On the symmetry of the Quadratic Assignment Problem through Elementary Landscape Decomposition

Xabier Benavides
xbenavides@bcamath.org
BCAM - Basque Center for Applied Mathematics

Josu Ceberio
josu.ceberio@ehu.eus
UPV/EHU - University of the Basque Country

Leticia Hernando
leticia.hernando@ehu.eus
UPV/EHU - University of the Basque Country

ABSTRACT

When designing meta-heuristic strategies to optimize the quadratic assignment problem (QAP), it is important to take into account the specific characteristics of the instance to be solved. One of the characteristics that has been pointed out as having the potential to affect the performance of optimization algorithms is the symmetry of the distance and flow matrices that form the QAP.

In this paper, we further investigate the impact of the symmetry of the QAP on the performance of meta-heuristic algorithms, focusing on local search based methods. The analysis is carried out using the elementary landscape decomposition (ELD) of the problem under the swap neighborhood. First, we study the number of local optima and the relative contribution of the elementary components on a benchmark composed of different types of instances. Secondly, we propose a specific local search algorithm based on the ELD in order to experimentally validate the effects of the symmetry. The analysis carried out shows that the symmetry of the QAP is a relevant feature that influences both the characteristics of the elementary components and the performance of local search based algorithms.

CCS CONCEPTS

• **Mathematics of computing** → **Combinatorial optimization**;
• **Computing methodologies** → **Discrete space search**;

KEYWORDS

Quadratic Assignment Problem, Elementary Landscapes, Symmetry

ACM Reference Format:
Xabier Benavides, Josu Ceberio, and Leticia Hernando. 2021. On the symmetry of the Quadratic Assignment Problem through Elementary Landscape Decomposition. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3449726.3463191

1 INTRODUCTION

Formalized by Koopmans and Beckmann [36] in 1957, the quadratic assignment problem (QAP) has been a recurring problem in combinatorial optimization due to its known complexity. In fact, in 1976, Sahni and Gonzalez [44] proved its NP-hardness. The QAP was originally proposed as a mathematical model for the location of a set of indivisible economic activities; however, in recent years, it has demonstrated to have many other real world applications such as facility layout design [24], parallel production scheduling [29], backboard wiring [6] or keyboard configuration [8]. Moreover, some major optimization problems can be expressed as particular cases of the QAP, as for example the traveling salesman problem (TSP) [34], the linear ordering problem (LOP) [13] or the DNA fragment assembly problem (DNA-FA) [38]. As a result of this, a number of survey papers and reference books about the QAP have been published over the years [2, 9, 37].

A great number of strategies to solve the QAP have been proposed in the literature. Among them, branch and bound [28] and branch and cut [25] algorithms have proved to be powerful for small problem sizes; however, in large problems, they are no longer viable. As a result, due to the computational limitations, meta-heuristic algorithms [5] postulate as an efficient alternative, though they do not guarantee the optimality of the solutions. Over the years, many different meta-heuristic algorithms have been proposed for the QAP, ranging from local search approaches [4, 49, 53] to population based evolutionary algorithms [1, 23, 27, 40]. Designing new specific algorithms requires a good knowledge of the problem to be solved, so many authors have felt the need to perform extensive prior analyses of the characteristics of the QAP [16, 21, 39, 41].

Nevertheless, due to the inherent complexity of NP-hard problems, it may be difficult to directly study the characteristics of the QAP. In this sense, an alternative strategy can be to decompose the problem into a set of components that provide a framework that eases the analysis [42]. This approach allows us to have a more detailed vision of the problem that can facilitate the detection of useful properties for optimization. Among all the possible decomposition techniques, one available for the QAP is the elementary landscape decomposition (ELD). Proposed by Chicano et al. [19], this decomposition technique allows the QAP to be decomposed as a linear combination of three independent components (elementary landscapes). However, in spite of the growing interest, its impact on the analysis of combinatorial optimization problems and the development of new algorithms have been quite limited. Ceberio et al. [12] used the ELD as a method for the multi-objectivization of the QAP. In that work, a reference to the correlation between the symmetry of the QAP and its ELD decomposition was also highlighted. In fact, the symmetry has been observed to be particularly important in the QAP [22].

In this work, the elementary landscape decomposition is used to better understand the way in which the symmetry of the problem influences the performance of meta-heuristic algorithms, specifically

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07…$15.00
https://doi.org/10.1145/3449726.3463191

local search based methods [48]. For this purpose, we first analyze the components of the decomposition on a benchmark of instances extracted from the QAPLIB library [7]. The analysis focuses on two main issues: the number of local optima and the relative contribution of the components of the decomposition to the structure of the problem. With this analysis, we intend to study the differences between different types of instances according to their symmetry. We build on the obtained results to propose a local search based algorithm that uses the elementary landscape decomposition to efficiently solve the QAP. This algorithm is used to validate the conclusions of the analysis through an experimental study of its performance on a benchmark of instances.

The rest of the paper is organized as follows. In Sections 2 and 3, the quadratic assignment problem and the theory on elementary landscape decomposition are introduced. Section 4 shows the analysis of the QAP through the ELD, and discusses the effects of the symmetry of the instances on the characteristics of the problem. In Section 5, a local search based algorithm that uses the ELD is proposed and an experimental study of its performance is introduced. Finally, general conclusions and ideas for future work are presented in Section 6.

2 QUADRATIC ASSIGNMENT PROBLEM

The *quadratic assignment problem* (QAP) [36] consists of n facilities that need to be assigned to n available locations, taking into account that a facility can only be assigned to one location, and vice versa. d_{ij} is defined as the distance between the location i and location j, and h_{pq} is defined as the work flow between the facility p and the facility q. The goal of the QAP is to find the configuration that minimizes the overall communication costs between facilities, computed by the following objective function:

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} h_{\sigma(i)\sigma(j)} \tag{1}$$

where σ is a permutation of size n and $\sigma(i)$ denotes the facility assigned to the location i. Therefore, any QAP instance is composed of two elements: a distance matrix $D = [d_{ij}]_{n\times n}$ that specifies the distances between locations and a flow matrix $H = [h_{pq}]_{n\times n}$ that denotes the work flows between facilities.

The symmetry of the QAP has been singled out as a relevant feature that might influence the performance of optimization algorithms. Thus, in this work we classify the QAP instances into three different groups: *symmetric* (both the distance and flow matrices are symmetric[1]), *semi-symmetric* (either the distance matrix or the flow matrix is symmetric, but not both), and *asymmetric* (neither of the distance and flow matrices is symmetric).

3 ELEMENTARY LANDSCAPE DECOMPOSITION

A *landscape* of a combinatorial optimization problem [43] is defined as a triplet (Ω, f, N), where Ω is the search space of the problem, $f : \Omega \rightarrow \mathbb{R}$ stands for the objective function that measures the fitness value of the solutions and N denotes a neighborhood function that assigns to each solution $x \in \Omega$ a set of neighboring solutions $N(x) \subset \Omega$. The concepts of local minimum and maximum in a combinatorial optimization problem are defined by the landscape:

- **Local minimum:** Any solution $x \in \Omega$ such that $f(x) \leq f(y)$ for all $y \in N(x)$.
- **Local maximum:** Any solution $x \in \Omega$ such that $f(x) \geq f(y)$ for all $y \in N(x)$.

Thus, landscapes are the basis of local search optimization processes. Among all possible landscapes, there are some that are of particular interest due to their properties: the *elementary landscapes* [45, 51]. These landscapes were discovered by L.K. Grover [31] when the author noticed that the local search procedures of some combinatorial optimization problems could be modeled by a discrete formula similar to the wave equation used in physics. This formula, known as Grover's wave equation, makes it possible to calculate the average value of the objective function f evaluated over all the neighborhood $N(x)$ based on the fitness value of x:

$$\underset{y\in N(x)}{avg}\{f(y)\} = f(x) + \frac{k}{|N(x)|}\big(\bar{f} - f(x)\big) \tag{2}$$

where $\bar{f}$ is the average value of the objective function over the entire search space and k is a characteristic constant that depends on the landscape. Any landscape that satisfies this equation is known to be elementary. More specifically, a landscape is elementary when its objective function f is elementary, that is, when f is an eigenfunction of the Laplacian matrix of the graph induced by Ω and N [45, 51].

Elementary landscapes always satisfy the following [20]:

- $f(x) < \bar{f} \Longrightarrow f(x) < \underset{y\in N(x)}{avg}\{f(y)\} < \bar{f}$
- $f(x) = \bar{f} \Longrightarrow f(x) = \underset{y\in N(x)}{avg}\{f(y)\} = \bar{f}$
- $f(x) > \bar{f} \Longrightarrow f(x) > \underset{y\in N(x)}{avg}\{f(y)\} > \bar{f}$

The above implies that all the local minima have a objective function value that is equal to or less than the average objective function value of the entire search space, while just the opposite happens in the case of the local maxima. These properties ensure that the elementary landscapes have a well known structure, which makes them particularly interesting for dealing with combinatorial optimization problems.

Although not every landscape (Ω, f, N) is elementary, any landscape whose neighborhood is regular ($N(x) = d$ for all $x \in \Omega$) and symmetric ($y \in N(x) \leftrightarrow x \in N(y)$ for all $x, y \in \Omega$) can be decomposed as a linear combination of several elementary landscapes which are known as elementary components of the problem. This process is called *elementary landscape decomposition* (ELD) [19].

3.1 Elementary Landscape Decomposition of the QAP

In the case of the QAP, no neighborhood that produces an elementary landscape is known. However, according to [17], the QAP under the *swap* neighborhood can be decomposed into a combination of three elementary landscapes. In the swap neighborhood, two solutions are neighbors if one can be transformed into the other by exchanging two items of the solution. Therefore, the landscape used to represent the QAP is $L = (S_n, f, N)$, where S_n is the set

[1] With respect to the main diagonal.

of all the permutations of size n, f is the objective function given by Equation (1) and N is the swap neighborhood. In what follows, an explanation of the results developed in [17] is presented in a summarized form. For a more detailed explanation, see [17].

The elementary landscape decomposition of L is done by searching for a set of elementary functions $\{f_1, f_2, f_3, ..., f_m\}$ that form m elementary landscapes together with the search space and the neighborhood function of L. This set of functions is created by the decomposition of the objective function of L, so it must satisfy $f(\sigma) = f_1(\sigma) + f_2(\sigma) + f_3(\sigma) + ... + f_m(\sigma)$ for all $\sigma \in S_n$. In order to facilitate the decomposition, we rewrite Equation (1) as follows:

$$f(\sigma) = \sum_{i,j,p,q=1}^{n} \psi_{i,j,p,q}\varphi_{(i,j)(p,q)}(\sigma) \tag{3}$$

where $\psi_{i,j,p,q} = d_{i,j}h_{p,q}$ and $\varphi_{(i,j)(p,q)}(\sigma) = \delta^{p}_{\sigma(i)}\delta^{q}_{\sigma(j)}$, considering that δ^{b}_{a} is the Kronecker's delta function that returns 1 if $a = b$ and 0 otherwise. In this new formulation, $\psi_{i,j,p,q}$ is the instance-related part while $\varphi_{(i,j)(p,q)}(\sigma)$ is the problem-related part that varies depending on σ. According to [17], f can be decomposed as the sum of three elementary functions:

$$f_1(\sigma) = \sum_{\substack{i,j,p,q=1\\ i\neq j\\ p\neq q}}^{n} \psi_{i,j,p,q}\frac{\phi^{1}_{(i,j)(p,q)}(\sigma)}{2n} \tag{4}$$

$$f_2(\sigma) = \sum_{\substack{i,j,p,q=1\\ i\neq j\\ p\neq q}}^{n} \psi_{i,j,p,q}\frac{\phi^{2}_{(i,j)(p,q)}(\sigma)}{2(n-2)} \tag{5}$$

$$f_3(\sigma) = \sum_{i,p=1}^{n} \psi_{i,i,p,p}\varphi_{(i,i)(p,p)}(\sigma) + \sum_{\substack{i,j,p,q=1\\ i\neq j\\ p\neq q}}^{n} \psi_{i,j,p,q}\frac{\phi^{3}_{(i,j)(p,q)}(\sigma)}{n(n-2)} \tag{6}$$

where $\phi^{1}_{(i,j)(p,q)}$, $\phi^{2}_{(i,j)(p,q)}$ and $\phi^{3}_{(i,j)(p,q)}$ are defined as:

$$\phi^{m}_{(i,j)(p,q)}(\sigma) = \begin{cases} \alpha & \text{if } \sigma(i) = p \wedge \sigma(j) = q \\ \beta & \text{if } \sigma(i) = q \wedge \sigma(j) = p \\ \gamma & \text{if } \sigma(i) = p \oplus \sigma(j) = q \\ \epsilon & \text{if } \sigma(i) = q \oplus \sigma(j) = p \\ \zeta & \text{if } \sigma(i) \neq p, q \wedge \sigma(j) \neq p, q \end{cases} \tag{7}$$

where $1 \leq i, j, p, q \leq n$ and $\alpha, \beta, \gamma, \epsilon, \zeta \in \mathbb{R}$. The operator $\oplus$ stands for the exclusive OR operator. The set of parameters for each of the functions $m = 1, 2, 3$ is:

	α	β	γ	ϵ	ζ
ϕ^1	n-3	1-n	-2	0	-1
ϕ^2	n-3	n-3	0	0	1
ϕ^3	2n-3	1	n-2	0	-1

By definition, $f(\sigma) = f_1(\sigma) + f_2(\sigma) + f_3(\sigma)$ for all $\sigma \in S_n$. The functions f_1, f_2 and f_3 form three elementary landscapes together with the search space and the neighborhood function of L. These three elementary landscapes, denoted as L_1, L_2 and L_3 respectively, are the elementary components of the decomposition of the QAP.

4 ANALYSIS OF THE QAP

In this section we experimentally analyze the characteristics of the elementary landscapes of the decomposition of the QAP on a benchmark of instances. The benchmark is composed of symmetric, semi-symmetric and asymmetric instances so as to find the particularities and influences of each type of QAP problem.

The experiments are divided into two groups. First, we estimate the number of local optima of the elementary landscapes as a measure of their complexity when solving them using local search algorithms. We take into account the plateaus formed by the local optima in order to better understand the ruggedness of the landscapes. This experimentation helps us to decide which components of the decomposition are easier to optimize using local search strategies. Secondly, we quantify the relative contribution of each of the elementary landscapes to both the objective function and the local optima of the overall problem. This experimentation allows us to investigate the relevance of each elementary component when solving QAP instances.

4.1 Benchmark of instances

The experimental framework used in this work consists of the following instances:

- 20 instances extracted from the QAPLIB library [7]: 8 symmetric instances (*chr15a*, *chr20a*, *esc16a*, *esc16b*, *had18*, *had20*, *rou15*, *rou20*), 8 semi-symmetric instances (*lipa20a*, *lipa20b*, *lipa30a*, *lipa30b*, *tai15b*, *tai20b*, *tai25b*, *tai30b*), 4 asymmetric instances (*bur26a*, *bur26b*, *bur26c*, *bur26d*).
- 4 asymmetric instances specifically generated for the experimentation (*xab20a*, *xab20b*, *xab20c*, *xab20d*).

The digits in the instance names indicate the size of the problems. Due to the lack of asymmetric instances in the QAPLIB library, we generate 4 additional asymmetric instances in order to have a more diverse benchmark for the analysis[2]. The instance generation technique used is loosely inspired by the one proposed in [47]:

- **Distance matrix (D):** First, n uniform random points are generated in a 100×100 rectangle, and the euclidean distance $e(i, j)$ between every pair of points $1 \leq i, j \leq n$ is computed. Then, for every $1 \leq i, j \leq n$ such that $i \neq j$, $d_{ij} = v \cdot e(i, j)$ where v is a uniform random value between 0.85 and 1.15. Finally, the matrix is scaled so that $0 \leq d_{ij} \leq 100$ for every $1 \leq i, j \leq n$. All the values are rounded to the closest integer.
- **Flow matrix (H):** For every $1 \leq i, j \leq n$ such that $i \neq j$, a uniform random value x between 0 and 1 is sampled. If $x > sp$, where sp is a parameter that indicates the sparsity of the matrix, h_{ij} is set to a uniform random integer between 0 and 100. Otherwise, $h_{ij} = 0$.

The entries on the main diagonals of both matrices are set to 0. Using this technique, we obtain instances that are formed by two asymmetric matrices: a semi-structured distance matrix and a random flow matrix. In order to enhance the diversity of the

[2]The generated instances are available in https://github.com/XB-Repositories/GECCO-Algorithms/tree/main/Instances.

benchmark, the generated instances are created using different parameter settings: $sp = 0$ (*xab20a*), $sp = 0.25$ (*xab20b*), $sp = 0.5$ (*xab20c*) and $sp = 0.75$ (*xab20d*). The higher the value of sp, the higher the sparsity of the flow matrix.

4.2 Number of local optima

The number of local optima of a combinatorial landscape can be related to the difficulty of finding the global optima using local search algorithms [26, 32]; therefore, it can be used as an indirect measure of complexity. However, it is unfeasible to exhaustively calculate the number of local optima of the benchmark instances due to the size of their search spaces. For that reason, in this study the *ChaoLee2* [14] estimator that is reviewed in [32] has been considered. This estimation technique approximates the total number of local optima from a sample of local optima that is obtained by performing a basic local search starting from M random solutions. The considered local search works under the swap neighborhood and chooses the best solution at each step. From the M initial solutions, r unique local optima $\Theta = \{\sigma_1^*, \sigma_2^*, ..., \sigma_r^*\} \subset S_n$ $(r \leq M)$ are obtained. Based on this sample, the estimator makes a distinction between easy-to-find and hard-to-find optima, which is defined by a δ parameter that indicates the minimum number of times an optimum must be observed to be considered easy-to-find. The number of easy-to-find and hard-to-find optima in the sample is then used to estimate the total number of local optima. For the sake of brevity, we do not explain the exact formula used by *ChaoLee2*, so we refer the interested reader to [32].

According to [32], when all the obtained local optima are different ($r = M$), the *ChaoLee2* method does not work. In these cases, the strategy followed is to take a random sample of solutions, and the proportion of local optima in the sample is used as an estimator of the proportion of local optima in the entire search space. This alternative estimation method is called *CountOptima*.

Until now, we have assumed that all the local optima are independent from each other, so we have purposely put aside some aspects that greatly influence the structure of the landscapes. One of the most important factors is the plateaus formed by local optima [33]. A plateau in a combinatorial landscape is a set of solutions $P \subseteq \Omega$ such that, for every pair of solutions $x, y \in P$, satisfies $f(x) = f(y)$ and there is a path $(x = a_1, a_2, ..., a_k = y)$ such that $a_i \in P$ and $a_{i+1} \in N(a_i)$. A plateau that is formed by multiple local optima can be considered as a unique local optimum when applying local search based algorithms, so a landscape in which most of the local optima are clustered on plateaus may be much less rugged than the estimates suggest.

Thus, in order to validate the obtained results, the local optima found in the sample (Θ) are grouped into plateaus. Given any pair of local optima $\sigma_1^*, \sigma_2^* \in \Theta$, we consider that σ_1^* and σ_2^* are part of the same plateau if $f(\sigma_1^*) = f(\sigma_2^*)$ and there is a path $(\sigma_1^* = a_1^*, a_2^*, .., a_k^* = \sigma_2^*)$ such that $a_i^* \in \Theta$ and $a_{i+1}^* \in N(a_i^*)$. Although considering only the local optima found in the sample speeds up the computations, the results are less precise since we could consider as different plateaus two sets of solutions that are connected by a path of local optima $(b_1^*, b_2^*, ..., b_k^*)$ such that $b_i \notin \Theta$. Nevertheless, this strategy provides an approximate upper bound for the number of plateaus in the sample in a reasonable amount of time.

4.2.1 Results. The number of local optima of each of the elementary landscapes for the benchmark instances are estimated based on a sample of 200.000 local searches per instance and landscape. The cutoff parameter of the *ChaoLee2* estimator is set to $\delta = 10$ according to the recommendations of [15]. When 100% of the encountered local optima appear only once in the sample, the alternative *CountOptima* method is used based on 200.000 random solutions. The obtained results are shown in Table 1.

As can be observed, the *CountOptima* method estimates that 100% of the solutions in the search space of L_1 are local optima in all the symmetric and semi-symmetric benchmark instances. Although this could lead us to believe that L_1 is a very rugged landscape, this is not really the case. According to [50], the objective function of L_1 turns out to be constant ($f_1(\sigma_1) = f_1(\sigma_2)$ for all $\sigma_1, \sigma_2 \in S_n$) when at least one of the matrices that form the instance is symmetric, which explains why every solution in the search space of the symmetric and semi-symmetric instances is a local optimum itself. This affirmation has already been formally demonstrated in [50], but, for the sake of completeness, we provide an alternative mathematical proof in Appendix A.

Table 2 shows the results for the plateaus formed by the local optima found in the samples. As the method used only computes an upper bound for the number of plateaus, when a landscape is constant it may return misleading results (more than 1 plateau). Therefore, to avoid confusion, we omit the L_1 landscape in symmetric and semi-symmetric instances when grouping the local optima.

In asymmetric benchmark instances, L_1 seems to be a very rugged landscape, since the estimated number of local optima and plateaus is much higher than in the rest of the elementary landscapes. Moreover, as not all the solutions in the search space of L_1 are local optima, we also know that in these cases L_1 is not constant. This is the major difference with respect to symmetric and semi-symmetric instances.

The results also show that, independently of the symmetry of the instance, the L_3 landscape is, in general, much less rugged than L_1 and L_2. In fact, in almost all the symmetric and semi-symmetric benchmark instances, L_3 seems to have just 1 plateau formed by local optima. Two exceptions are the *esc16* instances, in which the estimated proportion of local optima in the search space is 100%. However, we suspect that in these cases the objective function of L_3 is constant, just like in the case of L_1.

4.3 Contribution of the elementary components

The second analysis focuses on the study about the relative contribution of the elementary components of the decomposition to the structure of the problem. The aim is to find out the elementary landscapes that have the greatest influence in each type of instance. For this purpose, we first compute the spectral amplitudes [18, 35, 46] of the eigenvalues of f_1, f_2 and f_3 for the benchmark instances. The spectral amplitudes can be considered as a measure of the relative contribution of each of the elementary landscapes to the variance of the general objective function f, and are calculated as:

$$W_i = \frac{\overline{f_i^2} - \overline{f_i}^2}{\overline{f^2} - \overline{f}^2} \tag{8}$$

Table 1: Estimated number of local optima. In the cases where *CountOptima* is used (italic), the results are shown as the estimated percentage of local optima in the search space.

Symmetric Instances

	chr15a	chr20a	esc16a	esc16b	had18	had20	rou15	rou20
L1	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*
L2	72.730	3.628.387	47.831.068	9.999.900.000	81.995	445.134	150.772	6.153.458
L3	4	8	*100,00 %*	*100,00 %*	1.152	768	1	1

Semi-symmetric Instances

	lipa20a	lipa20b	lipa30a	lipa30b	tai15b	tai20b	tai25b	tai30b
L1	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*	*100,00 %*
L2	9.311.635	10.453.678	167.744.885.090	198.743	5.787	28.443	300.217	3.188.513
L3	9.999.900.000	32	*0,00 %*	48	2	1	1	1

Asymmetric Instances

	bur26a	bur26b	bur26c	bur26d	xab20a	xab20b	xab20c	xab20d
L1	5.458.734	999.900.000	6.971.361	1.999.900.000	16.750.465	11.065.476	13.993.626	7.358.769
L2	279.060	1.147.427	145.420	716.890	261.008	240.978	484.125	222.968
L3	13.823	21.324.218	41.400	7.763.110	7	6	16	4

Table 2: Approximate upper bound for the number of plateaus in the samples compared to the number of local optima (in brackets). L_1 is omitted in symmetric and semi-symmetric instances.

Symmetric Instances

	chr15a	chr20a	esc16a	esc16b	had18	had20	rou15	rou20
L1	-	-	-	-	-	-	-	-
L2	41.892 (41.892)	189.658 (189.736)	191.971 (199.581)	199.803 (199.998)	15.533 (34.165)	49.842 (95.928)	61.009 (61.009)	193.591 (193.591)
L3	1 (4)	1 (8)	200.000 (200.000)	200.000 (200.000)	1 (1.152)	1 (768)	1 (1)	1 (1)

Semi-symmetric Instances

	lipa20a	lipa20b	lipa30a	lipa30b	tai15b	tai20b	tai25b	tai30b
L1	-	-	-	-	-	-	-	-
L2	192.030 (192.312)	191.735 (191.739)	199.977 (199.979)	198.743 (198.743)	5.125 (5.125)	17.942 (17.942)	81.067 (81.067)	181.475 (181.475)
L3	199.691 (199.998)	1 (32)	200.000 (200.000)	1 (48)	1 (2)	1 (1)	1 (1)	1 (1)

Asymmetric Instances

	bur26a	bur26b	bur26c	bur26d	xab20a	xab20b	xab20c	xab20d
L1	174.259 (190.492)	199.580 (199.980)	185.442 (195.253)	199.834 (199.990)	198.397 (198.398)	197.194 (197.194)	198.067 (198.067)	195.557 (195.557)
L2	38.555 (83.135)	77.423 (141.690)	22.415 (55.346)	49.067 (126.507)	72.958 (72.960)	67.401 (67.402)	96.562 (96.564)	70.006 (70.011)
L3	2 (13.823)	169.852 (199.058)	6 (40.837)	120.587 (197.439)	7 (7)	6 (6)	16 (16)	4 (4)

where W_i is the spectral amplitude that measures the contribution of L_i. By definition, $W_1 + W_2 + W_3 = 1$.

Once the contribution of each of the elementary components to the overall fitness is quantified, the next step is to study the relationship between the local optima of L and the local optima of the elementary landscapes. In particular, we estimate the percentage of the local optima of L that are also local optima of L_1, L_2 or L_3, for each of the benchmark instances. This measure provides information about the relative contribution of the elementary landscapes to the local optima of the overall problem, which is relevant when working with local search based algorithms.

4.3.1 Results. The exact values of the spectral amplitudes for the benchmark instances are calculated as explained in [18]. The obtained results are shown as heat maps in Figure 1.

As the results show, the L_2 landscape seems to be especially important on symmetric and semi-symmetric benchmark instances. In fact, in almost all the cases, the spectral amplitude of L_2 is 0.7 or higher. On the other hand, the contribution of the L_3 landscape seems to be remarkable on asymmetric benchmark instances, since its spectral amplitude is higher than 0.5 in half of them.

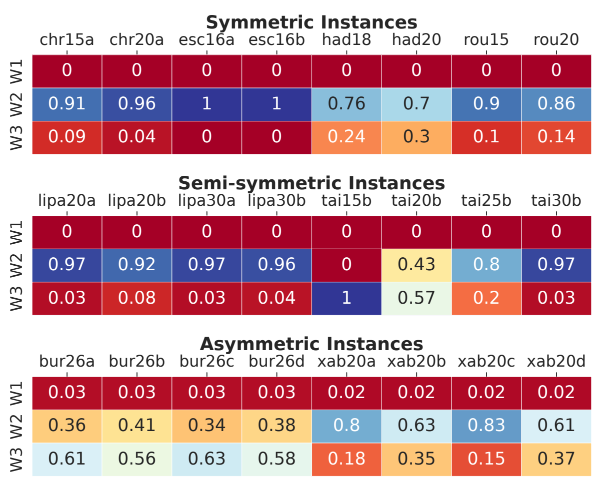


Figure 1: Spectral amplitudes. The red color indicates that the amplitude is closer to 0, while the blue color indicates that the amplitude is closer to 1.

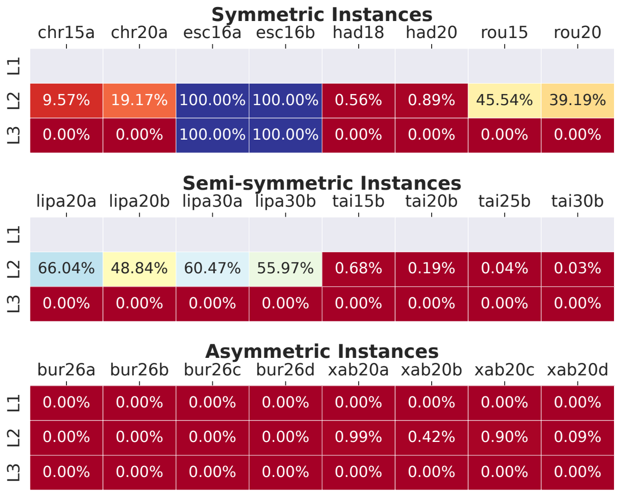


Figure 2: Estimated percentage of the local optima of L that are also local optima of L_1, L_2 or L_3. The red color indicates that the percentage is closer to 0%, while the blue color indicates that the percentage is closer to 100%. L_1 is omitted in symmetric and semi-symmetric instances.

In all cases, the least important landscape is L_1. As we have already explained, the objective function of this landscape is constant in symmetric and semi-symmetric instances ($W_1 = 0$), and its contribution in asymmetric benchmark instances is very limited, with a spectral amplitude that ranges from 0.02 to 0.03.

Next, we estimate the percentage of the local optima of L that are also local optima of L_1, L_2 or L_3 for each of the benchmark instances from the local optima found in the samples of Section 4.2. As the L_1 landscape is entirely composed of local optima in symmetric and semi-symmetric instances (constant objective function), we omit this case in order to avoid confusion. The obtained results are shown as heat maps in Figure 2.

In the view of the results, L_2 shares a significant number of local optima with L in the majority of symmetric and semi-symmetric benchmark instances. However, the percentage of shared local optima varies greatly from one instance to another, ranging from as low as 0.03% to as high as 100.00%. Be that as it may, it seems that the L_2 landscape has the most important influence on the local optima of L in these types of instances. The only exceptions in which the L_3 landscape also shares a high percentage of the local optima with L (100%) are the *esc16* instances, but, as we have already mentioned, we suspect that this is because L_3 is constant.

In asymmetric benchmark instances, on the other hand, none of the elementary landscapes shares a significant number of local optima with L (< 1.00% in all cases). Therefore, in these cases there does not seem to be a single elementary landscape that predominantly influences the local optima of the overall problem.

4.4 Discussion

The experiments carried out show that the symmetry of the instances has an important impact on the difficulty and relative contribution of the elementary landscapes that form the decomposition of the QAP. All these characteristics might be exploited when designing specific meta-heuristic approaches.

We have proven that in symmetric and semi-symmetric instances the objective function of L_1 is constant. Therefore, for optimization purposes these types of instances are composed of just two elementary landscapes. Among the non-constant landscapes, L_2 is generally the elementary landscape that has the greatest relative contribution to both the fitness and the local optima of the problem. Thus, it might be good idea to focus on L_2 when optimizing symmetric and semi-symmetric instances. In asymmetric instances, however, there is not a clearly predominant elementary landscape. This suggests that in these cases all the elementary landscapes should be considered during optimization, especially L_2 and L_3.

As we can see, symmetric and semi-symmetric instances seem to have similar characteristics. This is due to the fact that any semi-symmetric instance can be converted into a symmetric instance without modifying its fitness landscape [22]. Thus, semi-symmetric instances are, in fact, special cases of the symmetric QAP.

In addition to the aforementioned differences, we have also found some interesting characteristics that are common to all the considered types of instances. For example, L_3 has in general fewer local optima (and plateaus formed by local optima) than the rest of the elementary landscapes regardless of the symmetry of the instance, although this feature is especially evident in symmetric and semi-symmetric problems. This characteristic makes L_3 a particularly easy landscape for local search optimization.

5 LOCAL SEARCH BASED ALGORITHM

Based on the analysis made in the previous section, we propose a specific local search based algorithm [48] that uses the elementary landscape decomposition of the problem to efficiently optimize the QAP. Our main goal is to use this algorithm to experimentally verify the conclusions of the previous analysis. To this end, in this section we explain the proposed method and study its performance on a benchmark of instances[3].

5.1 Variable Function Search

One of the main problems of the local search algorithms is that they can get stuck in poor quality local optima. Because of that, when developing local search based algorithms, most of the work consists of finding strategies to efficiently escape from local optima in order to reach better solutions. In this work, we propose a strategy that is based on the elementary landscape decomposition of the QAP. Specifically, the proposed local search based algorithm, called *Variable Function Search* (VFS), consists of the following steps:

1. Starting from a random solution $\sigma \in S_n$, a basic local search is applied. The considered local search works under the swap neighborhood (N) and selects the best solution at each step until a local optimum $\sigma^* \in S_n$ is reached.
2. In order to escape from the local optimum, the swap neighborhood $N(\sigma^*)$ is explored according to the objective functions of the decomposition (f_1,f_2,f_3).

2.1. If the algorithm finds a solution $\sigma' \in N(\sigma^*)$ that is better than σ^* in at least one of the objective functions of

[3]The implemented algorithms are available in https://github.com/XB-Repositories/GECCO-Algorithms/tree/main/MetaHeuristics.

the decomposition, it returns to step 1 with $\sigma = \sigma'$. If there is more than one neighbor solution that satisfies this condition, the algorithm selects the best one among them according to f.

2.2. Otherwise, the algorithm stops and returns the best solution found according to f.

In order to avoid cycles, the VFS uses a tabu list [30] that stores the most recent neighborhood movements. Those movements cannot be undone until they leave the tabu list.

Although the VFS has been designed in the context of the QAP, it can be easily adapted to any problem for which an elementary landscape decomposition is known. This can be done by simply modifying the neighborhood and objective functions.

5.2 Experimental Study

In what follows, an experimental study is carried out in order to check whether the symmetry of the instances affects the performance of the proposed local search based algorithm. To this end, we compare the VFS with a classical *Tabu Search* (TS) [30] on the same benchmark of instances explained in Section 4.

For an instance of size n, the maximum tabu list size is set to n in both algorithms. With respect to the stopping criterion, a number of solution evaluations have been set: $1000n^2$. Taking this into account, each pair algorithm-instance is run 100 times. In Table 3, we show for each case the median of the obtained objective function values and relative errors.

The median results show that the TS is equal to or better than the VFS in all the symmetric benchmark instances. In semi-symmetric and asymmetric benchmark instances, however, both algorithms seem to have a more similar performance. In fact, in these cases the VFS outperforms the TS in a significant number of instances (3 semi-symmetric instances, 4 asymmetric instances). Although we cannot affirm that these differences are exclusively due to the symmetry of the instances, we firmly believe that it is an important factor.

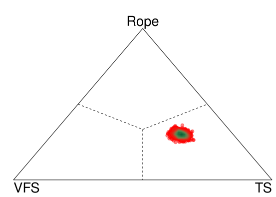
5.3 Statistical Analysis

In order to further compare the VFS and the TS, we carry out a statistical analysis using the Bayesian signed-rank test [3, 10], which is the Bayesian equivalent of the Wilcoxon test [52]. This technique considers the experimental data and computes the expected probability of each algorithm being the best among all the compared methods. The used implementation is available in the *scmamp* R package [11].

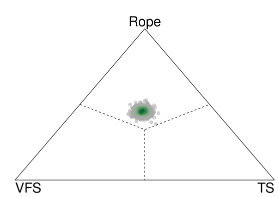
As it is our aim to study the differences between different types of instances, instead of performing one global statistical analysis we carry out three independent analyses, one per instance type (symmetric, semi-symmetric, asymmetric). The experimental data used in the statistical analyses is composed of the relative errors obtained in the experimentation of Section 5.2. The Bayesian signed-rank test requires the definition of the region of practical equivalence (*Rope*), that is, the interval in which the performance of two algorithms is considered equivalent. In this work, we consider that the performance of two algorithms is equivalent if the difference between their relative errors is smaller than 10^{-6}. The results of the performed statistical analyses are shown in Figure 3.

Table 3: Median of the fitness and relative errors obtained in 100 runs of the algorithms. The best algorithm for each of the benchmark instances is highlighted in green. If there is a tie, both algorithms are highlighted in yellow. The best-known solutions for *xab20a, xab20b, xab20c* and *xab20d* are the best solutions found in the experimentation.

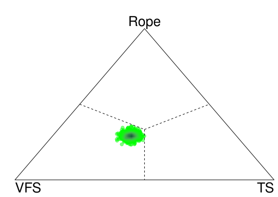
		Best known	VFS		Tabu Search	
			Fitness	Rel. Error	Fitness	Rel. Error
Symmetric	chr15a	*9.896*	10.037	0,014248	**9.936**	**0,004042**
	chr20a	*2.192*	2.404	0,096715	**2.383**	**0,087135**
	esc16a	*68*	70	0,029412	**68**	**0,000000**
	esc16b	*292*	**292**	**0,000000**	**292**	**0,000000**
	had18	*5.358*	**5.370**	**0,002240**	**5.370**	**0,002240**
	had20	*6.922*	6.948	0,003756	**6.941**	**0,002745**
	rou15	*354.210*	**354.210**	**0,000000**	**354.210**	**0,000000**
	rou20	*725.522*	728.258	0,003771	**726.988**	**0,002021**
Semi-Symmetric	lipa20a	*3.683*	**3.683**	**0,000000**	**3.683**	**0,000000**
	lipa20b	*27.076*	**27.076**	**0,000000**	**27.076**	**0,000000**
	lipa30a	*13.178*	**13.178**	**0,000000**	**13.178**	**0,000000**
	lipa30b	*151.426*	**151.426**	**0,000000**	**151.426**	**0,000000**
	tai15b	*51.765.268*	**51.765.268**	**0,000000**	**51.765.268**	**0,000000**
	tai20b	*122.455.319*	**135.121.165**	**0,103432**	135.431.383	0,105966
	tai25b	*344.355.646*	**390.736.718**	**0,134689**	391.656.357,5	0,137360
	tai30b	*637.117.113*	**713.843.365,5**	**0,120427**	716.630.590	0,124802
Asymmetric	bur26a	*5.426.670*	**5.435.393**	**0,001607**	5.435.461	0,001620
	bur26b	*3.817.852*	**3.828.150**	**0,002697**	3.829.348,5	0,003011
	bur26c	*5.426.795*	**5.437.167,5**	**0,001911**	5.437.623	0,001995
	bur26d	*3.821.225*	**3.831.026**	**0,002565**	3.831.146,5	0,002596
	xab20a	*847.472*	851.780	0,005083	**851.150**	**0,004340**
	xab20b	*465.155*	**465.598**	**0,000952**	**465.598**	**0,000952**
	xab20c	*382.155*	383.740	0,004148	**383.679,5**	**0,003989**
	xab20d	*96.815*	**97.787**	**0,010040**	**97.787**	**0,010040**



(a) Symmetric instances.



(b) Semi-symmetric instances.



(c) Asymmetric instances.

Figure 3: Results of the Bayesian signed-rank test for each instance type shown as simplex plots.

In short, the points in the simplex plots represent a sampling of the posterior distribution of the probability of win-lose-tie. That is, the closer a point is to the VFS vertex, the higher the probability that the VFS has a better performance, and vice versa. The same applies for the TS and *Rope* vertices, where the last one indicates

Table 4: Expected probabilities for each of the possibilities of the Bayesian signed-rank tests. The option with the highest probability in each test is highlighted in bold.

	VFS	TS	Rope
Symmetric	0,2021	**0,4978**	0,3001
Semi-symmetric	0,2818	0,2636	**0,4546**
Asymmetric	**0,4085**	0,3013	0,2901

that both algorithms are practically equivalent. The dashed lines delimit the dominance region of each possibility, that is, the area where the highest probability corresponds to its vertex. In order to better visualize the obtained results, the expected probabilities for each possibility (TS-VFS-*Rope*) are shown in Table 4.

As we can see, different situations can be observed depending on the symmetry of the instances. In symmetric instances, the TS is the best algorithm with a 0.4978 probability, while in asymmetric instances the VFS is better with a 0.4085 probability. In semi-symmetric instances, however, both algorithms seem to have a similar performance, since according to the statistical analysis the TS and the VFS are practically equivalent with a 0.4546 probability. As the spread of the points in the three plots is quite low, there is almost no uncertainty about the results of the analysis, so we can confirm that the VFS and the TS have different performances on the different sets of benchmark instances. As previously stated, this could be due to the symmetry of the instances.

Although we can only hypothesize the reason why this happens, we think that it may be mainly due to some of the characteristics that have been investigated in Section 4. For example, as the VFS uses the objective functions of the elementary landscapes to escape from the local optima, when one of the elementary landscapes is constant the escape ability of the algorithm may worsen. Moreover, when L shares a high percentage of the local optima with an elementary landscape, the escape ability may also be impacted. All these factors combined could explain why the performance of the VFS is worse in symmetric and semi-symmetric instances compared to asymmetric instances.

6 CONCLUSIONS AND FUTURE WORK

The symmetry of the distance and flow matrices is just one of the characteristics of QAP instances. Nevertheless, in this work we have seen that it has a great influence on the difficulty and importance of the elementary components that form the elementary landscape decomposition. As a result, the symmetry of the instances also appears to be very important when deciding which meta-heuristic strategies should be used to solve a particular QAP problem.

However, it is important to remark that this is only an exploratory research, so there is still much work to be done to measure the real influence of the symmetry of the QAP on the performance of optimization algorithms. On the one hand, the experiments on which we have based this paper have been performed on a relatively small benchmark of instances. Therefore, in the future, the number of instances considered in the experimentation should be increased in order to verify the conclusions of the analysis. On the other hand, this work has been mainly focused on local search based algorithms, so it would be interesting to extend the analysis to other types of meta-heuristics such as, for example, evolutionary algorithms.

APPENDIX A

In this appendix, we prove that the objective function of L_1 is constant when the distance matrix D or the flow matrix H is symmetric. As the demonstration is identical in both cases, we focus on the first one (symmetric distance matrix, $d_{a,b} = d_{b,a}$). First, we rewrite Equation (4) as follows:

$$f_1(\sigma) = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} \sum_{c=1}^{n-1} \sum_{d=c+1}^{n} g_{(a,b),(c,d)}(\sigma) \tag{9}$$

where $g_{(a,b),(c,d)}(\sigma) = \Big(d_{a,b}h_{c,d}\frac{\phi^1_{(a,b)(c,d)}(\sigma)}{2n} + d_{b,a}h_{c,d}\frac{\phi^1_{(b,a)(c,d)}(\sigma)}{2n} + d_{a,b}h_{d,c}\frac{\phi^1_{(a,b)(d,c)}(\sigma)}{2n} + d_{b,a}h_{d,c}\frac{\phi^1_{(b,a)(d,c)}(\sigma)}{2n}\Big)$ with $a \neq b$, $c \neq d$ and $1 \leq a, b, c, d \leq n$. If we can prove that $g_{(a,b),(c,d)}(\sigma)$ always has the same value regardless of σ, then we prove that f_1 is a constant function and, therefore, L_1 is a constant landscape. To this end, we analyze the five possible situations when D is symmetric:

- $\boxed{\sigma(a) = c \wedge \sigma(b) = d}$ The summands of $g_{(a,b),(c,d)}(\sigma)$ are:

Summand	Case	Value
$d_{a,b}h_{c,d}\frac{\phi^1_{(a,b)(c,d)}(\sigma)}{2n}$	$\sigma(i) = p \wedge \sigma(j) = q$	$d_{a,b}h_{c,d}\frac{n-3}{2n}$
$d_{b,a}h_{c,d}\frac{\phi^1_{(b,a)(c,d)}(\sigma)}{2n}$	$\sigma(i) = q \wedge \sigma(j) = p$	$d_{b,a}h_{c,d}\frac{1-n}{2n}$
$d_{a,b}h_{d,c}\frac{\phi^1_{(a,b)(d,c)}(\sigma)}{2n}$	$\sigma(i) = q \wedge \sigma(j) = p$	$d_{a,b}h_{d,c}\frac{1-n}{2n}$
$d_{b,a}h_{d,c}\frac{\phi^1_{(b,a)(d,c)}(\sigma)}{2n}$	$\sigma(i) = p \wedge \sigma(j) = q$	$d_{b,a}h_{d,c}\frac{n-3}{2n}$

  Therefore, we have $g_{(a,b),(c,d)}(\sigma) = d_{a,b}h_{c,d}\frac{n-3}{2n} + d_{b,a}h_{c,d}\frac{1-n}{2n} + d_{a,b}h_{d,c}\frac{1-n}{2n} + d_{b,a}h_{d,c}\frac{n-3}{2n} = \boldsymbol{d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)}$. For the sake of brevity, from now on we omit the explicit explanation of each of the summands. Their values can be calculated from Equation (7) based on the parameters of ϕ^1.
- $\boxed{\sigma(a) = d \wedge \sigma(b) = c}$ $g_{(a,b),(c,d)}(\sigma) = d_{a,b}h_{c,d}\frac{1-n}{2n} + d_{b,a}h_{c,d}\frac{n-3}{2n} + d_{a,b}h_{d,c}\frac{n-3}{2n} + d_{b,a}h_{d,c}\frac{1-n}{2n} = \boldsymbol{d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)}$.
- $\boxed{\sigma(a) = c \oplus \sigma(b) = d}$ $g_{(a,b),(c,d)}(\sigma) = d_{a,b}h_{c,d}\frac{-2}{2n} + d_{b,a}h_{d,c}\frac{-2}{2n} = \boldsymbol{d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)}$.
- $\boxed{\sigma(a) = d \oplus \sigma(b) = c}$ $g_{(a,b),(c,d)}(\sigma) = d_{b,a}h_{c,d}\frac{-2}{2n} + d_{a,b}h_{d,c}\frac{-2}{2n} = \boldsymbol{d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)}$.
- $\boxed{\sigma(a) \neq c, d \wedge \sigma(b) \neq c, d}$ $g_{(a,b),(c,d)}(\sigma) = d_{a,b}h_{c,d}\frac{-1}{2n} + d_{b,a}h_{c,d}\frac{-1}{2n} + d_{a,b}h_{d,c}\frac{-1}{2n} + d_{b,a}h_{d,c}\frac{-1}{2n} = \boldsymbol{d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)}$.

Thus, we have proved that $g_{(a,b),(c,d)}(\sigma) = d_{a,b}\Big(\frac{-1}{n}h_{c,d} + \frac{-1}{n}h_{d,c}\Big)$ for all $1 \leq a, b, c, d \leq n$ with $a \neq b$ and $c \neq d$. That is, the value of $g_{(a,b),(c,d)}(\sigma)$ is independent of σ. Therefore, f_1 is constant.

ACKNOWLEDGMENTS

This work has been partially funded by the Elkartek Program (KK-2020/00049), Consolidated Groups (IT1244-19) and BERC 2018-2021 program by the Basque Government, the PID2019-106453GA-I00/AEI/10.13039/501100011033 project and the BCAM Severo Ochoa excellence accreditation SEV-2017-0718 from the Spanish Ministry of Science and Innovation, and the European Research Council H2020 (the EMPHATIC project).

REFERENCES

[1] Ravindra Ahuja, James Orlin, and Ashish Tiwari. 2000. A Greedy Genetic Algorithm for the Quadratic Assignment Problem. *Computers and Operations Research* 27 (09 2000), 917–934.

[2] Kurt Anstreicher. 2003. Recent advances in the solution of quadratic assignment problems. *Math. Program.* 97 (07 2003), 27–42.

[3] Alessio Benavoli, Giorgio Corani, Janez Demsar, and Marco Zaffalon. 2016. Time for a change: A tutorial for comparing multiple classifiers through Bayesian analysis. *Journal of Machine Learning Research* 18 (06 2016).

[4] Una Benlic and Jin-Kao Hao. 2013. Breakout local search for the quadratic assignment problem. *Appl. Math. Comput.* 219 (01 2013), 4800–4815.

[5] Christian Blum and Andrea Roli. 2001. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Comput. Surv.* 35 (01 2001), 268–308.

[6] Nathan Brixius and Kurt Anstreicher. 2002. The Steinberg Wiring Problem. (01 2002).

[7] Rainer Burkard, Stefan Karisch, and Franz Rendl. 1991. QAPLIB-A quadratic assignment problem library. *European Journal of Operational Research* 55 (02 1991), 115–119.

[8] Rainer Burkard and J. Offermann. 1977. Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme. *Mathematical Methods of Operations Research* 21 (08 1977).

[9] Rainer Ernst Burkard. 1989. Locations with spatial interactions: the quadratic assignment problem. In *Discrete location theory.* Academic Press, 387–437.

[10] Borja Calvo, Josu Ceberio, and Jose Lozano. 2018. Bayesian inference for algorithm ranking analysis. 324–325.

[11] Borja Calvo and Guzmán Santafé. 2016. scmamp: Statistical Comparison of Multiple Algorithms in Multiple Problems. *R Journal* 8 (01 2016), 1–8.

[12] Josu Ceberio, Borja Calvo, Alexander Mendiburu, and Jose Lozano. 2018. Multi-Objectivising Combinatorial Optimisation Problems by Means of Elementary Landscape Decompositions. *Evolutionary Computation* 27 (02 2018), 1–22.

[13] Josu Ceberio, Alexander Mendiburu, and Jose Lozano. 2015. The linear ordering problem revisited. *European Journal of Operational Research* 241 (03 2015), 686–696.

[14] Anne Chao and shen-Ming Lee. 1992. Estimating the Number of Classes Via Sample Coverage. *J. Amer. Statist. Assoc.* 87 (03 1992), 210–217.

[15] Anne Chao and Mark Yang. 1993. Stopping rules and estimation for recapture debugging with unequal failure rates. *Biometrika* 80 (03 1993), 193–201.

[16] Francisco Chicano, Fabio Daolio, Gabriela Ochoa, Sebastien Verel, Marco Tomassini, and Enrique Alba. 2012. Local Optima Networks, Landscape Autocorrelation and Heuristic Search Performance, Vol. 7492.

[17] Francisco Chicano, Gabriel Luque, and Enrique Alba. 2010. Elementary landscape decomposition of the quadratic assignment problem. *Proceedings of the 12th Annual Genetic and Evolutionary Computation Conference, GECCO '10*, 1425–1432.

[18] Francisco Chicano, Gabriel Luque, and Enrique Alba. 2012. Autocorrelation measures for the quadratic assignment problem. *Appl. Math. Lett.* 25 (04 2012), 698–705.

[19] Francisco Chicano, Darrell Whitley, and Enrique Alba. 2011. A Methodology to Find the Elementary Landscape Decomposition of Combinatorial Optimization Problems. *Evolutionary computation* 19 (04 2011), 597–637.

[20] Bruno Codenotti and Luciano Margara. 1992. *Local properties of some NP-complete problems.* International Computer Science Institute.

[21] Fabio Daolio, Sebastien Verel, Gabriela Ochoa, and Marco Tomassini. 2010. Local Optima Networks of the Quadratic Assignment Problem. *IEEE Congress on Evolutionary Computation*, 1–8.

[22] Eranda Dragoti-Cela, Vladimir Deineko, and Gerhard Woeginger. 2014. Linearizable special cases of the QAP. *Journal of Combinatorial Optimization* 31 (09 2014).

[23] Zvi Drezner. 2003. A New Genetic Algorithm for the Quadratic Assignment Problem. *INFORMS Journal on Computing* 15 (08 2003), 320–330.

[24] Alwalid Elshafei. 1977. Hospital Layout as a Quadratic Assignment Problem. *Journal of The Operational Research Society* 28 (04 1977), 167–179.

[25] Günes Erdogan and Barbaros Tansel. 2007. A Branch-and-Cut Algorithm for Quadratic Assignment Problems Based on Linearizations. *Computers and Operations Research* 34 (04 2007).

[26] Anton Eremeev and Colin Reeves. 2002. Non-parametric Estimation of Properties of Combinatorial Landscapes, Vol. 2279. 31–40.

[27] Luca Maria Gambardella and D. Taillard. 2000. Ant Colonies for the Quadratic Assignment Problem. *Journal of the Operational Research Society* 50 (10 2000).

[28] J. Gavett and Norman Plyter. 1966. The Optimal Assignment of Facilities to Locations by Branch and Bound. *Operations Research* 14 (04 1966), 210–232.

[29] Arthur Geoffrion and Glenn Graves. 1976. Scheduling Parallel Production Lines with Changeover Costs: Practical Application of a Quadratic Assignment/ LP Approach. *Operations Research* 24 (08 1976), 595–610.

[30] Fred Glover. 1986. Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research* 13 (01 1986), 533–549.

[31] Lov Grover. 1992. Local search and the local structure of NP-complete problems. *Operations Research Letters* 12 (10 1992), 235–243.

[32] Leticia Hernando, Alexander Mendiburu, and Jose Lozano. 2012. An Evaluation of Methods for Estimating the Number of Local Optima in Combinatorial Optimization Problems. *Evolutionary computation* 21 (12 2012).

[33] Leticia Hernando, Alexander Mendiburu, and Jose Lozano. 2018. Anatomy of the Attraction Basins: Breaking with the Intuition. *Evolutionary Computation* 27 (05 2018), 1–32.

[34] Karla Hoffman, Manfred Padberg, and Giovanni Rinaldi. 2013. *Traveling Salesman Problem.* 1573–1578.

[35] Wim Hordijk and Peter Stadler. 1998. Amplitude Spectra of Fitness Landscapes. *Advances in Complex Systems* 1 (04 1998), 39–66.

[36] Tjalling Koopmans and Martin Beckmann. 1957. Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society* (1957), 53–76.

[37] Eliane Loiola, Nair Abreu, Paulo Boaventura-Netto, Peter Hahn, and Tania Querido. 2007. A survey of the quadratic assignment problem. *European Journal of Operational Research* 176 (01 2007), 657–690.

[38] Gabriel Luque and Enrique Alba. 2005. Metaheuristics for the DNA Fragment Assembly Problem. *International Journal of Computational Intelligence Research Research India Publications* 1 (01 2005), 973–1873.

[39] Peter Merz and Bernd Freisleben. 2000. Fitness landscape analysis and memetic algorithms for the quadratic assignment problem. *IEEE transactions on evolutionary computation* 4, 4 (2000), 337–352.

[40] Alfonsas Misevicius. 2004. An Improved Hybrid Genetic Algorithm: New Results for the Quadratic Assignment Problem. *Knowl.-Based Syst.* 17 (05 2004), 65–73.

[41] Mohammad Najaran and A. Prugel-Bennett. 2015. Quadratic assignment problem: a landscape analysis. *Evolutionary Intelligence* 8 (05 2015).

[42] Erik Pitzer and Michael Affenzeller. 2011. *A Comprehensive Survey on Fitness Landscape Analysis.* Vol. 378. 161–191.

[43] Christian Reidys and Peter Stadler. 2001. Combinatorial Landscapes. *SIAM Rev.* 44 (03 2001).

[44] Sartaj Sahni and T. Gonzales. 1976. P-complete approximation problem. *ACM J.* 23 (01 1976), 556–565.

[45] Peter Stadler. 1995. Towards a theory of landscapes. *Complex Systems and Binary Networks, Lecture Notes in Physics* 461 (01 1995), 77–163.

[46] Peter Stadler. 1996. Landscapes and their correlation functions. *Journal of Mathematical chemistry* 20, 1 (1996), 1–45.

[47] Thomas Stützle and Susana Fernandes. 2004. New Benchmark Instances for the QAP and the Experimental Analysis of Algorithms. *Lecture Notes in Computer Science* 3004, 199–209.

[48] Thomas Stützle and Holger Hoos. 2005. Stochastic Local Search-Foundations and Applications. (12 2005).

[49] Éric Taillard. 1991. Robust taboo search for the quadratic assignment problem. *Parallel computing* 17, 4-5 (1991), 443–455.

[50] Darrell Whitley and Francisco Chicano. 2014. Elementary Landscape Decomposition of the Hamiltonian Path Optimization Problem. 121–132.

[51] Darrell Whitley, Andrew Sutton, and Adele Howe. 2008. Understanding elementary landscapes. *GECCO'08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation 2008*, 585–592.

[52] Frank Wilcoxon. 1944. Individual Comparisons by Ranking Methods. *Biometrics* 1 (11 1944).

[53] Mickey Wilhelm and Thomas Ward. 1987. Solving Quadratic Assignment Problems By 'Simulated Annealing'. *Iie Transactions* 19 (03 1987), 107–119.