Estimating attraction basin sizes of combinatorial optimization problems

Anne Elorza, Leticia Hernando, Alexander Mendiburu, Jose A. Lozano

Abstract
Given a particular instance of a combinatorial optimization problem, the knowledge about the attraction basin sizes can help to analyze the difficulty encountered by local search algorithms while solving it. As calculating these sizes exhaustively is computationally intractable, we focus on methods for their estimation. The accuracy of some of these estimation methods depends on the way in which the sample of solutions of the search space is chosen. In this paper, we propose a novel sampling method, which incorporates the knowledge obtained by the already explored solutions into the sampling strategy. So, in contrast to those that already exist, our method can adapt its behavior to the characteristics of the particular attraction basin. We apply our proposal to a number of instances of three famous problems: the quadratic assignment problem, the linear ordering problem and the permutation flow shop scheduling problem. We consider permutation sizes $n = 10$ and $n = 12$ and three different neighborhoods: adjacent swap, 2-exchange and insert, and observe that the new method generally outperforms those that already exist.

Keywords Combinatorial optimization problems · Neighborhoods · Attraction basins · Local optima · Local search algorithms · Estimators

1 Introduction

Algorithms based on local search are efficient methods which are used to solve hard combinatorial optimization problems (COPs). These algorithms structure the search space defining a neighborhood, which provokes different properties that affect the performance of the algorithms: the landscape properties. Therefore, in the last few years, many authors have already noticed that the first step in the development of algorithms based on local search is the analysis of these properties [1–3,7,16,17,19,22,23,25–27,34–37,39,40]. Among these features, the two that have mainly attracted the attention of the research community are the local optima and their attraction basins (sets composed by all the initial solutions such that, when applying the algorithm, the same local optimum is obtained).

On the one hand, even though the number of local optima is not directly related to the difficulty of solving a problem [20], it can help in the analysis of its complexity. Therefore, many authors have tried to estimate or bound this number of local optima. These estimations have been conducted for problems when considering different neighborhoods [1,2,11]. Moreover, as it is known that the properties could vary for distinct instances of the same problem, methods for estimating the number of local optima for specific instances have also been developed [6,8–10,12,26,28]. One of the conclusions of these studies was that the accuracy of these methods is notably affected by the variance of the attraction basin sizes of the

Anne Elorza, anne.elorza@ehu.eus
Leticia Hernando, leticia.hernando@ehu.eus
Alexander Mendiburu, alexander.mendiburu@ehu.eus
Jose A. Lozano, joseantonio.lozano@ehu.eus

Intelligent Systems Group,
University of the Basque Country UPV/EHU,
20018 San Sebastián, Spain

local optima. In general, the more uniform the attraction basin sizes are, the better is the prediction. However, there are methods that are able to provide good estimations for instances where the variance of the attraction basin sizes is extremely large [12]. Looking at these methods, we can observe that their estimations rely on the concept of sample coverage. The sample coverage is the sum of the proportions of the sizes of the attraction basins of the local optima observed in the sample over the size of the search space. In other words, it measures the proportion of the search space occupied by the attraction basins found in the sample. Thus, the size of the attraction basins of the local optima plays an essential role when trying to estimate the number of local optima of an instance.

On the other hand, the difficulty encountered by local search algorithms has also been proved to be directly related with the size of the attraction basins of the global optima. This is due to the fact that, when taking uniformly at random an initial solution from the search space, the probability that the local search algorithm finishes at a local optimum is proportional to the relative size (with respect to the size of the search space) of its attraction basin [1,6,18]. Moreover, it has been proved for different problems that, on average, the better a local optimum is in terms of fitness, the larger is the size of its attraction basin [3,21,24,25,34–37].

We could say that the attraction basin sizes of the local optima are useful to understand the difficulty found by local search algorithms when trying to reach them. However, as there is no known method that calculates, in polynomial time, the attraction basin sizes, we must focus on methods that estimate these sizes. Commonly, the proportion of an attraction basin size over the size of the search space has been estimated as the proportion of the number of solutions in a sample that belong to the attraction basin over the sample size [14,34]. However, under this method it is supposed that there are no more local optima in the search space except just those encountered in the sample. In [13], two methods for estimating the attraction basin sizes were proposed. Once it is known that a solution is a local optimum, the first method (UM) consisted of taking solutions uniformly at random from the whole search space, and checking if they belong to its attraction basin. In the second method (DM), the search space was divided in different subsets, which corresponded to the sets of solutions at different distances from the local optima. A different number of samples were assigned to the different subsets. The way of choosing these sample sizes according to the distance to the local optimum was called the sampling strategy. The sampling strategy, then, could help in the estimation, or it could disorientate it. Moreover, the accuracy of DM using a specific sampling strategy could be different under distinct neighborhoods.

The aim of this paper is to propose a novel method to improve the already existing sampling strategies of DM. Our sampling method consists of two steps. Firstly, an initial sampling is carried out, using prefixed values for the sample sizes at different distances, as implemented in [13]. After this initial sampling, we obtain an initial estimation of the number of solutions belonging to the attraction basin at different distances. In the second part of the process, we continue sampling, but with a distinct strategy. At each step of the algorithm, we take advantage of the samples which have already been explored and our decision on where to sample next depends on this information. So, every time we need to extract a new sample, we calculate the expected improvement in our final estimation depending on where (at what distance) we sample. Then, the next sample is extracted at the distance which maximizes the expected improvement. So as to test the performance of our new method, we work with a number of instances of the quadratic assignment problem, the linear ordering problem and the permutation flow shop scheduling problem. Three neighborhoods are considered: the adjacent swap, the 2-exchange and the insert neighborhoods.

The rest of the paper is organized as follows. In Sect. 2, we provide some relevant definitions, and the estimation methods are explained in detail in Sect. 3. In Sect. 4, we report our experimental design and results, comparing our new sampling strategy with those that already exist, with problem sizes of $n = 10$ and $n = 12$. Finally, in Sect. 5, the main conclusions are drawn.

2 Definitions

2.1 Combinatorial optimization problem

In general, a combinatorial optimization problem (COP) consists of finding the optimal solutions of a function

$$\begin{aligned} f\colon \Omega &\longrightarrow \mathbb{R} \\ \pi &\longmapsto f(\pi) \end{aligned}$$

where the search space, Ω, is a finite or countable infinite set.

2.2 Quadratic assignment problem

The quadratic assignment problem (QAP) is a COP that consists of allocating a set of facilities to a set of locations, with a cost function associated with the distance and the flow between the facilities. The objective is to assign each facility to a location such that the total cost is minimized. Specifically, we are given two $n \times n$ input matrices with real values $\mathbf{H} = [h_{ij}]$ and $\mathbf{D} = [d_{kl}]$, where h_{ij} is the flow between facility i and facility j, and d_{kl} is the distance between location k and location l. Given n facilities, the solution of the QAP is codified as a permutation $\pi = (\pi(1)\pi(2)\cdots\pi(n))$ where each $\pi(i)$ $(i = 1, \ldots, n)$ represents the facility that is allocated to the ith location. Thus, the fitness of the permutation is given by the following objective function:

$$F(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{n} h_{\pi(i)\pi(j)} \cdot d_{ij}.$$

2.3 Linear ordering problem

Given a matrix $B = [b_{ij}]_{n \times n}$ of numerical entries, the linear ordering problem (LOP) consists of finding a simultaneous permutation π of the rows and columns of B, such that the sum of the entries above the main diagonal is maximized (or equivalently, the sum of the entries below the main diagonal is minimized). The equation below formalizes the LOP function:

$$F(\pi) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} b_{\pi(i)\pi(j)},$$

where $\pi(i)$ $(\pi(j))$ denotes the index of the row (column) ranked at position ij in the solution π.

2.4 Permutation flow shop scheduling problem

In the case of the permutation flow shop scheduling problem (PFSP), n jobs have to be scheduled on m machines in such a way that a criterion is minimized. A job consists of m operations, and the jth operation $(j = 1, \ldots, m)$ of each job must be processed on machine j for a given specific processing time without interruption. The processing times are fixed nonnegative values, and every job is available at time zero. At a given time, a job can start on jth machine when its $(j-1)$th operation has finished on the machine $(j-1)$, and machine j is idle.

The makespan is the total length of the schedule and, traditionally, has been the criterion to be optimized in the PFSP. However, recently, total flow time (TFT) has captured the attention of the scientific community since it is more meaningful for the current industry, and thus, this criterion will be used in this work. The following formula expresses mathematically the concept of TFT for a permutation π of jobs, where $c_{\pi(i),m}$ stands for the completion time of job $\pi(i)$ $(i = 1, \ldots, n)$ at machine m:

$$F(\pi) = \sum_{i=1}^{n} c_{\pi(i),m}.$$

Being $p_{\pi(i),j}$ the processing time required by job $\pi(i)$ on machine j, the completion time of job $\pi(i)$ on machine j can be recursively calculated as:

$$c_{\pi(i),j} = \begin{cases} p_{\pi(i),j} & i = j = 1 \\ p_{\pi(i),j} + c_{\pi(i)-1,j} & i > 1,\ j = 1 \\ p_{\pi(i),j} + c_{\pi(i),j-1} & i = 1,\ j > 1 \\ p_{\pi(i),j} + \max\{c_{\pi(i)-1,j}, c_{\pi(i),j-1}\} & i > 1,\ j > 1 \end{cases}.$$

In these three problems, the QAP, the LOP and the PFSP, the search space is the whole space of permutations of size n, so its size is $n!$.

2.5 Neighborhood

A neighborhood $\mathcal{N}$ in a search space Ω is a mapping that assigns, to each solution $\pi \in \Omega$, a non-empty set of neighboring solutions $\mathcal{N}(\pi)$:

$$\begin{aligned} N\colon \Omega &\longrightarrow P(\Omega)\setminus\emptyset \\ \pi &\longmapsto \mathcal{N}(\pi) \end{aligned},$$

where $P(\Omega)$ is the powerset of Ω.

Adding the concept of neighborhood to the instance of a COP, we define a landscape as the triple $(f, \Omega, \mathcal{N})$.

Three examples of the most commonly used neighborhoods in the space of permutations are given by the adjacent swap, the 2-exchange and the insert operators [4,7,31,32]. The adjacent swap neighborhood ($\mathcal{N}_A$) considers two solutions as neighbors if one is generated by swapping two adjacent elements of the other. The number of adjacent swap neighbors of a permutation of size n is $n-1$. For example, in the space of permutations of size $n = 4$, the set of adjacent swap neighbors of the permutation $\pi = (1234)$ is:

$$\mathcal{N}_A(1234) = \{(2134), (1324), (1243)\}.$$

The swap or 2-exchange neighborhood ($\mathcal{N}_S$) considers that two solutions are neighbors if one is generated by swapping two elements of the other, not necessarily adjacent. Under this neighborhood, a solution has $n(n-1)/2$ neighbors. Taking the same permutation $\pi = (1234)$ as in the previous case, the set formed by its neighbors under the 2-exchange neighborhood is:

$$\mathcal{N}_S(1234) = \{(2134), (3214), (4231), (1324), (1432), (1243)\}.$$

In the case of the insert neighborhood ($\mathcal{N}_I$), two solutions are neighbors if one is the result of moving an item of the other to a different position. The number of neighbors of a solution under the insert neighborhood is $n(n-1)-(n-1) = (n-1)^2$. Following the same example, the set composed of the insert neighbors of the permutation $\pi = (1234)$ is the following:

$$\mathcal{N}_I(1234) = \{(2134), (2314), (2341), (1324), (1342), (3124) \\ (1243), (4123), (1423)\}.$$

We say that two permutations π_1 and π_2 are at distance i if, starting from π_1, and moving from neighboring to neighboring solutions, the length of the shortest path to reach π_2 is i. Particularly, two neighboring permutations are at distance one. Under the adjacent swap neighborhood, the maximum distance between two permutations is $n(n-1)/2$, while for both the 2-exchange and the insert it is $n-1$.

2.6 Local optimum and attraction basin

Assuming a minimization problem, a solution $\pi^* \in \Omega$ is a local optimum (local minimum) under a neighborhood $\mathcal{N}$ if

$$f(\pi^*) \leq f(\pi), \ \forall \pi \in \mathcal{N}(\pi^*).$$

Each local optimum π^* is associated with its attraction basin $\mathcal{B}(\pi^*)$. That is, the set is composed of all the solutions which lead to the local optimum π^*, after applying a local search algorithm to them. We denote by $\mathcal{H}$ the operator that associates, to each solution π, the local optimum obtained after applying the algorithm. Different definitions could be given for the attraction basin depending on the nature of the operator $\mathcal{H}$ (see for example [38,40] for stochastic operators). We work with a deterministic $\mathcal{H}$, so the attraction basin of a local optimum, $\mathcal{B}(\pi^*)$, is the set that can be defined in the following way:

$$\mathcal{B}(\pi^*) = \left\{\pi \in \Omega \mid \mathcal{H}(\pi) = \pi^*\right\}.$$

The attraction basin of a local optimum depends on the algorithm used. Furthermore, when using a deterministic algorithm, an important property is derived from the concept of attraction basins of the local optima: They define a partition of Ω.

2.7 Deterministic best-improvement local search algorithm

We work with a deterministic best-improvement local search algorithm to solve the instances. The specific steps that the algorithm follows are detailed in Algorithm 1. It is important to notice that the neighbors are evaluated in a specific order, so that, in the case of two neighbors having the same function value, the algorithm will always choose the first encountered.

Algorithm 1 Deterministic Best-Improvement Local Search Algorithm

```
Choose an initial solution π ∈ Ω
repeat
  π* = π
  for each σ_i ∈ N(π*) do
    if f(σ_i) < f(π) then
      π = σ_i
    end if
  end for
until π = π*
```

3 Estimation of attraction basin size

Our new method is based on the DM estimator presented in [13]. In order to estimate the attraction basin size of a local optimum π^*, we select a number of solutions and then apply Algorithm 1 to them to check whether they belong to the attraction basin. These will constitute our sample, from which we will infer our estimation of the attraction basin size. The DM method is distance-based, which means that the sampling is not carried out uniformly at random in the whole search space. Contrarily, the search space is divided in different subsets, according to the distance to π^*. So, having this division of the search space, one has to decide how the samples are going to be distributed over the various subdivisions. To establish this distribution, we introduce a new sampling strategy, which addresses some of the limitations of those proposed in [13]. Our new method consists of two phases: an initial sampling and what we have named the dynamic sampling.

3.1 Initial sampling

In this part of the sampling process, we obtain initial estimations for the number of permutations which belong to the attraction basin, proceeding similarly to [13]. As stated before, this method does not directly take a random sample from the whole search space Ω. Instead, given a local optimum π^*, we choose the solutions from different subsets of Ω related to π^*. That is, we consider the different subsets $D_i = \{\pi_1^i, \pi_2^i, \ldots, \pi_{|D_i|}^i\} \subseteq \Omega$ that are composed of those solutions at distance i from the local optimum π^*, where $i = 1, \ldots, d_{max}$ and d_{max} denotes the maximum distance between two permutations.

Notice that any permutation in $\Omega \backslash \{\pi^*\}$ should belong to one, and just one, of these subsets D_i. That is:

$$D_i \cap D_j = \emptyset, \forall i \neq j$$

$$\bigcup_{i=1}^{d_{max}} D_i \cup \{\pi^*\} = \Omega.$$

So, given the local optimum π^*, we take samples $S_1, S_2, \ldots,$ of uniformly random solutions at distances $1, 2, \ldots,$ respectively, from π^*:

$$S_1 = \{\pi_1^1, \pi_2^1, \ldots, \pi_{M_1}^1\} \subseteq D_1$$
$$S_2 = \{\pi_1^2, \pi_2^2, \ldots, \pi_{M_2}^2\} \subseteq D_2$$
$$\vdots$$

$$S_i = \{\pi_1^i, \pi_2^i, \ldots, \pi_{M_i}^i\} \subseteq D_i$$

$$\vdots$$

$$S_{d_{max}} = \{\pi_1^{d_{max}}, \pi_2^{d_{max}}, \ldots, \pi_{M_{d_{max}}}^{d_{max}}\} \subseteq D_{d_{max}}.$$

We use the methods described in [15] to obtain these uniformly random solutions π_j^i for the different distances. In order to estimate the attraction basin size of π^*, one works with the different subsets D_i independently. That is, we record the number of solutions that belong to the attraction basin of π^* in each sample set S_i.

In this initial sampling, we set beforehand the number of solutions which will be sampled at each distance. $M_1, M_2, \ldots, M_{d_{max}}$ denote the sizes of the sample sets. Thus, we extract M_1 samples at distance 1, M_2 at distance 2, etc. The choice of these sizes is up to the user. In our case, we will establish them according to the strategies proposed in [13] for the DM. In brief, we will be using three different configurations:

1. Equal sample sizes for each distance (ES): $M_i = M_j$ $\forall i, j$.
2. Sample sizes proportional to the number of permutations at each distance (SP): $M_i \propto |D_i|$.
3. Sample sizes decreasing as the distance increases (SD): $M_i \propto \dfrac{1}{i}$ and $M_i = 0, i \geq d_{max}/2$.

So, we extract samples according to the given distribution of initial sample sizes and, then, we record the number of these samples which belong to the attraction basin at each of the distances. Algorithm 2 details this process. The input of the algorithm is the sample sizes $M_1, M_2, \ldots, M_{d_{max}}$. On the other hand, the values k_i denote the number of samples at distance i that belong to the attraction basin.

The last part of the algorithm has a correction in M_i and k_i whenever the following two conditions are satisfied simultaneously: There is a distance i which has an associated $k_i = 0$ and another distance $j > i$ such that $k_j > 0$. This means that none of the samples extracted at distance i belong to the attraction basin, but we have found samples at distance j inside the attraction basin. If this happens, even though we have not detected any solution of D_i belonging to the attraction basin, we know that there must exist at least one. The reason is that any solution of D_j belonging to the attraction basin is connected with the local optimum following the path established by the best-improvement local search algorithm, and this path has permutations belonging to any distance $l < j$. So, whenever this situation occurs, we can be sure that, at distance i, there must be permutations belonging to the attraction basin, and a correction is applied to k_i by imposing $k_i = 1$ and $M_i = M_i + 1$.

Algorithm 2 Algorithm that provides the initial estimations of the number of solutions at the different distances inside the attraction basin of the local optimum π^*.

Input: $\mathbf{M} = \{M_1, \ldots, M_{d_{max}}\}$
Initialize $(k_1, k_2, \ldots, k_{d_{max}}) = (0, 0, \ldots, 0)$
for $dist = 1 \rightarrow d_{max}$ **do**
  for $j = 1 \rightarrow M_{dist}$ **do**
    take a random permutation $\sigma_j \in D_{dist}$
    $\sigma = \mathcal{H}(\sigma_j)$
    if $\sigma == \pi^*$ **then**
      $k_{dist}++$
    end if
  end for
end for
for $j = 1 \rightarrow d_{max}$ **do**
  if $k_j = 0$ & $(\exists z > j$ s.t. $k_z > 0)$ **then**
    $k_j = 1$
    M_j++
  end if
end for
Output: $\mathbf{k} = (k_1, k_2, \ldots, k_{d_{max}})$, $\mathbf{M} = \{M_1, \ldots, M_{d_{max}}\}$

3.2 Dynamic sampling

The aim of this second part of the method is to continue sampling, but with a different strategy. The way of proceeding with the sampling is decided in light of the information of the already sampled solutions. As opposed to the methods used in [13], the user does not have to establish the sample sizes for each distance beforehand. On the contrary, the user sets a single size T, which is the total number of samples which will be extracted, in this second phase. Thus, the algorithm decides, every time it has to extract a new sample, which distance it should sample next to obtain the most valuable information.

Our problem shares certain similarity with the multi-armed bandit problem [29], in which a gambler plays a slot machine with multiple levers, each of them which has a different reward distribution (which is unknown), and aims at maximizing the total payout obtained after playing a number of times. For this purpose, the player faces the dilemma between exploration and exploitation or, in other words, whether it is more convenient to pull the arm which has led to greater profit or to explore other levers in case one of them has a better reward. One solution to this problem is the popular Thompson sampling [29], which has partially inspired us. In our case, we are facing the problem of choosing the most convenient distance and, at each step of the algorithm, we will be computing the expected improvement in our estimation depending on where we sample next. Subsequently, we will sample at that distance. This choice, however, is deterministic once we have the information of the previously explored samples, as opposed to Thompson sampling.

Our final goal is to minimize the error in our estimation, that is, to sample in such a way that the difference between

Table 1 Number of permutations of size 10 at different distances from a given solution, according to the neighborhood

2-Exchange		Insert	
dist	#perms	dist	#perms
1	45	1	81
2	870	2	2521
3	9450	3	38,281
4	63,273	4	296,326
5	269,325	5	1,100,902
6	723,680	6	1,604,098
7	1,172,700	7	569,794
8	1,026,576	8	16,795
9	362,880	9	1

Adjacent swap									
dist	#perms	dist	#perms	dist	#perms	dist	#perms		
1	9	10	21,670	19	211,089	28	162,337	37	8095
2	44	11	32,683	20	230,131	29	135,853	38	4489
3	155	12	47,043	21	243,694	30	110,010	39	2298
4	440	13	64,889	22	250,749	31	86,054	40	1068
5	1068	14	86,054	23	250,749	32	64,889	41	440
6	2298	15	110,010	24	243,694	33	47,043	42	155
7	4489	16	135,853	25	230,131	34	32,683	43	44
8	8095	17	162,337	26	211,089	35	21,670	44	9
9	13,640	18	187,959	27	187,959	36	13,640	45	1

the real size and the estimated one, $\left||\mathcal{B}(\pi^*)| - |\hat{\mathcal{B}}(\pi^*)|\right|$, is minimized. For this purpose, the only thing we can control is, at each step of the algorithm, what distance we should sample next. So, let us assume we have extracted t samples and denote our current estimation by $|\hat{\mathcal{B}}_t(\pi^*)|$ (notice that in the first phase of the algorithm we have already explored some solutions and that, at the beginning of this second phase, we already have an initial estimation). If M_i is the number of solutions explored at distance i, k_i is the number of these explored samples that belong to the attraction basin, and $|D_i|$ is the total number of solutions at distance i from the local optimum π^*, our current estimation of the attraction basin size would be as follows:

$$|\hat{\mathcal{B}}_t(\pi^*)| = 1 + \frac{k_1}{M_1} \cdot |D_1| + \frac{k_2}{M_2} \cdot |D_2| + \cdots + \frac{k_{d_{max}}}{M_{d_{max}}} \cdot |D_{d_{max}}|.$$

If we sample at distance i, our new estimation, which we denote by $|\hat{\mathcal{B}}^i_{t+1}(\pi^*)|$, would be as follows:

$$1 + \frac{k_1}{M_1} \cdot |D_1| + \cdots + \frac{k'_i}{M_i + 1} \cdot |D_i| + \cdots + \frac{k_{d_{max}}}{M_{d_{max}}} \cdot |D_{d_{max}}|,$$

where k'_i can have a value of k_i or $k_i + 1$, depending on whether the solution sampled in the last step belongs to the attraction basin or not. We approach the problem of deciding what distance we should sample in the following way: We choose the distance that on average provokes the highest change in the estimation, i.e., the distance that currently has the highest influence in the global estimation (note that the estimation of $|\mathcal{B}(\pi^*)|$ depends on the estimation of each proportion k_i/M_i). Therefore, the expected change in our estimation depends on the distance i that we sample and can be computed with the following expression:

$$E\left[\left||\hat{\mathcal{B}}^i_{t+1}(\pi^*)| - |\hat{\mathcal{B}}_t(\pi^*)|\right|\right].$$

This is the expected difference between our current estimation and the new estimation after checking a new solution, and provides a measure of how much the estimation would change with more sampling. So sampling at the distance i_{max} that maximizes this expression will be our approach to minimizing our final error.

Theorem 1 *The expected influence of sampling at distance i can be estimated using the following close formula:*

$$\frac{2|D_i| \cdot (M_i - k_i) \cdot k_i}{M_i^2 (M_i + 1)},$$

Table 2 Number of permutations of size 12 at different distances from a given solution, according to the neighborhood

2-Exchange		Insert	
dist	#perms	dist	#perms
1	66	1	121
2	1925	2	5941
3	32,670	3	153,341
4	357,423	4	2,250,887
5	2,637,558	5	18,943,343
6	13,339,535	6	87,116,283
7	45,995,730	7	192,422,979
8	105,258,076	8	153,315,999
9	150,917,976	9	24,584,693
10	120,543,840	10	208,011
11	39,916,800	11	1

Adjacent swap											
dist	#perms	dist	#perms	dist	#perms	dist	#perms	dist	#perms	dist	#perms
1	11	12	330,121	23	10,624,132	34	25,380,120	45	7,097,310	56	113,906
2	65	13	526,581	24	12,604,826	35	24,736,324	46	5,615,807	57	61,997
3	274	14	808,896	25	14,664,752	36	23,697,232	47	4,342,688	58	31,758
4	923	15	1,200,626	26	16,739,858	37	22,311,069	48	3,277,965	59	15,159
5	2640	16	1,726,701	27	18,757,500	38	20,640,357	49	2,411,747	60	6655
6	6655	17	2,411,747	28	20,640,357	39	18,757,500	50	1,726,701	61	2640
7	15,159	18	3,277,965	29	22,311,069	40	16,739,858	51	1,200,626	62	923
8	31,758	19	4,342,688	30	23,697,232	41	14,664,752	52	808,896	63	274
9	61,997	20	5,615,807	31	24,736,324	42	12,604,826	53	526,581	64	65
10	113,906	21	7,097,310	32	25,380,120	43	10,624,132	54	330,121	65	11
11	198,497	22	8,775,209	33	25,598,186	44	8,775,209	55	198,497	66	1

Table 3 Parameter setting used in the experiments of our new method

n	Sample size	QAP						LOP					
		Adjacent		2-Exch.		Insert		Adjacent		2-Exch.		Insert	
		ϵ	r	ϵ	r	ϵ	r	ϵ	r	ϵ	r	ϵ	r
10	2250	0.1	2	0.1	2	0.01	2	0.5	3	0.5	1.5	0.01	1.5
	4500	0.01	2	0.5	2	0.001	2	0.5	2	0.01	3	0.01	1.5
12	13,200	0.5	10	0.1	10	0.01	10	0.5	3	0.5	2	0.01	2.5

n	Sample size	PFSP					
		Adjacent		2-Exch.		Insert	
		ϵ	r	ϵ	r	ϵ	r
10	2250	0.5	2	0.5	1.5	0.001	2.5
	4500	0.5	2	0.1	1.5	0.01	3
12	13,200	0.5	2.5	0.5	1.5	0.5	3

Table 4 Sample distribution for the distance methods with prefixed sample sizes for each distance

Sample size	Adjacent swap			2-Exchange			Insert		
	DM-ES	DM-SP	DM-SD ($i \leq 22$)	DM-ES	DM-SP	DM-SD ($i \leq 4$)	DM-ES	DM-SP	DM-SD ($i \leq 4$)
2250	50	$\left\lceil \frac{\lvert D_i \rvert}{1630} \right\rceil$	$\left\lceil \frac{302}{i} \right\rceil$	250	$\left\lceil \frac{\lvert D_i \rvert}{1616} \right\rceil$	$\left\lceil \frac{1080}{i} \right\rceil$	250	$\left\lceil \frac{\lvert D_i \rvert}{1617} \right\rceil$	$\left\lceil \frac{1080}{i} \right\rceil$
4500	100	$\left\lceil \frac{\lvert D_i \rvert}{811} \right\rceil$	$\left\lceil \frac{302}{i} \right\rceil$	500	$\left\lceil \frac{\lvert D_i \rvert}{807} \right\rceil$	$\left\lceil \frac{2160}{i} \right\rceil$	500	$\left\lceil \frac{\lvert D_i \rvert}{807.3} \right\rceil$	$\left\lceil \frac{2160}{i} \right\rceil$

Table 5 Comparison of the average relative errors (and variances) of the estimations of the attraction basin sizes of the QAP with different methods for $n = 10$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent swap				
$M = 2250$				
Inst1	0.210 (0.027)	1.029 (1.498)	0.165 (0.018)	**0.140 (0.009)**
Inst2	0.186 (0.075)	1.360 (6.642)	0.139 (0.086)	**0.099 (0.017)**
Inst3	0.192 (0.039)	1.175 (3.616)	0.105 (0.012)	**0.077 (0.006)**
Inst4	0.245 (0.034)	0.825 (0.651)	**0.116 (0.015)**	0.132 (0.010)
Inst5	0.196 (0.048)	0.890 (1.048)	0.134 (0.018)	**0.128 (0.007)**
Inst6	0.341 (0.136)	1.308 (2.252)	0.324 (0.214)	**0.202 (0.014)**
$M = 4500$				
Inst1	0.137 (0.012)	1.308 (3.096)	0.103 (0.007)	**0.087 (0.004)**
Inst2	0.119 (0.010)	0.995 (2.421)	0.085 (0.007)	**0.057 (0.002)**
Inst3	0.135 (0.018)	0.878 (0.919)	0.068 (0.005)	**0.061 (0.004)**
Inst4	0.164 (0.020)	1.482 (3.338)	0.115 (0.005)	**0.092 (0.005)**
Inst5	0.120 (0.012)	1.064 (3.637)	0.099 (0.006)	**0.080 (0.004)**
Inst6	0.262 (0.100)	1.754 (3.188)	0.226 (0.066)	**0.191 (0.013)**
2-Exchange				
$M = 2250$				
Inst1	**0.151 (0.014)**	0.186 (0.044)	0.796 (0.905)	0.163 (0.018)
Inst2	**0.145 (0.014)**	0.268 (0.122)	0.805 (4.644)	0.147 (0.011)
Inst3	**0.176 (0.016)**	0.356 (0.142)	0.459 (0.632)	0.198 (0.016)
Inst4	0.192 (0.026)	0.226 (0.070)	0.833 (1.748)	**0.182 (0.020)**
Inst5	0.163 (0.020)	0.514 (2.010)	0.741 (2.448)	**0.159 (0.014)**
Inst6	0.136 (0.014)	**0.111 (0.010)**	1.223 (3.069)	**0.111 (0.010)**
$M = 4500$				
Inst1	0.132 (0.011)	**0.109 (0.011)**	0.839 (1.820)	0.093 (0.006)
Inst2	0.121 (0.009)	0.151 (0.034)	0.681 (1.219)	**0.093 (0.004)**
Inst3	**0.137 (0.019)**	0.256 (0.086)	0.671 (8.210)	0.154 (0.012)
Inst4	**0.129 (0.013)**	0.178 (0.042)	1.302 (8.620)	0.130 (0.011)
Inst5	**0.103 (0.009)**	0.299 (0.252)	0.904 (2.395)	**0.103 (0.008)**
Inst6	0.080 (0.004)	**0.075 (0.005)**	1.888 (47.258)	0.079 (0.005)
Insert				
$M = 2250$				
Inst1	0.358 (0.105)	0.693 (0.493)	0.944 (14.115)	0.309 (0.048)
Inst2	0.434 (0.752)	1.073 (1.082)	**0.341 (0.154)**	0.404 (0.089)
Inst3	0.457 (0.516)	1.587 (6.529)	**0.310 (0.071)**	0.341 (0.052)
Inst4	0.477 (0.131)	1.285 (0.743)	**0.439 (0.099)**	0.438 (0.069)
Inst5	0.497 (0.268)	1.241 (1.842)	**0.283 (0.036)**	0.324 (0.07)
Inst6	0.268 (0.108)	0.650 (2.556)	0.503 (3.372)	**0.322 (0.2)**
$M = 4500$				
Inst1	0.282 (0.083)	0.817 (1.591)	0.495 (3.230)	**0.234 (0.031)**
Inst2	0.311 (0.050)	0.831 (0.548)	**0.304 (0.042)**	0.320 (0.113)
Inst3	0.270 (0.051)	1.065 (0.885)	**0.263 (0.037)**	0.272 (0.059)
Inst4	0.424 (0.265)	0.855 (0.292)	**0.344 (0.038)**	0.351 (0.036)
Inst5	0.296 (0.131)	0.844 (1.028)	0.279 (0.042)	**0.240 (0.030)**
Inst6	0.251 (0.096)	0.479 (1.459)	0.649 (11.673)	**0.222 (0.060)**

Bold values indicate lower errors (on average) than other estimating methods

where M_i is the number of solutions explored at distance i, k_i is the number of these explored samples that belong to the attraction basin, and $|D_i|$ is the total number of solutions at distance i from the local optimum π^*.

Proof The estimated change in our estimation can be computed as

$$E\left[\left||\hat{\mathcal{B}}^i_{t+1}(\pi^*)| - |\hat{\mathcal{B}}_t(\pi^*)|\right|\right].$$

If we denote the estimated number of samples inside the attraction basin at distance i in instant t by N_i and in instant $t+1$ by N_i', then we can rewrite the expected improvement:

$$E\left[\left|1+\sum_{k\neq i} N_k + N_i' - \left(1+\sum_{k=1}^{d_{max}} N_k\right)\right|\right] = E\left[|N_i - N_i'|\right].$$

According to our sampling, $N_i = k_i/M_i \cdot |D_i| = p_i \cdot |D_i|$, where p_i denotes the estimated proportion of samples inside the attraction basin at distance i. On the other hand, we cannot estimate N_i'. However, we know that N_i' can take either one of the following values:

$$|D_i| \cdot \frac{k_i+1}{M_i+1}$$

if the new sampled solution belongs to the attraction basin, or

$$|D_i| \cdot \frac{k_i}{M_i+1}$$

if it belongs to a different one. In addition, we can estimate the probability of sampling a solution inside the attraction basin at distance i as p_i. Hence,

$$\begin{aligned}
E\left[|N_i - N_i'|\right] &\approx p_i \cdot \left||D_i|\cdot\frac{k_i}{M_i} - |D_i|\cdot\frac{k_i+1}{M_i+1}\right| \\
&+ (1-p_i)\cdot\left||D_i|\cdot\frac{k_i}{M_i} - |D_i|\frac{k_i}{M_i+1}\right| \\
&= |D_i|\left(p_i\cdot\left|\frac{k_i}{M_i} - \frac{k_i+1}{M_i+1}\right| + (1-p_i)\cdot\left|\frac{k_i}{M_i} - \frac{k_i}{M_i+1}\right|\right) \\
&= |D_i|\left(\frac{k_i}{M_i}\cdot\left|\frac{k_i-M_i}{M_i(M_i+1)}\right| + \frac{M_i-k_i}{M_i}\cdot\left|\frac{k_i}{M_i(M_i+1)}\right|\right) \\
&= \frac{2|D_i|(M_i-k_i)k_i}{M_i^2(M_i+1)}.
\end{aligned}$$

□

Once a dynamic sampling strategy has been established which exploits the information obtained step by step by the algorithm, there are still some aspects which deserve further comment. A problem which needs to be addressed is what happens whenever $k_i = 0$ or $k_i = M_i$, which occurs if none of the permutations checked at distance i belong to the attraction basin or, conversely, if all of them belong to it. In this case, the expression in Theorem 1 holds the value 0, implying that we will stop sampling at distance i. This undesired behavior has its root in estimating the real proportion as p_i when computing the expected improvement. If $k_i = 0$, the problem is that, by setting $p_i = 0$, we are assuming that since we have not found any solution that belongs to the attraction basin at distance i, there must not be any solution at that distance. However, there may exist solutions belonging to the attraction basin which we have not found. So, if this situation occurred, our estimation would be biased, because, regardless of how many samples T we extracted, we would never detect the permutations inside the attraction basin which are at distance i. The case where $k_i = M_i$ is analogous. Hence, this will be corrected by assuming that, whenever $k_i = 0$ or $k_i = M_i$, the real proportion is slightly different than the one we are estimating. So we set $p_i = (k_i + \epsilon)/(M_i + 2\epsilon)$, where $\epsilon > 0$ is a small variation. In this case, the expression of Theorem 1 becomes

$$|D_i| \frac{\epsilon}{(M_i + 2\epsilon)(M_i + 1)}.$$

Lastly, the probability of finding permutations inside an attraction basin decreases as the distance to the local optimum increases. Thus, the attraction basin does not necessarily contain permutations at the furthest distances from the local optimum. So, sampling at those distances may be not as useful as sampling at the nearest ones. Accordingly, to avoid the misuse of our resources, the ϵ value used when $k_i = 0$ will be decreasing with the distance. If i_0 is the first index fulfilling the condition $k_{i_0} = 0$ and r is a decreasing factor, then the expression of Theorem 1 will be

$$|D_i| \frac{\epsilon/(r^{i-i_0})}{(M_i + 2\epsilon/(r^{i-i_0}))(M_i + 1)}.$$

We have summarized this second part of the estimation method in Algorithm 3, which should be executed after Algorithm 2. The final estimation is computed by dividing, for each distance, the number of samples inside the attraction basin, k_i, by the sample size, M_i, and multiplying it by the total number of permutations that exist in each subset D_i ($|D_i|$). Thus, we obtain the estimated number of solutions inside the attraction basin at the different distances. So, the sum of these quantities plus one (π^* itself is in its attraction basin and has not been considered in any subset) is the resultant attraction basin size of the local optimum π^*.

Algorithm 3 Method to estimate the size of the attraction basin of a local optimum π^*.

Inputs: $T, \epsilon, r, \mathbf{k} = (k_1, \ldots, k_{d_{max}}), \mathbf{M} = \{M_1, \ldots, M_{d_{max}}\}$
for $j = 1 \to T$ **do**
  set q to the last index s.t. $k_q > 0$
  set $f\_obj_{dist} = \frac{2|D_{dist}| \cdot (M_{dist} - k_{dist}) \cdot k_{dist}}{M_{dist}^2 (M_{dist} + 1)}$, $\forall dist = 1, \ldots, q$ s.t. $k_{dist} \neq M_{dist}$
  set $f\_obj_{dist} = \frac{|D_{dist}| \cdot \epsilon}{(M_{dist} + 2\epsilon)(M_{dist} + 1)}$, $\forall dist = 1, \ldots, q$, s.t. $k_{dist} = M_{dist}$
  set $f\_obj_{dist} = \frac{|D_{dist}| \cdot \epsilon/(r^{dist-(q+1)})}{(M_{dist} + 2\epsilon/(r^{dist-(q+1)}))(M_{dist} + 1)}$, $\forall dist = q+1, \ldots, d_{max}$
  $d = \arg\max_{dist} \{f\_obj_{dist}\}$
  take a random permutation $\sigma_j \in D_d$
  M_d ++
  $\sigma = \mathcal{H}(\sigma_j)$
  if $\sigma = \pi^*$ **then**
    k_d ++
    for $l = 1 \to (d-1)$ **do**
      if $k_l = 0$ **then**
        $k_l = 1$
        M_l ++
      end if
    end for
  end if
end for
$|\hat{\mathcal{B}}(\pi^*)| = 1$
for $j = 1 \to d_{max}$ **do**
  $|\hat{\mathcal{B}}(\pi^*)| = |\hat{\mathcal{B}}(\pi^*)| + \frac{k_j}{M_j} \cdot |D_j|$
end for
Output: $|\hat{\mathcal{B}}(\pi^*)|$

4 Experiments

4.1 Experimental design

The QAP, LOP and PFSP instances used in the experiments have been taken from the QAPLIB [5], the xLOLIB [30] and the Taillard's Benchmark [33], respectively. In each of the problems, we have chosen 6 instances for which the number of local optima and their attraction basins have been exhaustively calculated according to the adjacent swap ($\mathcal{N}_A$), the 2-exchange ($\mathcal{N}_S$) and the insert ($\mathcal{N}_I$) neighborhoods. That is, Algorithm 1 has been applied to each solution of the search space, for the three problems and neighborhoods. As this implies a high computational cost, the size of the original instances has been reduced in order to work with permutation sizes of 10 and 12, so that the experimentation is computationally affordable: $|\Omega_{10}| = 10! \approx 3.63 \times 10^6$ and $|\Omega_{12}| = 12! \approx 4.79 \times 10^8$. Tables 1 and 2 show the number of permutations at different distances depending on the

Table 6 Comparison of the average relative errors (and variances) of the estimations of the attraction basin sizes of the LOP with different methods for $n = 10$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent swap				
$M = 2250$				
Inst1	0.288 (0.071)	1.174 (4.491)	0.303 (0.075)	**0.217 (0.020)**
Inst2	0.562 (0.907)	1.91 (31.138)	0.423 (0.18)	**0.349 (0.061)**
Inst3	0.370 (0.180)	1.054 (1.633)	0.262 (0.052)	**0.218 (0.019)**
Inst4	0.392 (0.143)	1.401 (4.060)	0.252 (0.056)	**0.206 (0.014)**
Inst5	0.551 (0.418)	2.553 (84.411)	0.316 (0.075)	**0.267 (0.031)**
Inst6	0.528 (0.368)	1.690 (9.923)	0.467 (0.512)	**0.264 (0.026)**
$M = 4500$				
Inst1	0.264 (0.076)	1.173 (3.403)	0.166 (0.016)	**0.136 (0.012)**
Inst2	0.378 (0.131)	1.879 (12.036)	0.387 (0.273)	**0.269 (0.049)**
Inst3	0.241 (0.038)	2.119 (14.497)	0.208 (0.028)	**0.171 (0.013)**
Inst4	0.234 (0.031)	2.078 (18.396)	0.186 (0.033)	**0.144 (0.008)**
Inst5	0.273 (0.074)	0.990 (0.485)	0.228 (0.038)	**0.205 (0.020)**
Inst6	0.318 (0.078)	1.733 (10.662)	0.331 (0.180)	**0.199 (0.017)**
2-Exchange				
$M = 2250$				
Inst1	0.061 (0.004)	**0.053 (0.002)**	1.635 (3.213)	0.065 (0.003)
Inst2	0.513 (0.291)	0.95 (0.942)	**0.397 (0.049)**	0.416 (0.083)
Inst3	0.163 (0.02)	**0.147 (0.013)**	1.403 (7.791)	0.183 (0.02)
Inst4	0.125 (0.008)	**0.090 (0.005)**	1.309 (4.333)	0.126 (0.011)
Inst5	0.192 (0.027)	**0.144 (0.016)**	1.308 (17.649)	0.171 (0.018)
Inst6	0.146 (0.018)	0.105 (0.018)	1.255 (4.579)	**0.102 (0.009)**
$M = 4500$				
Inst1	0.046 (0.001)	**0.034 (0.001)**	1.641 (3.623)	0.046 (0.002)
Inst2	0.322 (0.088)	0.529 (0.258)	0.446 (0.04)	**0.302 (0.032)**
Inst3	0.128 (0.01)	**0.101 (0.006)**	3.514 (110.997)	0.127 (0.009)
Inst4	0.082 (0.006)	**0.075 (0.005)**	1.932 (9.43)	**0.075 (0.004)**
Inst5	**0.103 (0.006)**	0.109 (0.007)	1.464 (18.51)	0.120 (0.009)
Inst6	**0.070 (0.005)**	0.086 (0.011)	1.062 (1.029)	0.096 (0.007)
Insert				
$M = 2250$				
Inst1	0.147 (0.013)	0.144 (0.012)	1.125 (1.223)	**0.143 (0.009)**
Inst2	0.339 (0.047)	**0.319 (0.027)**	0.900 (1.059)	0.330 (0.031)
Inst3	0.444 (0.023)	**0.431 (0.023)**	1.167 (2.388)	0.442 (0.023)
Inst4	0.175 (0.013)	**0.169 (0.011)**	1.181 (2.279)	0.171 (0.011)
Inst5	0.366 (0.068)	**0.284 (0.048)**	1.645 (21.734)	0.299 (0.044)
Inst6	0.154 (0.014)	**0.123 (0.011)**	1.150 (1.359)	0.139 (0.011)
$M = 4500$				
Inst1	0.148 (0.011)	**0.134 (0.009)**	1.216 (1.527)	0.142 (0.008)
Inst2	0.334 (0.029)	**0.322 (0.026)**	1.171 (3.902)	0.332 (0.029)
Inst3	0.444 (0.023)	**0.435 (0.023)**	0.894 (0.365)	0.438 (0.023)
Inst4	0.178 (0.011)	**0.167 (0.009)**	1.071 (0.684)	0.177 (0.008)
Inst5	0.328 (0.048)	0.354 (0.042)	2.416 (79.265)	**0.319 (0.034)**
Inst6	0.131 (0.012)	**0.122 (0.012)**	1.289 (2.155)	0.137 (0.012)

Bold values indicate lower errors (on average) than other estimating methods

Table 7 Comparison of the average relative errors (and variances) of the estimations of the attraction basin sizes of the PFSP with different methods for $n = 10$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent swap				
$M = 2250$				
Inst1	0.403 (0.263)	1.778 (6.584)	0.255 (0.066)	**0.182 (0.014)**
Inst2	0.35 (0.210)	1.649 (5.940)	0.345 (0.438)	**0.229 (0.080)**
Inst3	0.446 (0.756)	2.273 (11.598)	0.370 (0.477)	**0.268 (0.098)**
Inst4	0.465 (0.394)	1.616 (6.156)	0.325 (0.085)	**0.228 (0.026)**
Inst5	0.458 (0.429)	1.466 (4.453)	0.544 (1.806)	**0.286 (0.038)**
Inst6	0.708 (0.757)	2.910 (33.510)	0.599 (0.512)	**0.358 (0.103)**
$M = 4500$				
Inst1	0.211 (0.044)	1.393 (1.827)	0.192 (0.040)	**0.143 (0.010)**
Inst2	0.277 (0.119)	2.208 (14.421)	0.204 (0.058)	**0.187 (0.024)**
Inst3	0.323 (0.189)	1.682 (2.626)	0.269 (0.099)	**0.189 (0.018)**
Inst4	0.332 (0.504)	2.000 (7.178)	0.279 (0.092)	**0.205 (0.034)**
Inst5	0.329 (0.152)	2.157 (15.163)	0.330 (0.248)	**0.243 (0.061)**
Inst6	0.424 (0.293)	2.422 (12.415)	0.386 (0.209)	**0.248 (0.035)**
2-Exchange				
$M = 2250$				
Inst1	0.169 (0.023)	**0.147 (0.019)**	0.983 (3.364)	0.154 (0.013)
Inst2	0.125 (0.009)	**0.101 (0.017)**	0.942 (1.226)	0.125 (0.013)
Inst3	0.206 (0.055)	**0.194 (0.055)**	0.839 (1.710)	0.201 (0.021)
Inst4	0.197 (0.229)	0.390 (1.119)	1.15 (3.046)	**0.155 (0.023)**
Inst5	0.185 (0.025)	**0.134 (0.020)**	1.416 (7.114)	0.191 (0.029)
Inst6	0.116 (0.009)	0.101 (0.011)	1.046 (1.126)	**0.095 (0.007)**
$M = 4500$				
Inst1	0.113 (0.011)	0.114 (0.008)	2.657 (117.965)	**0.104 (0.007)**
Inst2	0.114 (0.018)	**0.087 (0.008)**	1.554 (12.009)	0.094 (0.006)
Inst3	**0.129 (0.012)**	0.133 (0.019)	1.045 (5.505)	0.135 (0.010)
Inst4	0.104 (0.010)	0.206 (0.202)	1.422 (8.115)	**0.102 (0.010)**
Inst5	0.124 (0.012)	**0.109 (0.012)**	2.084 (29.721)	0.118 (0.009)
Inst6	0.081 (0.007)	**0.076 (0.005)**	1.263 (3.740)	0.084 (0.008)
Insert				
$M = 2250$				
Inst1	0.331 (0.040)	**0.306 (0.028)**	1.275 (12.954)	0.307 (0.030)
Inst2	**0.248 (0.043)**	0.253 (0.034)	0.866 (4.100)	0.251 (0.045)
Inst3	0.315 (0.111)	0.339 (0.119)	0.920 (1.662)	**0.312 (0.076)**
Inst4	0.217 (0.040)	**0.186 (0.041)**	1.116 (3.864)	0.189 (0.042)
Inst5	0.210 (0.041)	**0.190 (0.047)**	1.168 (6.034)	0.192 (0.025)
Inst6	0.167 (0.018)	**0.112 (0.008)**	1.271 (20.318)	0.131 (0.011)
$M = 4500$				
Inst1	**0.309 (0.043)**	0.318 (0.025)	1.228 (8.955)	0.310 (0.023)
Inst2	0.245 (0.036)	0.232 (0.028)	1.412 (22.779)	**0.230 (0.031)**
Inst3	0.289 (0.064)	**0.285 (0.058)**	1.181 (4.147)	0.304 (0.074)
Inst4	0.202 (0.037)	0.189 (0.037)	1.114 (4.021)	**0.179 (0.039)**
Inst5	0.199 (0.038)	0.193 (0.036)	2.330 (106.195)	**0.186 (0.032)**
Inst6	0.137 (0.012)	0.113 (0.009)	0.945 (1.764)	**0.111 (0.010)**

Bold values indicate lower errors (on average) than other estimating methods

Table 8 Results of the statistical test for $n = 10$

$M = 2250$						$M = 4500$					
Adjacent		2-Exchange		Insert		Adjacent		2-Exchange		Insert	
Method	Rank value	Method	Rank value	Method	Rank value	Method	Rank value	Method	Rank value	Method	Rank value
QAP											
DM-DS	3.22	**DM-ES**	2.96	**DM-DS**	2.79	**DM-DS**	3.14	**DM-DS**	2.95	**DM-DS**	2.80
DM-SD	2.91	**DM-DS**	2.91	**DM-ES**	2.77	**DM-SD**	3.04	**DM-ES**	2.92	**DM-ES**	2.79
DM-ES	2.58	DM-SP	2.62	**DM-SP**	2.72	DM-ES	2.67	DM-SP	2.74	DM-SD	2.60
DM-SP	1.29	DM-SD	1.50	DM-SD	2.71	DM-SP	1.15	DM-SD	1.38	DM-SP	1.81
LOP											
DM-DS	2.99	**DM-DS**	2.88	**DM-SP**	3.03	**DM-DS**	3.07	**DM-ES**	2.98	**DM-SP**	2.99
DM-SD	2.82	**DM-ES**	2.87	**DM-DS**	2.90	**DM-SD**	2.93	**DM-SP**	2.95	**DM-ES**	2.90
DM-ES	2.64	**DM-SP**	2.87	DM-ES	2.85	DM-ES	2.70	**DM-DS**	2.84	**DM-DS**	2.90
DM-SP	1.54	DM-SD	1.38	DM-SD	1.22	DM-SP	1.29	DM-SD	1.23	DM-SD	1.20
PFSP											
DM-DS	3.11	**DM-SP**	3.05	**DM-SP**	2.95	**DM-DS**	3.06	**DM-SP**	2.97	**DM-DS**	2.98
DM-SD	2.82	**DM-DS**	2.93	**DM-DS**	2.94	DM-SD	2.87	**DM-DS**	2.93	**DM-SP**	2.91
DM-ES	2.69	DM-ES	2.80	**DM-ES**	2.79	DM-ES	2.81	**DM-ES**	2.93	**DM-ES**	2.83
DM-SP	1.38	DM-SD	1.22	DM-SD	1.32	DM-SP	1.26	DM-SD	1.17	DM-SD	1.28

Bold values indicate the method that is ranked first by the statistical test and the ones that show no statistical difference with the best method

three neighborhoods. In the experiments, we have compared our distance method with dynamic sampling (DM-DS) and the original distance method. Regarding the original distance method, which uses prefixed sample sizes, we have used the three strategies described in [13].

For each instance and neighborhood, we have considered 10 local optima and repeated each of the estimations 10 times. The error in a single estimation is computed as the relative error over the real size, that is,

$$\frac{||\hat{\mathcal{B}}(\pi^*)| - |\mathcal{B}(\pi^*)||}{|\mathcal{B}(\pi^*)|}.$$

Our new method requires two parameters, ϵ and r, to be adjusted. These have been set after a series of preliminary experiments (see Online Resource 1, where the error tables of the preliminary experiments are presented), and they vary depending on the dimension n, the type of problem, the sample size ($M = 2250$ or $M = 4500$ when $n = 10$ and $M = 13{,}200$ when $n = 12$) and the type of neighborhood. However, they remain the same for the 6 instances. These preliminary experiments consist in a grid search to find the combination of values of ϵ and r that minimizes the estimation error. The grid search has been carried over 2 local optima chosen uniformly at random from each of the 6 instances (these optima have been excluded from the final results). ϵ has been selected among the values $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5\}$, while r has been selected among $\{1.5, 2, 2.5, 3\}$. Table 3 shows the final parameter setting of our experiments.

Table 9 Comparison of the average relative errors (and variances) of the estimations of attraction basin sizes of the QAP with different methods for $n = 12$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent S.				
$M = 13{,}200$				
Inst1	0.240 (0.088)	1.054 (2.855)	0.162 (0.038)	**0.115 (0.008)**
Inst2	0.150 (0.019)	1.409 (11.911)	0.088 (0.004)	**0.083 (0.003)**
Inst3	0.152 (0.028)	0.994 (1.388)	0.067 (0.006)	**0.056 (0.003)**
Inst4	0.124 (0.011)	0.921 (0.852)	0.070 (0.003)	**0.053 (0.003)**
Inst5	0.137 (0.018)	1.287 (7.707)	0.095 (0.006)	**0.068 (0.003)**
Inst6	0.315 (0.202)	1.693 (11.536)	0.203 (0.035)	**0.162 (0.016)**
2-Exchange				
$M = 13{,}200$				
Inst1	0.085 (0.005)	0.099 (0.013)	1.579 (13.411)	**0.084 (0.006)**
Inst2	**0.136 (0.014)**	0.264 (0.105)	1.925 (44.961)	0.137 (0.009)
Inst3	0.153 (0.025)	0.389 (0.191)	0.632 (2.668)	**0.144 (0.014)**
Inst4	0.159 (0.025)	0.320 (0.138)	0.801 (1.340)	**0.155 (0.019)**
Inst5	0.210 (0.060)	0.299 (0.089)	0.582 (0.481)	**0.167 (0.014)**
Inst6	0.119 (0.011)	**0.104 (0.008)**	0.981 (0.603)	0.106 (0.008)
Insert				
$M = 13{,}200$				
Inst1	0.363 (0.301)	**0.312 (0.261)**	0.375 (0.042)	0.316 (0.027)
Inst2	0.532 (0.635)	1.319 (4.280)	0.367 (0.399)	**0.293 (0.018)**
Inst3	0.334 (0.077)	1.642 (11.804)	0.392 (0.208)	**0.294 (0.028)**
Inst4	0.557 (0.222)	1.124 (2.294)	0.510 (0.354)	**0.437 (0.056)**
Inst5	0.476 (1.892)	1.197 (3.646)	**0.285 (0.053)**	0.308 (0.039)
Inst6	0.432 (0.422)	**0.167 (0.014)**	0.322 (0.028)	0.341 (0.033)

Bold values indicate lower errors (on average) than other estimating methods

Table 10 Comparison of the average relative errors (and variances) of the estimations of attraction basin sizes of the LOP with different methods for $n = 12$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent S.				
$M = 13{,}200$				
Inst1	**0.139 (0.013)**	0.152 (0.025)	0.156 (0.019)	0.195 (0.021)
Inst2	0.484 (0.825)	0.345 (0.215)	0.391 (0.165)	**0.281 (0.052)**
Inst3	0.441 (0.429)	0.533 (0.654)	0.627 (3.189)	**0.372 (0.159)**
Inst4	0.728 (5.667)	0.595 (0.413)	0.605 (0.625)	**0.290 (0.045)**
Inst5	0.292 (0.134)	0.386 (0.211)	0.344 (0.220)	**0.272 (0.037)**
Inst6	0.440 (0.404)	**0.254 (0.061)**	0.300 (0.227)	0.265 (0.053)
2-Exchange				
$M = 13{,}200$				
Inst1	**0.186 (0.078)**	3.006 (195.797)	2.010 (68.036)	0.209 (0.032)
Inst2	0.331 (2.587)	3.066 (86.330)	2.324 (10.400)	**0.069 (0.003)**
Inst3	0.608 (2.652)	1.705 (68.296)	0.774 (0.009)	**0.226 (0.035)**
Inst4	0.287 (0.782)	2.787 (69.616)	2.528 (26.020)	**0.053 (0.002)**
Inst5	0.852 (37.188)	1.312 (6.145)	2.522 (112.578)	**0.132 (0.014)**
Inst6	**0.094 (0.008)**	4.204 (463.418)	4.545 (331.268)	0.144 (0.017)
Insert				
$M = 13{,}200$				
Inst1	**0.112 (0.005)**	0.950 (0.866)	0.801 (0.002)	0.148 (0.009)
Inst2	**0.191 (0.017)**	0.724 (0.552)	0.724 (0.439)	0.201 (0.027)
Inst3	**0.117 (0.008)**	3.167 (76.837)	0.731 (0.029)	0.154 (0.016)
Inst4	**0.140 (0.012)**	1.621 (9.035)	1.977 (15.643)	0.146 (0.016)
Inst5	0.119 (0.017)	0.815 (0.747)	1.145 (5.875)	**0.106 (0.009)**
Inst6	0.931 (1.452)	1.114 (3.575)	1.352 (6.310)	**0.301 (0.003)**

Bold values indicate lower errors (on average) than other estimating methods

4.2 Comparison of sampling strategies for a permutation size $n = 10$

We have computed the errors in the estimations for two different total sample sizes: $M = 2250$ and $M = 4500$. In the case of the original distance-based method, the sampling strategies are the ones described in [13]. Table 4 shows the sample distribution used in each of these three cases, whereas Tables 5, 6 and 7 report the performance of the three original distance-based methods (DM), which can be classified as static methods, and our new DM-DS for the QAP, the LOP and the PFSP, respectively. The errors are computed as the average of the relative errors for each of the instances, when considering 10 local optima and with 10 repetitions.

Regarding our DM-DS, we have used half of the samples for the initial sampling and the remaining half for the dynamic sampling. So, for a total sample size of 2250, 1125 samples would constitute the initial sampling and 1125 the dynamic sampling. Besides, the initial sampling requires prefixed sample sizes for each distance and we have chosen these sizes based on the results for the three static methods. We have selected different initial sampling strategies depending on the type of problem and the neighborhood, in view of the results of the static methods shown in Tables 5, 6 and 7. For this purpose, we have computed the average error of each of the three static methods (DM-ES, DM-SP and DM-SD) over all the instances and have selected the initial sampling strategy of the DM-DS according to the static method which minimizes the average error. This means that, in the case of the QAP (using the results of the static methods of Table 5), for the adjacent swap neighborhood we have used sample sizes which decrease as the distance increases, because the DM-DS is the method which gives the lowest average error. On the contrary, in the case of the 2-exchange and insert neighborhoods, we have used equal sample sizes for each distance, since the DM-ES is the static method having the lowest average error. In the case of the LOP (Table 6) and the PFSP (Table 7), we have also used sample sizes which decrease as the distance increases for the adjacent swap neighborhood and equal sample sizes for each distance for the 2-exchange neighborhood, while for the insert neighborhood, we have used sample sizes proportional to the number of permutations at each distance.

After computing the errors of the estimations, we have conducted a nonparametric Friedman's test with a level of significance $\alpha = 0.05$ to check whether there are statisti-

Table 11 Comparison of the average relative errors (and variances) of the estimations of attraction basin sizes of the PFSP with different methods for $n = 12$

	DM-ES	DM-SP	DM-SD	DM-DS
Adjacent S.				
$M = 13{,}200$				
Inst1	0.312 (0.131)	0.432 (0.785)	0.254 (0.077)	**0.248 (0.050)**
Inst2	**0.189 (0.047)**	0.308 (0.552)	0.224 (0.039)	0.210 (0.028)
Inst3	0.233 (0.076)	**0.184 (0.027)**	0.294 (0.154)	0.215 (0.020)
Inst4	0.410 (0.746)	0.372 (0.337)	0.371 (0.397)	**0.246 (0.026)**
Inst5	0.492 (0.629)	0.537 (0.783)	0.700 (2.823)	**0.338 (0.100)**
Inst6	0.431 (0.500)	0.269 (0.110)	0.324 (0.314)	**0.231 (0.055)**
2-Exchange				
$M = 13{,}200$				
Inst1	0.310 (0.197)	0.879 (2.909)	0.875 (1.471)	**0.182 (0.020)**
Inst2	0.176 (0.052)	1.472 (18.989)	0.807 (0.311)	**0.149 (0.016)**
Inst3	0.266 (0.085)	1.050 (7.733)	1.967 (92.343)	**0.179 (0.016)**
Inst4	0.455 (1.986)	1.654 (23.013)	1.625 (20.264)	**0.096 (0.013)**
Inst5	0.180 (0.067)	1.695 (21.699)	1.929 (16.669)	**0.084 (0.005)**
Inst6	0.150 (0.023)	0.780 (0.095)	0.787 (0.102)	**0.108 (0.009)**
Insert				
$M = 13{,}200$				
Inst1	0.209 (0.050)	0.816 (0.718)	1.767 (36.316)	**0.202 (0.017)**
Inst2	**0.118 (0.011)**	1.416 (6.510)	3.183 (198.732)	0.161 (0.019)
Inst3	**0.216 (0.026)**	1.708 (31.032)	1.550 (16.641)	0.242 (0.029)
Inst4	**0.081 (0.005)**	1.537 (7.817)	2.132 (13.834)	0.101 (0.009)
Inst5	0.266 (0.089)	1.038 (2.730)	1.053 (3.002)	**0.197 (0.039)**
Inst6	0.075 (0.004)	2.009 (8.667)	1.738 (6.744)	**0.062 (0.003)**

Bold values indicate lower errors (on average) than other estimating methods

Table 12 Results of the statistical test for $n = 12$

Adjacent		2-Exchange		Insert	
Method	Rank value	Method	Rank value	Method	Rank value
QAP					
DM-DS	3.64	**DM-ES**	2.65	**DM-DS**	3.19
DM-SD	3.20	**DM-DS**	2.62	**DM-ES**	3.15
DM-ES	1.61	DM-SD	2.37	DM-SP	1.84
DM-SP	1.54	DM-SP	2.35	DM-SD	1.82
LOP					
DM-SD	2.53	**DM-DS**	3.45	**DM-DS**	3.41
DM-DS	2.51	**DM-ES**	3.41	**DM-ES**	3.36
DM-ES	2.49	DM-SD	1.58	DM-SD	1.63
DM-SP	2.47	DM-SP	1.55	DM-SP	1.60
PFSP					
DM-SP	2.53	**DM-DS**	3.49	**DM-DS**	3.46
DM-DS	2.50	DM-ES	3.27	**DM-ES**	3.38
DM-SD	2.50	DM-SP	1.63	DM-SP	1.59
DM-ES	2.46	DM-SD	1.61	DM-SD	1.57

Bold values indicate the method that is ranked first by the statistical test and the ones that show no statistical difference with the best method

cal differences among the four methods. This test provides a ranking of the methods while also giving an average rank value for each method. These rankings are shown in Table 8, where our method is ranked in the first position in the majority of the cases (a higher rank is associated with a lower error). As we always find statistical differences, we have proceeded with a post hoc test, which carries out all pairwise comparisons. In particular, we have used Holm's procedure (with a level of significance $\alpha = 0.05$). We have used bold font to distinguish the methods which are ranked first or have no statistical difference with the method in the first position. In most of the cases, there is no statistical difference between the method ranked in the first position and the one ranked as second. What is more, in all the cases in which our method does not hold the first position, there is no statistical difference with the best one. However, in the following cases, our method is ranked in the first position and is significantly better than the second one: For the adjacent swap neighborhood and a sample size $M = 2250$, independently of the type of problem, and, in the case of the PFSP, for the same neighborhood and a sample size $M = 4500$.

4.3 Comparison of sampling strategies for a permutation size $n = 12$

We have repeated, for $n = 12$, the experiments conducted for $n = 10$ with a sample size of 13,200. Tables 9, 10 and 11 gather the average of the relative errors for 10 local optima and 10 repetitions, for the QAP, the LOP and the PSFP, when using the distance method with prefixed sample sizes and our dynamic sampling strategy. Regarding the DM-DS, its initial sampling has been established analogously to the case $n = 10$. That is, the initial sampling constitutes half of the total amount of samples and its distribution is decided according to the results of the DM methods with prefixed sample sizes. So, for the QAP, in the case of the adjacent swap and insert neighborhoods, we have used sample sizes which decrease as the distance increases and, for the 2-exchange neighborhood, equal sample sizes for each distance. For the LOP, in the case of the adjacent swap neighborhood, we have used sample sizes proportional to the number of permutations, while we have used equal sample sizes for the other two neighborhoods. In the case of the PFSP, we have used equal sample sizes for the three neighborhoods.

As for $n = 10$, we have conducted the Friedman's test and the Holm's procedure (with $\alpha = 0.05$) to check for statistical differences among the methods. Table 12 shows the rankings of the methods. Similar to the case $n = 10$, in all the cases in which our method does not hold the first position, there is no statistical difference with the best one. There is one case where our method is statistically better than the rest: when we are working with the PFSP under the 2-exchange neighborhood. On the other hand, when we have the LOP and PFSP under the adjacent swap neighborhood, there is no statistical difference among any of the methods.

5 Conclusions and future work

Calculating the attraction basin sizes of an instance of a combinatorial optimization problem can be useful if one wants to measure the complexity of solving an instance with a specific algorithm. Furthermore, the knowledge about the attraction basin sizes helps in the estimation of the number of local optima. Nevertheless, estimating the attraction basin sizes for COPs is a non-trivial task, because of the size of the search space and the complicated neighborhood structure. So, few methods exist which fulfill this purpose.

In [13], two estimators were presented: the UM and the DM. Based on the latter, we have proposed, in this paper, a novel method for estimating the attraction basin size of a local optimum. In contrast to the DM, which uses prefixed sample sizes for each distance, our new algorithm decides its strategy during its execution by analyzing the permutations which have already been sampled. The main advantage of our method is that one does not have to set the sample sizes manually. On the contrary, this is established by the behavior of the algorithm given a specific attraction basin, so that the number of solutions to be explored at each of the distances from the local optimum depends on the information given by the already sampled solutions. The appropriateness of this approach is confirmed by the experimentation, since our new proposal generally has lower errors and, when this does not hold, there is no case in which it performs statistically worse than the original DM.

Future work will consider studying the estimations of higher dimensions. Besides, it could be a good idea to attempt to find the most adequate balance between the initial sampling and the dynamic sampling, since we have only tried an equally sized configuration (that is, half of the samples for the initial sampling and the other half for the dynamic sampling).

Acknowledgements This work has been partially supported by the Research Groups 2013–2018 (IT-609-13) programs (Basque Government) and TIN2016-78365R (Spanish Ministry of Economy, Industry and Competitiveness). Anne Elorza holds a predoctoral grant (ref. PIF17/293) from the University of the Basque Country UPV/EHU.

References

1. Albrecht, A., Lane, P., Steinhofel, K.: Combinatorial landscape analysis for k-SAT instances. In: IEEE Congress on Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence), pp. 2498 –2504 (2008)
2. Albrecht, A., Lane, P., Steinhofel, K.: Analysis of local search landscapes for k-SAT instances. Math. Comput. Sci. **3**(4), 465–488 (2010)
3. Alyahya, K., Rowe, J.E.: Phase transition and landscape properties of the number partitioning problem. In: Evolutionary Computation in Combinatorial Optimisation: 14th European Conference, EvoCOP 2014, Granada, Spain, April 23–25, 2014, Revised Selected Papers, pp. 206–217. Springer, Berlin (2014)
4. Angel, E., Zissimopoulos, V.: On the hardness of the quadratic assignment problem with metaheuristics. J. Heuristics **8**(4), 399–414 (2002)
5. Burkard, R.E., Karisch, S.E., Rendl, F.: QAPLIB—a quadratic assignment problem library. J. Glob. Optim. **10**(4), 391–403 (1997)
6. Caruana, R., Mullin, M.: Estimating the number of local minima in big, nasty search spaces. In: Proceedings of IJCAI-99 Workshop on Statistical Machine Learning for Large-Scale Optimization (1999)
7. Chicano, F., Daolio, F., Ochoa, G., Verel, S., Tomassini, M., Alba, E.: Local optima networks, landscape autocorrelation and heuristic search performance. In: Proceedings of Parallel Problem Solving from Nature, PPSN XII. Lecture Notes in Computer Science, vol. 7492, pp. 337–347. Springer (2012)
8. Eremeev, A.V., Reeves, C.R.: Non-parametric estimation of properties of combinatorial landscapes. In: Cagnoni, S., Gottlieb, J., Hart, E., Middendorf, M., Raidl, G. (eds.) Applications of Evolutionary Computing. Lecture Notes in Computer Science, vol. 2279, pp. 31–40. Springer, Berlin (2002)
9. Eremeev, A.V., Reeves, C.R.: On confidence intervals for the number of local optima. In: Proceedings of EvoWorkshops 2003, pp. 224–235 (2003)

10. Garnier, J., Kallel, L.: How to detect all maxima of a function. In: Kallel, L., Naudts, B., Rogers, A. (eds.) Theoretical Aspects of Evolutionary Computing, pp. 343–370. Springer, London (2001)
11. Grundel, D., Krokhmal, P., Oliveira, C., Pardalos, P.: On the number of local minima for the multidimensional assignment problem. J. Comb. Optim. **13**, 1–18 (2007)
12. Hernando, L., Mendiburu, A., Lozano, J.A.: An evaluation of methods for estimating the number of local optima in combinatorial optimization problems. Evol. Comput. **21**(4), 625–658 (2013)
13. Hernando, L., Mendiburu, A., Lozano, J.A.: Estimating Attraction Basin Sizes, pp. 458–467. Springer, Cham (2016)
14. Hernando, L., Pascual, J.A., Mendiburu, A., Lozano, J.A.: A study on the complexity of TSP instances under the 2-exchange neighbor system. In: IEEE Symposium on Foundations of Computational Intelligence (FOCI), 2011, pp. 15–21 (2011)
15. Irurozki, E., Calvo, B., Lozano, J.A.: An R package for permutations, Mallows and Generalized Mallows models. J. Stat. Softw. **3**(4), 1–16 (2016)
16. Merz, P.: Advanced fitness landscape analysis and the performance of memetic algorithms. Evol. Comput. **12**(3), 303–325 (2004)
17. Merz, P., Freisleben, B.: Memetic algorithms for the travelling salesman problem. Complex Syst. **13**(4), 297–345 (2001)
18. Morgan, R., Gallagher, M.: Using landscape topology to compare continuous metaheuristics: a framework and case study on edas and ridge structure. Evol. Comput. **20**(2), 277–299 (2012)
19. Moser, I., Gheorghita, M., Aleti, A.: Identifying features of fitness landscapes and relating them to problem difficulty. Evol. Comput. **25**(3), 407–437 (2016)
20. Ochoa, G., Chicano, F., Tinós, R., Whitley, D.: Tunnelling crossover networks. In: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, pp. 449–456. ACM (2015)
21. Ochoa, G., Tomassini, M., Verel, S., Darabos, C.: A study of NK landscapes' basins and local optima networks. In: Genetic and Evolutionary Computation Conference, GECCO 2008, Proceedings, Atlanta, GA, USA, July 12–16, 2008, pp. 555–562 (2008)
22. Ochoa, G., Verel, S., Daolio, F., Tomassini, M.: Clustering of local optima in combinatorial fitness landscapes. In: Learning and Intelligent Optimization, vol. 6683, pp. 454–457. Springer, Berlin (2012)
23. Ochoa, G., Verel, S., Daolio, F., Tomassini, M.: Local optima networks: a new model of combinatorial fitness landscapes. In: Richter, H., Engelbrecht, A. (eds.) Recent Advances in the Theory and Application of Fitness Landscapes, Emergence, Complexity and Computation, vol. 6, pp. 233–262. Springer, Berlin (2014)
24. Ochoa, G., Verel, S., Tomassini, M.: First-improvement vs. best-improvement local optima networks of NK landscapes. In: Parallel Problem Solving from Nature, PPSN XI: 11th International Conference, Kraków, Poland, September 11–15, 2010, Proceedings, Part I, pp. 104–113. Springer, Berlin (2010)
25. Prügel-Bennett, A., Tayarani-Najaran, M.H.: Maximum satisfiability: anatomy of the fitness landscape for a hard combinatorial optimization problem. IEEE Trans. Evol. Comput. **16**(3), 319–338 (2012)
26. Reeves, C., Aupetit-Bélaidouni, M.: estimating the number of solutions for SAT problems. In: Yao, X., Burke, E., Lozano, J., Smith, J., Merelo-Guervós, J., Bullinaria, J., Rowe, J., Tino, P., Kabán, A., Schwefel, H.P. (eds.) Parallel Problem Solving from Nature—PPSN VIII. Lecture Notes in Computer Science, vol. 3242, pp. 101–110. Springer, Berlin (2004)
27. Reeves, C.R.: Landscapes, operators and heuristic search. Ann. Oper. Res. **86**, 473–490 (1999)
28. Reeves, C.R.: Statistical properties of combinatorial landscapes: an application to scheduling problems. In: MIC'2001: Proceedings of the 4th Metaheuristic International Conference, pp. 691–695 (2001)
29. Russo, D., Van Roy, B., Kazerouni, A., Osband, I.: A tutorial on Thompson sampling. arXiv preprint arXiv:1707.02038 (2017)
30. Schiavinotto, T., Stützle, T.: The linear ordering problem: Instances, search space analysis and algorithms. J. Math. Model. Algorithms **3**(4), 367–402 (2004)
31. Schiavinotto, T., Stützle, T.: The linear ordering problem: instances, search space analysis and algorithms. J. Math. Model. Algorithms **3**(4), 367–402 (2005)
32. Taillard, E.: Some efficient heuristic methods for the flow shop sequencing problem. Eur. J. Oper. Res. **47**(1), 65–74 (1990)
33. Taillard, E.: Benchmarks for basic scheduling problems. Eur. J. Oper. Res. **64**(2), 278–285 (1993)
34. Tayarani-Najaran, M.H., Prügel-Bennett, A.: On the landscape of combinatorial optimization problems. IEEE Trans. Evol. Comput. **18**(3), 420–434 (2014)
35. Tayarani-Najaran, M.H., Prügel-Bennett, A.: Anatomy of the fitness landscape for dense graph-colouring problem. Swarm Evol. Comput. **22**, 47–65 (2015)
36. Tayarani-Najaran, M.H., Prügel-Bennett, A.: Quadratic assignment problem: a landscape analysis. Evol. Intel. **8**(4), 165–184 (2015)
37. Tomassini, M., Vérel, S., Ochoa, G.: Complex-network analysis of combinatorial spaces: the NK landscape case. Phys. Rev. E **78**(6), 66–114 (2008)
38. Verel, S., Daolio, F., Ochoa, G., Tomassini, M.: Local optima networks with escape edges. In: International Conference on Artificial Evolution (EA-2011), pp. 10–23. Angers, France (2011)
39. Verel, S., Ochoa, G., Tomassini, M.: Local optima networks of NK landscapes with neutrality. IEEE Trans. Evol. Comput. **15**(6), 783–797 (2011)
40. Watson, J.P.: An introduction to fitness landscape analysis and cost models for local search. In: Gendreau, M., Potvin, J-Y. (eds.) Handbook of Metaheuristics, vol. 146, pp. 599–623. Springer, Berlin (2010)