An Evolutionary Algorithm applied to the Multi-Objective Multi-Dimensional Knapsack Problem

Luis Henrique Pauleti Mendes
Instituto de Computação - UNICAMP
Av. Albert Einstein, 1251, Cidade Universitária, 13083-852, Campinas, SP, Brasil
luis.mendes@ic.unicamp.br

Fábio Luiz Usberti
Instituto de Computação - UNICAMP
Av. Albert Einstein, 1251, Cidade Universitária, 13083-852, Campinas, SP, Brasil
fusberti@ic.unicamp.br

Mário César San Felice
Departamento de Computação - UFSCar
Rod. Washington Luís, Km 235, 13565-905, São Carlos, SP, Brasil
felice@ufscar.br

ABSTRACT

This paper presents an evolutionary algorithm for multi-objective optimization problems, based on the Biased Random-Key Genetic Algorithm (BRKGA) and on the Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II). Computational experiments applied to the Multi-Objective Multi-Dimensional Knapsack Problem compared our algorithm with other multi-objective metaheuristics algorithms from the literature, namely the NSGA-II, the Non-dominated Sorting Particle Swarm Optimization (NSPSO), the Multi-Objective Evolutionary Algorithm by Decomposition with Differential Evolution (MOEA/D-DE), the Multi-objective Hypervolume-based Ant Colony Optimizer (MHACO), and the Improved Harmony Search (IHS). The results show that our methodology obtained competitive results with respect to the hypervolumes from the obtained non-dominated fronts.

KEYWORDS. Multi-Objective Optimization. BRKGA. NSGA-II.

ADM - Apoio à Decisão Multicritério. MH - Metaheurísticas. OC - Otimização Combinatória

https://proceedings.science/p/157504?lang=pt-br

1. Introduction

Multi-objective Optimization Problems (MOPs) are an important field of Operations Research (OR) and has a tremendous practical importance. Most real-world problems require taking into account multiple conflicting point of views, that are ideally suited to be modeled using multiple conflicting objectives [Deb, 2014].

Formally, a MOP can be stated as

$$\max_{x \in \mathcal{X}} \mathbf{f}(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T,$$

where $\mathcal{X} \subset \mathbb{R}^n$ is an n-dimensional decision space, and $\mathbf{f} : \mathcal{X} \to \mathbb{R}^m$ consists of m real-valued objective functions, where each one needs to be maximized, and $\mathcal{Y} = \{\mathbf{f}(x) : x \in \mathcal{X}\} \subset \mathbb{R}^m$ is an attainable objective space.

The m objective functions in $\mathbf{f}$ often conflict with each other. Improvement of one objective function $f_i \in \mathbf{f}$ may lead to deterioration of another objective function $f_j \in \mathbf{f}$. Thus, no single solution can optimize all the objectives simultaneously. Instead, the best trade-off solutions, called efficient solutions, are of interest to a decision maker [Peng et al., 2009; Zhou et al., 2011]. Some definitions that characterize these efficient solutions are presented next.

First, the dominance relation is defined [Lust e Teghem, 2010]:

Definition 1.1 (Dominance relation)**.** An objective point $y' \in \mathbb{R}^m$ dominates another objective point $y'' \in \mathbb{R}^m$, denoted as $y' \prec y''$, if and only if:

1. $y'_i \geq y''_i$ for every $i \in \{1, \ldots, m\}$, and
2. $y'_j > y''_j$ for at least one $j \in \{1, \ldots, m\}$.

If y' does not dominate objective point y'', then $y' \not\prec y''$. Similarly, we write $y' \succ y''$ if objective point y' is dominated by objective point y'', and $y' \not\succ y''$ if objective point y' is not dominated by objective point y''.

Now, an efficient solution, a non-dominated point, the efficient set, and the non-dominated front can be defined [Lust e Teghem, 2010]:

Definition 1.2 (Efficient solution)**.** A feasible solution $x^\star \in \mathcal{X}$ is called efficient if and only if there does not exist any other feasible solution $x \in \mathcal{X}$ such that $\mathbf{f}(x) \prec \mathbf{f}(x^\star)$.

Definition 1.3 (Non-dominated point)**.** The image $\mathbf{f}(x^\star) \in \mathcal{Y}$ of an efficient solution $x^\star \in \mathcal{X}$ is called a non-dominated point.

Definition 1.4 (Efficient set)**.** The efficient set, denoted by $\mathcal{X}_E$, contains all the efficient solutions.

Definition 1.5 (Non-dominated front)**.** The image of the efficient set $\mathcal{X}_E$ in the attainable objective space $\mathcal{Y}$ is called the non-dominated front, and is denoted by $\mathcal{Y}_N$.

Many MOPs may have a huge (or even infinite) number of efficient solutions. It could be very time-consuming, if possible, to obtain the complete efficient set $\mathcal{X}_E$ [Peng et al., 2009]. Hence, unlike the goal in single-objective optimization of finding a solution with objective value as close to the optimum as possible, in multi-objective optimization there are three distinct goals [Deb, 2001; Riquelme et al., 2015]: *cardinality*: to find a set with a considerable number of solutions; *convergence*: to find a set of solutions as close as possible to the theoretical efficient set $\mathcal{X}_E$; and *diversity*: to find a set of maximally spread solutions.

The classical methods to solve MOPs are mainly focused on scalarization, which involves formulating single-objective optimization problems related to the original problem. When such a method is to be used for finding multiple solutions, it has to be applied many times, hopefully finding a different solution at each simulation run [Deb, 2014]. In contrast, population-based methods such as Evolutionary Algorithms (EAs) have become increasingly popular for solving MOPs over the past decades [Li, 2003]. Besides the generic ability to handle large complex real-world problems of traditional EAs, Multi-Objective Evolutionary Algorithms (MOEAs) have their unique advantage, namely finding a set of representative efficient solutions rather than just one solution in a run, because of their inherent parallel character and group strategy [Yan et al., 2003].

Our contribution consists of a new MOEA applied to the Multi-Objective Multi-Dimensional Knapsack Problem (MOMDKP), whose performance is compared with other MOEAs from the literature. Our methodology combines the benefits of Biased Random-Key Genetic Algorithms (BRKGAs) and of the Elitist Non-Dominated Sorting Genetic Algorithm (NSGA-II). We believe our methodology can be easily applied to MOPs in general.

2. Base algorithms

In this section we describe the meta-heuristic frameworks in which our proposed algorithm was based.

2.1. Biased Random-Key Genetic Algorithms

A Genetic Algorithm (GA) [Goldberg, 1989; Holland, 1975] seeks to find good quality solutions for optimization problems by a paradigm that mimics evolution. An analogy is made between a pool of solutions and a population of individuals. Each individual in the population is encoded as a chromosome, that consists of a string of genes. Each gene can take on a value, called allele, from some alphabet. The individuals that make up the population are evolved over a number of generations. At each generation, a new population is created by combining individuals of the current population, using a crossover operator, to produce offspring that make up the next generation. Random mutation also takes place as a mean to escape entrapment in local optimum. The Darwinian concept of survival of the fittest is applied when individuals are selected to mate and produce offspring. Individuals are selected at random, but those with better fitness are preferred over those that are less fit.

Random-Key Genetic Algorithms (RKGAs) were first introduced by Bean [1994] for solving single-objective combinatorial optimization problems involving sequencing. In a RKGA, chromosomes are represented as vectors of randomly generated real numbers in the interval $[0, 1]$, that is, as points in the n-dimensional unit hypercube $[0, 1]^n$. A deterministic algorithm, called decoder, maps any individual to a solution of the optimization problem, for which a fitness value can be computed.

The initial population P_0 is made up of vectors of random-keys, where each allele is generated independently at random in the interval $[0, 1]$. After the fitness of each individual is computed by the decoder, the population P_0 is partitioned into a small set of elite individuals P_0^e, that is, those with the best fitness value, and a larger set of non-elite individuals $P_0^{\bar{e}} = P_0 \setminus P_0^e$, containing the remaining individuals. In each generation t, the elite set P_t^e is copied unchanged to generation $t + 1$ and a small set P_t^m of random solutions called mutants are also introduced into the new population. Considering that $|P_t^e \cup P_t^m| \leq |P_t|$, the population is completed by offspring, generated through the process of mating.

A Biased Random-Key Genetic Algorithm (BRKGA) [Gonçalves e Resende, 2011] differs from a RKGA in the way parents are selected for mating. In a RKGA [Bean, 1994], the two parents are selected at random from the entire population. Whereas in a BRKGA, each offspring is

generated combining one individual from the elite set P_t^e and one individual from the non-elite set $P_t^{\bar{e}}$. Since $|P_t^e| < |P_t^{\bar{e}}|$, the probability that a given elite individual is selected for mating $(1/|P_t^e|)$ is greater than that of a given non-elite individual $(1/|P_t^{\bar{e}}|)$, and therefore the given elite individual has a higher likelihood to pass on its characteristics to future generations than does a given non-elite individual. Moreover, mating is done using parameterized uniform crossover [Spears e De Jong, 1995], where a gene is taken from the elite individual with probability $\rho > 0.5$, or otherwise it is taken from the other individual.

Two variants of BRKGA were presented by Lucena et al. [2014]. In the Gender Defining Biased Random-Key Genetic Algorithm (BRKGA-GD), each chromosome has a bit flag indicating its gender, and mating is allowed only between individuals of different gender. In the Multi-Parent Biased Random-Key Genetic Algorithm (BRKGA-MP), several parents are used to generate a new offspring. A BRKGA-MP selects as many parents as the parameter π determines, and rank them according to their fitness values. Among these, π^e are elite parents and $\pi^{\bar{e}} = \pi - \pi^e$ are non-elite parents, where $\pi^e \geq \pi^{\bar{e}}$. Then, a parameterized uniform crossover is performed such that each allele is taken from an individual chosen by the roulette method. Each parent has a probability of passing its alleles to the offspring, which is calculated by taking the bias of the parent into account. Parent bias is defined by a pre-determined, non-increasing weighting bias function $\Phi : \mathbb{N} \to \mathbb{R}_+^*$ over its rank r. The following bias functions, proposed by Bresina [1996], are used: *logarithmic*: $\Phi(r) = \frac{1}{\log(r+1)}$; *linear*: $\Phi(r) = \frac{1}{r}$; *polynomial (in d)*: $\Phi(r) = \frac{1}{r^d}$; *exponential*: $\Phi(r) = \frac{1}{e^r}$; *constant*: $\Phi(r) = \frac{1}{\pi}$.

2.2. Elitist Non-dominated Sorting Genetic Algorithm

The Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II) [Deb et al., 2002] for finding multiple Pareto-optimal solutions in a MOP has the following three features: it uses an elitist principle; it uses an explicit diversity preserving mechanism; and it emphasizes the non-dominated solutions.

Initially, a random parent population P_0 is created. The population is sorted based on the non-domination. Each solution receives a rank equal to its non-domination level. At first, the binary tournament selection, simulated binary crossover and polynomial mutation are used to create an offspring population Q_0 of size $|P_0|$. Elitism is introduced by comparing current population with previously found best non-dominated solutions.

Next, we describe the t-th generation of the NSGA-II. First, a combined population $R_t = P_t \cup Q_t$, of size $2|P_t|$, is formed. Then, the fast non-dominated sorting is used to classify the entire population R_t. Although this requires more computation compared to performing a non-dominated sorting on Q_t alone, it allows a global non-domination check among offspring and parent solutions, ensuring elitism. Now, solutions belonging to the best non-dominated set $S_1^\star$ are the best solution in R_t and must be emphasized more than any other solution in R_t. If the size of $S_1^\star$ is smaller than $|P_t|$, we choose all members of $S_1^\star$ for the next population P_{t+1}. The remaining members of the population P_{t+1} are chosen from subsequent non-dominated set in order of their ranking. Thus, solutions from the set $S_2^\star$ are chosen next, followed by solutions from the set $S_3^\star$, and so on. This procedure is continued until no more sets can be accommodated. When the last acceptable non-dominated set $S_l^\star$ is being considered, there may exist more solutions in the $S_l^\star$ than the remaining slots in the new population P_{t+1}. Instead of arbitrarily discarding some members of the $S_l^\star$, a niching strategy is used to choose the solutions that reside in the least crowded region of $S_l^\star$, increasing the diversity of the solutions chosen.

A strategy like the above does not affect the proceedings of the algorithm much in the early stages of evolution. This is because, early on, there exist many non-dominated sets in the combined

https://proceedings.science/p/157504?lang=pt-br

population R_t. It is likely that solutions of many good non-dominated sets are already included in the new population P_{t+1}, before they add up to $|P_t|$. It then hardly matters which solution is included to fill up the new population P_{t+1}. However, during the latter stages of the evolution, it is likely that most solutions in the combined population R_t lie in the best non-dominated set $S_1^\star$, exceeding the size $|P_t|$. The algorithm then ensures that niching will choose a diverse set of solutions from R_t. When the entire population P_t converges to the Pareto-optimal set, the algorithm will ensure a better spread among the solutions.

3. Non-dominated Sorting Biased Random-Key Genetic Algorithm

In this work, we extend the single-objective BRKGA-MP framework for solving MOPs. The fitness of each chromosome must be taken into account for all objective functions. Thus, the main question that arises is how to determine an order for solutions of each population to select the elite set. To that end, we used the fast non-dominated sorting and the crowding distance assignment from NSGA-II.

Our proposed algorithm, that combines features from the BRKGA-MP and the NSGA-II, is named Non-dominated Sorting Biased Random-key Genetic Algorithm (NS-BRKGA). The t-th generation of the NS-BRKGA is presented in Algorithm 1.

Algorithm 1 NS-BRKGA

1: Sort P_t using fast-non-dominated-sort and crowding-distance-sort
2: $P_t^e \leftarrow$ select-elite-set$(P_t, S_1^\star, D, p^e_{\min}, p^e_{\max})$
3: $P_{t+1} \leftarrow P_t^e$ ▷ Initialize next population
4: **while** $|P_{t+1}| < |P_t|$ **do**
5: $\quad c \leftarrow$ biased-multi-parent-crossover$(P_t, P_t^e, \pi, \pi^e, \Phi)$
6: $\quad c' \leftarrow$ polynomial-mutation(c)
7: $\quad P_{t+1} \leftarrow P_{t+1} \cup \{c'\}$ ▷ Add offspring to next population
8: $t \leftarrow t + 1$

The standard BRKGA have already been hybridized with NSGA-II to tackle MOPs [Tangpattanakul et al., 2015; Damm e Ronconi, 2021; Chagas et al., 2021]. However, to the best of our knowledge, no other work from the literature have tried to adapt the multi-parent version of the BRKGA for multi-objective optimization. Furthermore, we do not insert mutant individuals into the population. Instead, we apply polynomial mutation on the offsprings.

Another difference from the NS-BRKGA to the standard BRKGA is the way the elite set is selected. The crowding distance sorting preserves the diversity on the objective space. So, to enhance the diversity on the genotype space, we emphasize solutions with high diversity to be part of the elite set. Such mechanism is described on Algorithm 2.

Line 1 initializes the elite set P_t^e with the first $p^e_{\min}$ elements of the current population P_t. Lines $2-3$ make sure that P_t^e contains every non-dominated solutions $S_1^\star$ from P_t. Lines $4-5$ ensure that P_t^e do not exceed the maximum size $p^e_{\min}$. The loop of lines $6-8$ selects the prefix of P_t that maximizes the value of the diversity function D.

The following diversity measures are available [Lacevic e Amaldi, 2011]: average distance to centroid: $D_v(P) = \dfrac{1}{|P|}\displaystyle\sum_{i=1}^{|P|} \|P[i] - \bar{P}\|$, where $\bar{P} = \dfrac{1}{|P|}\displaystyle\sum_{i=1}^{|P|} P[i]$ is the centroid of the population; average distance between all pairs: $D_v(P) = \dfrac{2}{|P|(|P|-1)}\displaystyle\sum_{i=1}^{|P|-1}\sum_{j=i+1}^{|P|} \|P[i] - P[j]\|$;

https://proceedings.science/p/157504?lang=pt-br

Algorithm 2 Select elite set

$P_t^e \leftarrow P_t[1 : p^e_{\min}]$
if $|P_t^e| < |S_1^\star|$ **then**
  $P_t^e \leftarrow S_1^\star$
if $|P_t^e| > p^e_{\max}$ **then**
  $P_t^e \leftarrow P_t[1 : p^e_{\max}]$
for $p \leftarrow |P_t^e| + 1$ **to** $p^e_{\max}$ **do**
  if $D(P_t^e) < D(P_t[1 : p])$ **then**
    $P_t^e \leftarrow P_t[1 : p]$

power mean based: $D_v(P) = \sqrt[\beta]{\frac{1}{|P|}\sum_{i=1}^{|P|-1}\left(\frac{1}{|P|-1}\sum_{j=1}^{|P|}\|P[i]-P[j]\|^\alpha\right)}$, with $\alpha = -\infty$ and $\beta = 1$.

4. Multi-Objective Multi-Dimensional Knapsack Problem

The Knapsack Problem (KP), introduced by Dantzig [1957], is among the most popular combinatorial optimization problems in OR, with a vast range of real-world applications. The KP consists of a set of items, with weight and profit associated to each item, and an upper bound for the capacity of the knapsack. The task is to find a subset of items which maximizes the total profit, while respecting the capacity of the knapsack [Martello e Toth, 1990]. Despite being easy to understand and simple to formulate, the KP is notoriously difficult to solve. In the theory of computational complexity, the KP belongs to the $\mathcal{NP}$-complete class [Garey e Johnson, 1979].

When in presence of several constraints, the KP is known in the OR literature as the Multi-Dimensional Knapsack Problem (MDKP), and is far more difficult than the KP [Kellerer et al., 2004]. Since the real-world applications are characterized by their multi-attribute nature, the extension of the MDKP to multi-objective is very natural [Mavrotas et al., 2015]. This new problem is known as the Multi-Objective Multi-Dimensional Knapsack Problem (MOMDKP). The KP can also be extended directly to the MOMDKP by allowing an arbitrary number of knapsacks [Zitzler e Thiele, 1999].

Formally, an instance of the MOMDKP consists of a set of items $I = \{1, \ldots, n\}$ and a set of knapsacks $K = \{1, \ldots, m\}$, along with the profit $p_{i,k} \in \mathbb{N}^*$ and the weight $w_{i,k} \in \mathbb{N}^*$ of each item $i \in I$ according to each knapsack $k \in K$, and the capacity $c_k \in \mathbb{N}^*$ of each knapsack $k \in K$. The goal is to find a subset of items $I' \subseteq I$ which maximizes the total profits while not exceeding the capacities of the knapsacks. Mathematically, the MOMDKP is formulated as follows:

$$\max \mathbf{f}(x) = (f_1(x), \ldots, f_m(x))^T, \text{ where } f_k(x) = \sum_{i \in I} p_{i,k} x_i$$

subject to:

$$\sum_{i \in I} w_{i,k} x_i \leq c_k, \forall k \in K$$
$$x_i \in \{0, 1\}, \qquad \forall i \in I$$

where $x_i = 1$ if and only if $i \in I'$.

Determining whether a solution is inefficient is an $\mathcal{NP}$-complete task for many MOPs [Samanlioglu et al., 2008]. Hence, the MOMDKP has the difficulty of the MDKP itself combined with the difficulty of having multiple objectives. Many practical problems, such as capital budgeting and allocating processors, can be formulated as MOMDKP [Cerqueus et al., 2015]. Consequently, the MOTSP is frequently employed as a benchmark problem to evaluate the performance of multi-objective metaheuristics [Zitzler e Thiele, 1999; Shah e Reed, 2011; Mavrotas et al., 2015; Cerqueus et al., 2015].

5. Computational experiments

Instances. In this paper, we consider the MOMDKP instances from Zitzler e Thiele [1999], that have two, three, and four knapsacks, in combination with 100, 250, 500, and 750 items. These 12 instances have uncorrelated profits and weights, that are random integers in the interval $[10, 100]$. The knapsack capacities are equal to half the total weight of the items of the corresponding knapsack.

Decoder. A chromosome that encodes a solution for an instance of the MOMDKP is represented by a vector with n real numbers in the interval $[0, 1]$. The value of each of the n genes determine the order that each item $i \in I$ will be inserted in the knapsacks. An item is selected to be part of the solution I' only if, in doing so, the m capacities are not exceeded; otherwise, the next item in order is considered. Hence, the decoder sorts the genes of the chromosome, sweeps the items selecting the ones that will be inserted in the solution I', and sums up the total weight and total profit for each of the m knapsacks, which gives a computational time complexity of $\mathcal{O}(n \log n + nm)$.

Benchmark. The proposed algorithm was compared with the following algorithms, whose implementations are available at the PAGMO2 library [Biscani e Izzo, 2020], maintained by the European Space Agency[1]: Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II) [Deb et al., 2002]; Non-dominated Sorting Particle Swarm Optimization (NSPSO) [Li, 2003, 2004]; Multi-Objective Evolutionary Algorithm by Decomposition with Differential Evolution (MOEA/D-DE) [Zhang e Li, 2007; Li e Zhang, 2008]; Multi-objective Hypervolume-based Ant Colony Optimizer (MHACO) [Acciarini et al., 2020]; Improved Harmony Search (IHS) [Mahdavi et al., 2007].

Parameters settings. Five independent optimization runs with a time limit of one hour were performed for each algoritm and instance on a computer with an Intel Xeon CPU E3-1230 V2 of 3.30GHz and 32GB of RAM. The population size varies with the number of items and knapsacks of the instances, as can be seen in Table 1. For the NS-BRKGA we used $\pi = 3$ parents, with $\pi^e = 2$

Table 1: Population sizes used for computational experiments.

Population size					
		Number of items			
		100	250	500	750
Number of knapsacks	2	152	152	200	252
	3	200	200	252	300
	4	252	252	300	352

elite parents, logarithmic bias function, mutation probability equal to 0.006, elite set size varying from 10% to 50% of the population size, and average distance to centroid as diversity measure. For the NSGA-II we fixed the crossover and mutation probabilities as 0.80 and 0.006, respectively,

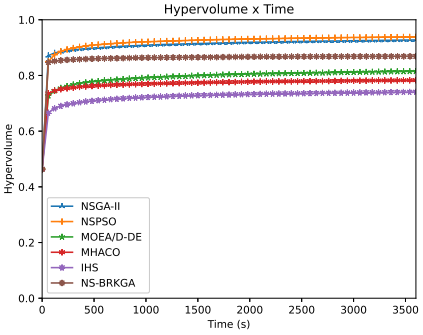
[1] https://esa.github.io/pagmo2/

https://proceedings.science/p/157504?lang=pt-br

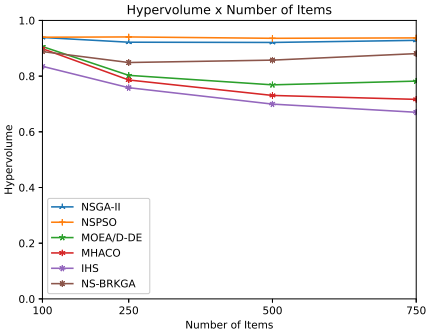
following the configurations of Shah e Reed [2011]. For the others parameters, we used the default values of the PAGMO2 library [Biscani e Izzo, 2020].

Evaluation metric. The metric chosen to compare the MOEAs in terms of the convergence and diversity of the solutions is the hypervolume indicator I_{HV} [Shang et al., 2021], also known as Lebesgue measure or S-metric [Nowak et al., 2014]. The I_{HV} associated with a set of solutions $S \subset \mathcal{X}$ and a reference objective point $\mathbf{z}^{\mathbf{ref}} \in \mathbb{R}^m$, is calculated as the size of the portion of the objective space $\mathbb{R}^m$ weakly dominated by S and bounded by $\mathbf{z}^{\mathbf{ref}}$. Formally, we have that $I_{HV}(S) = \Lambda(\{\mathbf{z} \in \mathbb{R}^m | \exists x \in S(\mathbf{f}(x) \preceq \mathbf{z} \preceq \mathbf{z}^{\mathbf{ref}})\})$, where $\Lambda(\cdot)$ denotes the Lebesgue measure. Since a fixed reference point $\mathbf{z}^{\mathbf{ref}}$ is assumed, it is omitted as an argument of $I_{HV}(\cdot)$. In this study, we adopted $\mathbf{z}^{\mathbf{ref}} = \mathbf{0}$.

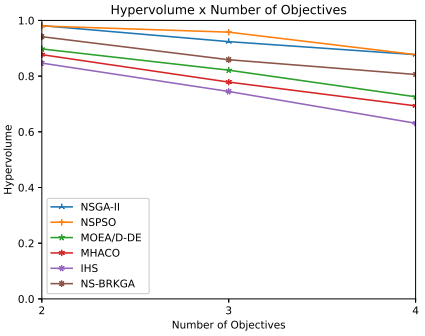
Results. Analyzing the plots in Figure 1, we can conclude that the NS-BRKGA has a competitive performance compared with the other MOEAs from the literature. On Figure 1(a) we can see that the NS-BRKGA converges more rapidly to solutions close to the efficient set X_E, however it seems to get stuck in a local-optimum region of the objective space. Moreover, on Figure 1(b) we can notice that the NS-BRKGA is scalable, showing increasing margins of improvements for bigger instances. However, despite achieving a worse hypervolume than NSPSO and NSGA-2, Figure 2 shows that,



(a)



(b)



(c)

Figure 1: Hypervolume obtained by each MOEA

for the biggest instance with $m = 2$ knapsacks, the solutions found by NS-BRKGA dominates a considerable number of solutions found by NSPSO and NSGA-2, whereas the solutions found by NSPSO or NSGA-2 does not dominate any solution found by NS-BRKGA. A similar behaviour was noticed for the other instances.

https://proceedings.science/p/157504?lang=pt-br

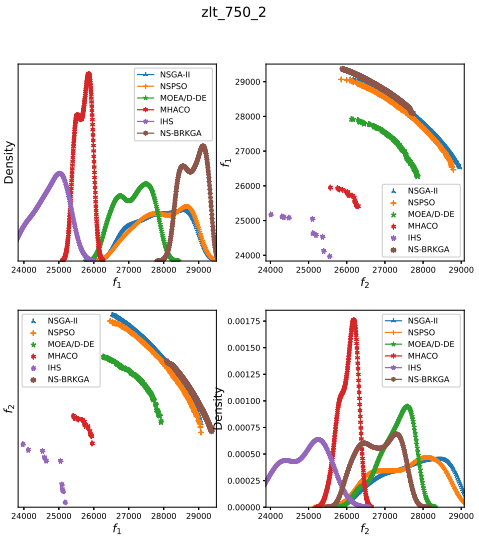


Figure 2: Solutions obtained by the best run of each algorithm to the instance with $n = 750$ items and $m = 2$ knapsacks.

6. Future work

In this work, we have studied the performance of the NS-BRKGA using the MOMDKP as a benchmark. A promising line of future research is to assess the performance of the NS-BRKGA in solving other multi-objective optimization problems. As good candidate problems, we can cite the Multi-Objective Traveling Salesman Problem [Lust e Teghem, 2010], the Flexible Job-Shop Scheduling Problem [Kacem et al., 2002], the Multi-Objective Facility Location Problem [Karatas e Yakıcı, 2018] and the Multi-Objective Portfolio Optimization Problem [Metaxiotis e Liagkouras, 2012].

Andrade et al. [2021] presented the Multi-parent Biased Random-key Genetic Algorithm with Implicit Path-Relinking (BRKGA-MP-IPR), a variant of the BRKGA-MP hybridized with an implicit path-relinking local search procedure. By operating over the unit hypercube $[0, 1]^n$, such path-relinking mechanism leverages the population representation of the BRKGA and thus provides complete independence between the local search procedure and the problem definition and implementation, contrasting with traditional path-relinking procedures that are tied to the problem structure. The BRKGA-MP-IPR obtained better results than the BRKGA-MP-IPR. So, incorporating this implicit path-relinking into the NS-BRKGA is another promising line for further research.

One of the main drawbacks of the BRKGA and its variants is the large number of parameters that need to be tuned in order to achieve a good performance. Some works in the literature attempt to solve this problem by using an adaptive strategy, such as the Adaptive Biased Random-Key Genetic Algorithm using Q-Learning Method (BRKGA-QL) [Chaves e Lorena, 2021]. Hence, another interesting research line is the incorporation of such adaptive strategies into the NS-BRKGA framework.

References

Acciarini, G., Izzo, D., e Mooij, E. (2020). Mhaco: a multi-objective hypervolume-based ant colony optimizer for space trajectory optimization. In *2020 IEEE Congress on Evolutionary*

Computation (CEC), p. 1–8. IEEE.

Andrade, C. E., Toso, R. F., Gonçalves, J. F., e Resende, M. G. (2021). The Multi-Parent Biased Random-Key Genetic Algorithm with Implicit Path-Relinking and its real-world applications. *European Journal of Operational Research*, 289(1):17–30. ISSN 0377-2217.

Bean, J. C. (1994). Genetic Algorithms and Random Keys for Sequencing and Optimization. *ORSA Journal on Computing*, 6(2):154–160.

Biscani, F. e Izzo, D. (2020). A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, 5(53):2338.

Bresina, J. L. (1996). Heuristic-biased stochastic sampling. In *AAAI/IAAI, Vol. 1*, p. 271–278.

Cerqueus, A., Przybylski, A., e Gandibleux, X. (2015). Surrogate upper bound sets for bi-objective bi-dimensional binary knapsack problems. *European Journal of Operational Research*, 244(2): 417–433. ISSN 0377-2217.

Chagas, J. B., Blank, J., Wagner, M., Souza, M. J., e 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.

Chaves, A. A. e Lorena, L. H. N. (2021). An Adaptive and Near Parameter-Free BRKGA Using Q-Learning Method. In *2021 IEEE Congress on Evolutionary Computation (CEC)*, p. 2331–2338. IEEE.

Damm, R. d. B. e Ronconi, D. P. (2021). A Multi-objective Biased Random-Key Genetic Algorithm for Service Technician Routing and Scheduling Problem. In Mes, M., Lalla-Ruiz, E., e Voß, S., editors, *International Conference on Computational Logistics*, p. 471–486, Cham. Springer, Springer International Publishing. ISBN 978-3-030-87672-2.

Dantzig, G. B. (1957). Discrete-Variable Extremum Problems. *Operations Research*, 5(2):266–288.

Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., USA. ISBN 047187339X.

Deb, K. (2014). Multi-objective Optimization. In Burke, E. K. e Kendall, G., editors, *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, p. 403–449. Springer US, Boston, MA. ISBN 978-1-4614-6940-7.

Deb, K., Pratap, A., Agarwal, S., e Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Garey, M. R. e Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-completeness*, volume 174 of *Mathematical Sciences Series*. W. H. Freeman. ISBN 9780716710448.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., USA, 1st edition. ISBN 0201157675.

Gonçalves, J. F. e Resende, M. G. (2011). Biased Random-Key Genetic Algorithms for Combinatorial Optimization. *Journal of Heuristics*, 17(5):487–525.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA, USA.

Kacem, I., Hammadi, S., e Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. *Mathematics and Computers in Simulation*, 60(3-5):245–276. ISSN 0378-4754. Intelligent Forecasting, Fault Diagnosis, Scheduling, and Control.

Karatas, M. e Yakıcı, E. (2018). An iterative solution approach to a multi-objective facility location problem. *Applied Soft Computing*, 62:272–287. ISSN 1568-4946.

Kellerer, H., Pferschy, U., e Pisinger, D. (2004). *Multidimensional Knapsack Problems*, p. 235–283. Springer, Berlin, Heidelberg. ISBN 978-3-540-24777-7.

Lacevic, B. e Amaldi, E. (2011). Ectropy of diversity measures for populations in Euclidean space. *Information Sciences*, 181(11):2316–2339. ISSN 0020-0255.

Li, H. e Zhang, Q. (2008). Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation*, 13(2):284–302. ISSN 1941-0026.

Li, X. (2003). A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. In Cantú-Paz, E., Foster, J. A., Deb, K., Davis, L. D., Roy, R., O'Reilly, U.-M., Beyer, H.-G., Standish, R., Kendall, G., Wilson, S., Harman, M., Wegener, J., Dasgupta, D., Potter, M. A., Schultz, A. C., Dowsland, K. A., Jonoska, N., e Miller, J., editors, *Genetic and Evolutionary Computation Conference — GECCO 2003*, p. 37–48, Berlin, Heidelberg. Springer, Springer. ISBN 978-3-540-45105-1.

Li, X. (2004). Better Spread and Convergence: Particle Swarm Multiobjective Optimization Using the Maximin Fitness Function. In Deb, K., editor, *Genetic and Evolutionary Computation – GECCO 2004*, p. 117–128, Berlin, Heidelberg. Springer, Springer Berlin Heidelberg. ISBN 978-3-540-24854-5.

Lucena, M. L., Andrade, C. E., Resende, M. G., e Miyazawa, F. K. (2014). Some extensions of biased random-key genetic algorithms. In *Proceedings of the 46th Brazilian Symposium of Operational Research*, p. 1–12.

Lust, T. e Teghem, J. (2010). Two-phase pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics*, 16(3):475–510.

Mahdavi, M., Fesanghary, M., e Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, 188(2):1567–1579. ISSN 0096-3003.

Martello, S. e Toth, P. (1990). *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, Inc., New York, NY, USA. ISBN 0-471-92420-2.

Mavrotas, G., Florios, K., e Figueira, J. R. (2015). An improved version of a core based algorithm for the multi-objective multi-dimensional knapsack problem: A computational study and comparison with meta-heuristics. *Applied Mathematics and Computation*, 270:25–43. ISSN 0096-3003.

https://proceedings.science/p/157504?lang=pt-br

Metaxiotis, K. e Liagkouras, K. (2012). Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review. *Expert Systems with Applications*, 39(14):11685–11698. ISSN 0957-4174.

Nowak, K., Märtens, M., e Izzo, D. (2014). Empirical Performance of the Approximation of the Least Hypervolume Contributor. In Bartz-Beielstein, T., Branke, J., Filipič, B., e Smith, J., editors, *International Conference on Parallel Problem Solving From Nature – PPSN XIII*, p. 662–671, Cham. Springer International Publishing. ISBN 978-3-319-10762-2.

Peng, W., Zhang, Q., e Li, H. (2009). Comparison between MOEA/D and NSGA-II on the Multi-Objective Travelling Salesman Problem. In Goh, C.-K., Ong, Y.-S., e Tan, K. C., editors, *Multi-Objective Memetic Algorithms*, p. 309–324. Springer, Berlin, Heidelberg. ISBN 978-3-540-88051-6.

Riquelme, N., Von Lücken, C., e Baran, B. (2015). Performance metrics in multi-objective optimization. In *2015 Latin American Computing Conference (CLEI)*, p. 1–11. IEEE.

Samanlioglu, F., Ferrell, W. G., e Kurz, M. E. (2008). A memetic random-key genetic algorithm for a symmetric multi-objective traveling salesman problem. *Computers & Industrial Engineering*, 55(2):439–449. ISSN 0360-8352.

Shah, R. e Reed, P. (2011). Comparative analysis of multiobjective evolutionary algorithms for random and correlated instances of multiobjective d-dimensional knapsack problems. *European Journal of Operational Research*, 211(3):466–479. ISSN 0377-2217.

Shang, K., Ishibuchi, H., He, L., e Pang, L. M. (2021). A Survey on the Hypervolume Indicator in Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 25 (1):1–20. ISSN 1941-0026.

Spears, W. M. e De Jong, K. D. (1995). On the virtues of parameterized uniform crossover. Technical report, Naval Research Lab Washington DC.

Tangpattanakul, P., Jozefowiez, N., e Lopez, P. (2015). Biased Random Key Genetic Algorithm for Multi-user Earth Observation Scheduling. In Fidanova, S., editor, *Recent Advances in Computational Optimization: Results of the Workshop on Computational Optimization WCO 2013*, p. 143–160. Springer International Publishing, Cham. ISBN 978-3-319-12631-9.

Yan, Z., Zhang, L., Kang, L., e Lin, G. (2003). A New MOEA for Multi-objective TSP and Its Convergence Property Analysis. In Fonseca, C. M., Fleming, P. J., Zitzler, E., Thiele, L., e Deb, K., editors, *Evolutionary Multi-Criterion Optimization*, p. 342–354, Berlin, Heidelberg. Springer, Springer. ISBN 978-3-540-36970-7.

Zhang, Q. e Li, H. (2007). MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731. ISSN 1941-0026.

Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P. N., e Zhang, Q. (2011). Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*, 1(1):32–49. ISSN 2210-6502.

Zitzler, E. e Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271.

https://proceedings.science/p/157504?lang=pt-br