F. S. (2006). A hybrid genetic algorithm for the early/tardy scheduling problem. *Asia-Pacific Journal of Operational Research*, *23*(3), 393–405. http://dx.doi.org/10.1142/S0217595906000978, [ISSN: 02175959 (ISSN)].

Velasco, L., Wright, P., Lord, A., & Junyent, G. (2013). Saving CAPEX by extending flexgrid-based core optical networks toward the edges. *Journal of Optical Communications and Networking*, *5*(10), A171–A183. http://dx.doi.org/10.1364/JOCN.5.00A171, [ISSN: 19430620 (ISSN)].

Villicaña-Cervantes, D., & Ibarra-Rojas, O. J. (2022). Accessible location of mobile labs for COVID-19 testing. *Health Care Management Science*, 1–19. http://dx.doi.org/10.1007/s10729-022-09614-3.

Wang, C., & Uzsoy, R. (2002). A genetic algorithm to minimize maximum lateness on a batch processing machine. *Computers & Operations Research*, *29*(12), 1621–1640. http://dx.doi.org/10.1016/S0305-0548(01)00031-4, [ISSN: 03050548 (ISSN)].

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*, 279–292. http://dx.doi.org/10.1007/BF00992698.

Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS Quarterly*, xiii–xxiii.

Whitley, D., Rana, S., & Heckendorn, R. B. (1999). The island model genetic algorithm: On separability, population size and convergence. *Journal of Computing and Information Technology*, *7*(1), 33–47.

Won, S., Bae, H., & Sutrisnowati, R. A. (2018). Production scheduling in steel manufacturing with cutting and parallel operations. *ICIC Express Letters, Part B: Applications*, *9*(8), 869–877. http://dx.doi.org/10.24507/icicelb.09.08.869, [ISSN: 21852766 (ISSN)].

Xie, Y., Sheng, Y., Qiu, M., & Gui, F. (2022). An adaptive decoding biased random key genetic algorithm for cloud workflow scheduling. *Engineering Applications of Artificial Intelligence*, *112*, Article 104879. http://dx.doi.org/10.1016/j.engappai.2022.104879.

Xu, Y., Guo, P., & Zeng, Y. (2022). An iterative neighborhood local search algorithm for capacitated centered clustering problem. *IEEE Access*, *10*, 34497–34510. http://dx.doi.org/10.1109/ACCESS.2022.3162692.

Yang, X.-S. (2012). Flower pollination algorithm for global optimization. In *International conference on unconventional computing and natural computation* (pp. 240–249). Springer, http://dx.doi.org/10.1007/978-3-642-32894-7_27.

Yu, X., Qian, J., Zhang, Y., & Kong, M. (2023). Supply chain scheduling method for the coordination of agile production and port delivery operation. *Mathematics*, *11*(15), 3276. http://dx.doi.org/10.3390/math11153276.

Zhang, J., & Sanderson, A. C. (2009). JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*, *13*(5), 945–958. http://dx.doi.org/10.1109/TEVC.2009.2014613.

Zhang, X., Shan, M., & Zeng, J. (2022). Parallel batch processing machine scheduling under two-dimensional bin-packing constraints. *IEEE Transactions on Reliability*, 1265. http://dx.doi.org/10.1109/TR.2022.3201333.

Zhao, W., Liu, Y., Wang, Y., & Yue, X. (2023). Distributed spot welding task allocation and sequential planning for multi-station multi-robot coordinate assembly processes. *International Journal of Advanced Manufacturing Technology*, *127*(11–12), 5233–5251. http://dx.doi.org/10.1007/s00170-023-11750-1.

Zheng, J., Chien, C., & Gen, M. (2015). Multi-objective multi-population biased random-key genetic algorithm for the 3-D container loading problem. *Computers & Industrial Engineering*, *89*, 80–87. http://dx.doi.org/10.1016/j.cie.2014.07.012, [ISSN: 03608352 (ISSN)].

Zitzler, E., & Künzli, S. (2004). Indicator-based selection in multiobjective search. In *International conference on parallel problem solving from nature* (pp. 832–842). Springer, http://dx.doi.org/10.1007/978-3-540-30217-9_84.

Zudio, A., da Silva Costa, D. H., Masquio, B. P., Coelho, I. M., & Pinto, P. E. D. (2018). BRKGA/VND hybrid algorithm for the classic three-dimensional bin packing problem. *Electronic Notes in Discrete Mathematics*, *66*, 175–182. http://dx.doi.org/10.1016/j.endm.2018.03.023, [ISSN: 15710653 (ISSN)].