Autocorrelation Measures for the Quadratic Assignment Problem

Francisco Chicano, Gabriel Luque, Enrique Alba

E.T.S. Ingeniería Informática
University of Málaga, Spain

Abstract

In this article we provide an exact expression for computing the autocorrelation coefficient ξ and the autocorrelation length ℓ of any arbitrary instance of the Quadratic Assignment Problem (QAP) in polynomial time using its elementary landscape decomposition. We also provide empirical evidence of the autocorrelation length conjecture in QAP and compute the parameters ξ and ℓ for the 137 instances of the QAPLIB. Our goal is to better characterize the difficulty of this important class of problems to ease the future definition of new optimization methods. Also, the advance that this represents helps to consolidate QAP as an interesting and now better understood problem.

Keywords: Fitness landscapes, elementary landscapes, quadratic assignment problem, autocorrelation coefficient, autocorrelation length

1. Introduction

A *landscape* for a combinatorial optimization problem is a triple (X, N, f), where $f : X \to \mathbb{R}$ is the objective function to be minimized (or maximized) and the *neighborhood* function N maps a solution $x \in X$ to the set of neighboring solutions. If $y \in N(x)$ then y is a neighbor of x. There is a especial kind of landscape, called *elementary landscape*, which is of particular interest in present research due to their properties. They are characterized by the *Grover's wave equation* [1]:

$$\operatorname*{avg}_{y \in N(x)}\{f(y)\} = f(x) + \frac{k}{d}\left(\overline{f} - f(x)\right) \tag{1}$$

where d is the size of the neighborhood, $|N(x)|$, which we assume the same for all the solutions in the search space (regular neighborhood), $\overline{f}$ is the average solution evaluation over the entire search space, and k is a characteristic (problem-dependent) constant. A general landscape (X, N, f) can not always be said to be elementary, but even in this case it is possible to characterize the function f as a sum of elementary landscapes [2], called the *elementary components* of the landscape.

Email addresses: chicano@lcc.uma.es (Francisco Chicano), gabriel@lcc.uma.es (Gabriel Luque), eat@lcc.uma.es (Enrique Alba)

The Quadratic Assignment Problem (QAP) is a well-known NP-hard combinatorial optimization problem that is at the core of many real-world optimization problems [3]. A lot of research has been devoted to analyze and solve the QAP itself, and in fact some other problems can be formulated as special cases of the QAP, e.g., the Traveling Salesman Problem (TSP). Let P be a set of n facilities and L a set of n locations. For each pair of locations i and j, an arbitrary distance is specified r_{ij} and for each pair of facilities p and q, a flow is specified w_{pq}. The QAP consists in assigning to each location in L one facility in P in such a way that the total cost of the assignment is minimized. Each location can only contain one facility and all the facilities must be assigned to one location. For each pair of locations the cost is computed as the product of the distance between the locations and the flow associated to the facilities in the locations. The total cost is the sum of all the costs associated to each pair of locations. One solution to this problem is a bijection between L and P, that is, $x : L \to P$ such that x is bijective. Without loss of generality, we can just assume that $L = P = \{1, 2, \ldots, n\}$ and each solution x is a permutation in S_n, the set permutations of $\{1, 2, \ldots, n\}$. The cost function to be minimized can be formally defined as:

$$f(x) = \sum_{i,j=1}^{n} r_{ij} w_{x(i)x(j)} \tag{2}$$

In [4, 5] the authors analyzed the QAP from the point of view of landscapes theory [6] and they found the elementary landscape decomposition of the problem using the methodology presented in [7], providing expressions for each elementary component. In this paper we use the elementary decomposition of the previous work to compute the autocorrelation length ℓ and the autocorrelation coefficient ξ of any QAP instance in polynomial time (Section 2). We also present in Section 3 empirical evidence of the autocorrelation length conjecture [8], which links these values to the number of local optima of a problem, and we numerically compute ℓ and ξ for the well-known public instances of the QAPLIB [9].

2. Autocorrelation of QAP

Let us consider an infinite random walk $\{x_0, x_1, \ldots\}$ on the solution space such that $x_{i+1} \in N(x_i)$. The *random walk autocorrelation function* $r : \mathbb{N} \to \mathbb{R}$ is defined as [10]:

$$r(s) = \frac{\langle f(x_t) f(x_{t+s}) \rangle_{x_0,t} - \langle f(x_t) \rangle_{x_0,t}^2}{\langle f(x_t)^2 \rangle_{x_0,t} - \langle f(x_t) \rangle_{x_0,t}^2} \tag{3}$$

where the subindices x_0 and t indicate that the averages are computed over all the starting solutions x_0 and along the complete random walk. The *autocorrelation coefficient* ξ of a problem is a parameter proposed by Angel and Zissimopoulos [11] that gives a measure of its ruggedness. It is defined after $r(s)$ by $\xi = (1 - r(1))^{-1}$ [12]. Another measure of ruggedness is the *autocorrelation length* ℓ [13] whose definition is $\ell = \sum_{s=0}^{\infty} r(s)$. The autocorrelation coefficient ξ for the QAP was exactly computed by Angel and Zissimopoulos in [14]. However, recent results (see [4]) suggest that the expression in [14] could be invalid for some instances of the QAP. Using the landscape decomposition of the QAP we provide here a simple derivation for the expressions of ξ and ℓ. First, let us present (without proof) the results of [5] that are relevant to our goal.

Proposition 1 (Decomposition of the QAP). *For the swap neighborhood, the function f defined in (2) can be written as the sum of at most three elementary landscapes with constants $k_1 = 2n$, $k_2 = 2(n-1)$, and $k_3 = n$: $f = f_{c1} + f_{c2} + f_{c3}$. The elementary components can be defined as*

$$f_{c1} = \sum_{\substack{i,j,p,q=1 \\ i\neq j, p\neq q}}^{n} \psi_{ijpq} \frac{\Omega^1_{(i,j),(p,q)}}{2n} \tag{4}$$

$$f_{c2} = \sum_{\substack{i,j,p,q=1 \\ i\neq j, p\neq q}}^{n} \psi_{ijpq} \frac{\Omega^2_{(i,j),(p,q)}}{2(n-2)} \tag{5}$$

$$f_{c3} = \sum_{\substack{i,j,p,q=1 \\ i\neq j, p\neq q}}^{n} \psi_{ijpq} \frac{\Omega^3_{(i,j),(p,q)}}{n(n-2)} + \sum_{i,p=1}^{n} \psi_{iipp}\ \varphi_{(i,i),(p,p)} \tag{6}$$

where $\psi_{ijpq} = r_{ij}w_{pq}$, $\varphi_{(i,i),(p,p)}$ is the function defined using the Kronecker's delta by $\varphi_{(i,i),(p,p)}(x) = \delta^p_{x(i)}$, and the Ω functions are particular cases of the parameterized ϕ functions defined as:

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

The definition of the Ω functions is as follows: $\Omega^1_{(i,j),(p,q)} = \phi^{n-3,1-n,-2,0,-1}_{(i,j),(p,q)}$, $\Omega^2_{(i,j),(p,q)} = \phi^{n-3,n-3,0,0,1}_{(i,j),(p,q)}$, and $\Omega^3_{(i,j),(p,q)} = \phi^{2n-3,1,n-2,0,-1}_{(i,j),(p,q)}$.

Proof. See [5] for the proof. □

Proposition 2 (Autocorrelation measures). *The autocorrelation coefficient ξ, the autocorrelation length ℓ, and the autocorrelation function $r(s)$ can be computed from the actual problem data (instance) using the expressions:*

$$\xi = \left(W_1\frac{4}{n-1} + W_2\frac{4}{n} + W_3\frac{2}{n-1}\right)^{-1} = \frac{n(n-1)}{2n(1+W_1) + 2W_2(n-2)} \tag{8}$$

$$\ell = d\left(\frac{W_1}{2n} + \frac{W_2}{2(n-1)} + \frac{W_3}{n}\right) = \frac{W_1(1-n) + W_2(2-n) + 2(n-1)}{4} \tag{9}$$

$$r(s) = W_1\left(1 - \frac{4}{n-1}\right)^s + W_2\left(1 - \frac{4}{n}\right)^s + W_3\left(1 - \frac{2}{n-1}\right)^s \tag{10}$$

where the coefficients W_i for $i = 1,2,3$ are defined by

$$W_i = \frac{\overline{f^2_{ci}} - \overline{f_{ci}}^2}{\overline{f^2} - \overline{f}^2} \tag{11}$$

Proof. A proof for (8) and (11) can be found in [5]. Equation (9) is justified in [13] and (10) is proven in [2]. We also used the fact that $W_1 + W_2 + W_3 = 1$ to remove W_3 in the expressions for ξ and ℓ. □

As a consequence, we only need to compute W_1 and W_2 to obtain ξ and ℓ. Thus, we provide in this paper some propositions that allow us to efficiently compute W_1 and W_2. According to (11) we need to compute $\overline{f^2}$, $\overline{f}^2$, $\overline{f_{c1}^2}$, $\overline{f_{c1}}^2$, $\overline{f_{c2}^2}$, and $\overline{f_{c2}}^2$. Let us start with $\overline{f_{c1}}$ and $\overline{f_{c2}}$.

Proposition 3. *Two expressions for* $\overline{f_{c1}}$ *and* $\overline{f_{c2}}$ *are:*

$$\overline{f_{c1}} = -\frac{r_t w_t}{2n} \tag{12}$$

$$\overline{f_{c2}} = \frac{r_t w_t (n-3)}{2(n-1)(n-2)}, \tag{13}$$

where r_t *and* w_t *are defined as:*

$$r_t = \sum_{\substack{i,j=1 \\ i \neq j}}^{n} r_{ij} \quad ; \quad w_t = \sum_{\substack{p,q=1 \\ p \neq q}}^{n} w_{pq} \tag{14}$$

Proof. The average value of Ω^1 and Ω^2 is $\overline{\Omega^1} = -1$, and $\overline{\Omega^2} = (n-3)/(n-1)$ [4]. Using these average values we can compute $\overline{f_{c1}}$ and $\overline{f_{c2}}$ with the help of (4) and (5) as:

$$\overline{f_{c1}} = \frac{-1}{2n} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \psi_{ijpq} \quad ; \quad \overline{f_{c2}} = \frac{n-3}{2(n-1)(n-2)} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \psi_{ijpq} \tag{15}$$

Taking into account that $\psi_{ijpq} = r_{ij} w_{pq}$ and using the notation r_t, w_t defined above we can transform (15) in (12) and (13). □

Both expressions (12) and (13) can be computed in $O(n^2)$. Before giving an expression for $\overline{f}$ let us first introduce a new function t_n defined as:

$$\begin{aligned} t_n : \mathcal{P}(\{1,\ldots,n\}^2) &\rightarrow \mathbb{N} \\ Q &\mapsto t_n(Q) = \sum_{x \in S_n} \prod_{(i,p) \in Q} \delta^p_{x(i)} \end{aligned} \tag{16}$$

This function will be useful later in the computation of $\overline{f}$, $\overline{f^2}$, $\overline{f_{c1}^2}$, and $\overline{f_{c2}^2}$. According to its definition, the evaluation of t_n is not efficient since it requires a summation over all the permutations in S_n. However, we can simplify the expression of t_n to make the computation more efficient as the following proposition states.

Proposition 4. *The function* t_n *satisfies the following equality:*

$$t_n(Q) = \begin{cases} (n - |Q|)! & \text{if } |Q_1| = |Q_2| = |Q| \\ 0 & \text{otherwise} \end{cases}, \tag{17}$$

where Q_1 (Q_2) *denotes the set of all the first (second) elements of the pairs in* Q.

Proof. The function t_n is, in fact, a counting function that is counting the number of elements in S_n that fulfill the condition $\bigwedge_{(i,p)\in Q} x(i) = p$. Now, we must observe that if we find two pairs (i,p) and (j,q) in Q such that $i = j$ and $p \neq q$, then the value of $t_n(Q)$ must be zero because it is not possible to satisfy at the same time $x(i) = p$ and $x(j) = q$. We can characterize this situation using the condition $|Q_1| \neq |Q|$. That is, if the number of pairs in Q is not equal to the number of first elements of these pairs, then there exist in Q at least two pairs of the form (i,p) and (i,q) with $p \neq q$ and $t_n(Q) = 0$. For the same reason, $t(Q) = 0$ if $|Q_2| \neq |Q|$. If $|Q| = |Q_1| = |Q_2|$ then the pairs in Q fix the value for $|Q|$ components of the solution vector and the number of solutions in S_n with the fixed components is $t_n(Q) = (n - |Q|)!$. □

Once we have defined the t_n function and we know an efficient way of computing it we can provide an expression for $\overline{f}$.

Proposition 5. *An expression for $\overline{f}$ is:*

$$\overline{f} = \frac{r_t w_t}{n(n-1)} + \frac{r_d w_d}{n} \tag{18}$$

where $r_d = \sum_{i=1}^{n} r_{ii}$ and $w_d = \sum_{p=1}^{n} w_{pp}$.

Proof. Using the definition of f and t_n we can write:

$$\overline{f} = \frac{1}{|S_n|} \sum_{i,j,p,q=1}^{n} \psi_{ijpq} \left(\sum_{x\in S_n} \delta^p_{x(i)} \delta^q_{x(j)} \right) = \frac{1}{n!} \sum_{i,j,p,q=1}^{n} \psi_{ijpq} t_n(\{(i,p),(j,q)\}) \tag{19}$$

If we take into account that t_n can only take two different values, we can rewrite the previous expression as:

$$\overline{f} = \frac{(n-2)!}{n!} \sum_{\substack{i,\,j,\,p,\,q\,=\,1 \\ i\neq j,\, p\neq q}}^{n} \psi_{ijpq} + \frac{(n-1)!}{n!} \sum_{i,p=1}^{n} \psi_{iipp} = \frac{r_t w_t}{n(n-1)} + \frac{r_d w_d}{n} \tag{20}$$

□

With the help of the function t_n we can also provide an expression for $\overline{f^2}$.

Proposition 6. *An expression for $\overline{f^2}$ is:*

$$\overline{f^2} = \frac{1}{n!} \sum_{i,j,p,q=1}^{n} \sum_{i',j',p',q'=1}^{n} \psi_{ijpq}\psi_{i'j'p'q'} t_n\left(\{(i,p),(j,q),(i',p'),(j',q')\}\right) \tag{21}$$

which can be computed in $O(n^8)$.

Proof. Using the definition of f we can write:

$$\overline{f^2} = \frac{1}{|S_n|} \sum_{x\in S_n} \left(\sum_{i,j,p,q=1}^{n} \psi_{ijpq} \delta^p_{x(i)} \delta^q_{x(j)} \right)^2 = \frac{1}{n!} \sum_{x\in S_n} \sum_{i,j,p,q=1}^{n} \sum_{i',j',p',q'=1}^{n} \psi_{ijpq}\psi_{i'j'p'q'} \delta^p_{x(i)} \delta^q_{x(j)} \delta^{p'}_{x(i')} \delta^{q'}_{x(j')} \tag{22}$$

which can be transformed into (21) by commuting the sums and using the definition of t_n. □

The computation of $\overline{f_{c1}^2}$, $\overline{f_{c2}^2}$ requires a more complex treatment. We present their expressions in the following

Proposition 7. *Two expressions for $\overline{f_{c1}^2}$ and $\overline{f_{c2}^2}$ are:*

$$\overline{f_{c1}^2} = \frac{1}{4n^2 \cdot n!} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \sum_{\substack{i',j',p',q'=1 \\ i' \neq j', p' \neq q'}}^{n} \psi_{ijpq}\psi_{i'j'p'q'} \left(\sum_{m=1}^{7} \sum_{m'=1}^{7} c_m^{\Omega^1} c_{m'}^{\Omega^1} t_n \left(v_m^{i,j,p,q} \cup v_{m'}^{i',j',p',q'} \right) \right) \quad (23)$$

$$\overline{f_{c2}^2} = \frac{1}{4(n-2)^2 \cdot n!} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \sum_{\substack{i',j',p',q'=1 \\ i' \neq j', p' \neq q'}}^{n} \psi_{ijpq}\psi_{i'j'p'q'} \left(\sum_{m=1}^{7} \sum_{m'=1}^{7} c_m^{\Omega^2} c_{m'}^{\Omega^2} t_n \left(v_m^{i,j,p,q} \cup v_{m'}^{i',j',p',q'} \right) \right) \quad (24)$$

where the 7-dimensional parameterized vectors $v \in \left(\mathcal{P}(\mathbb{N}^2)\right)^7$ and $c \in \mathbb{R}^7$ are given in Table 1 and c^{Ω^1} and c^{Ω^2} denote the c vectors whose parameters $\alpha, \beta, \gamma, \varepsilon, \zeta$ are those of Ω^1 and Ω^2, respectively, that is, $c^{\Omega^1} = c^{n-3,1-n,-2,0,-1}$ and $c^{\Omega^2} = c^{n-3,n-3,0,0,1}$.

Component (m)	$v^{i,j,p,q}$	$c^{\alpha,\beta,\gamma,\varepsilon,\zeta}$
1	$\emptyset$	ζ
2	$\{(i,p)\}$	$(\gamma - \zeta)$
3	$\{(i,q)\}$	$(\varepsilon - \zeta)$
4	$\{(j,q)\}$	$(\gamma - \zeta)$
5	$\{(j,p)\}$	$(\varepsilon - \zeta)$
6	$\{(i,p),(j,q)\}$	$(\alpha - 2\gamma + \zeta)$
7	$\{(i,q),(j,p)\}$	$(\beta - 2\varepsilon + \zeta)$

Table 1: Content of the vectors $v^{i,j,p,q}$ and $c^{\alpha,\beta,\gamma,\varepsilon,\zeta}$.

Proof. After the definition of f_{c1} and f_{c2} we can write:

$$\overline{f_{c1}^2} = \frac{1}{4n^2 \cdot n!} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \sum_{\substack{i',j',p',q'=1 \\ i' \neq j', p' \neq q'}}^{n} \psi_{ijpq}\psi_{i'j'p'q'} \left(\sum_{x \in S_n} \Omega^1_{(i,j),(p,q)}(x) \Omega^1_{(i',j'),(p',q')}(x) \right) \quad (25)$$

$$\overline{f_{c2}^2} = \frac{1}{4(n-2)^2 \cdot n!} \sum_{\substack{i,j,p,q=1 \\ i \neq j, p \neq q}}^{n} \sum_{\substack{i',j',p',q'=1 \\ i' \neq j', p' \neq q'}}^{n} \psi_{ijpq}\psi_{i'j'p'q'} \left(\sum_{x \in S_n} \Omega^2_{(i,j),(p,q)}(x) \Omega^2_{(i',j'),(p',q')}(x) \right) \quad (26)$$

In this case it is not so simple to write the inner summation as a function of t_n. We will write the Ω functions as linear combinations of Kronecker's deltas using the definition of the Ω functions and the following characterization of the ϕ functions, which can be easily obtained after (7):

$$\phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i,j),(p,q)}(x) = \alpha\delta^p_{x(i)}\delta^q_{x(j)} + \beta\delta^q_{x(i)}\delta^p_{x(j)} + \gamma(\delta^p_{x(i)} - \delta^q_{x(j)})^2 +$$
$$+ \varepsilon(\delta^q_{x(i)} - \delta^p_{x(j)})^2 + \zeta(1-\delta^p_{x(i)})(1-\delta^q_{x(i)})(1-\delta^p_{x(j)})(1-\delta^q_{x(j)}) =$$
$$= (\gamma-\zeta)(\delta^p_{x(i)} + \delta^q_{x(j)}) + (\varepsilon-\zeta)(\delta^q_{x(i)} + \delta^p_{x(j)}) +$$
$$+ \delta^p_{x(i)}\delta^q_{x(j)}(\alpha - 2\gamma + \zeta) + \delta^q_{x(i)}\delta^p_{x(j)}(\beta - 2\varepsilon + \zeta) + \zeta \tag{27}$$

Thus, $\phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i,j),(p,q)}$ is a sum of six terms with δ and one constant, and the summation

$$\sum_{x \in S_n} \phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i,j),(p,q)}(x)\phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i',j'),(p',q')}(x) \tag{28}$$

can be written as a weighted sum of 49 t_n terms. In order to write this summation in a compact way we define one vector denoted with $v^{i,j,p,q}$ containing the sets to be considered in the t_n terms and a vector $c^{\alpha,\beta,\gamma,\varepsilon,\zeta}$ containing the coefficients for the t_n terms. The content of the previous vectors is shown in Table 1. Using v and c we can write the summation of the product of ϕ functions in the following way:

$$\sum_{x \in S_n} \phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i,j),(p,q)}(x)\phi^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{(i',j'),(p',q')}(x) = \sum_{m=1}^{7}\sum_{m'=1}^{7} c^{\alpha,\beta,\gamma,\varepsilon,\zeta}_m c^{\alpha,\beta,\gamma,\varepsilon,\zeta}_{m'} t_n\left(v^{i,j,p,q}_m \cup v^{i',j',p',q'}_{m'}\right) \tag{29}$$

and using the previous equality in (25) and (26) we obtain (23) and (24). □

Now we have efficient expressions for computing $\overline{f}$, $\overline{f^2}$, $\overline{f_{c1}}$, $\overline{f^2_{c1}}$, $\overline{f_{c2}}$, and $\overline{f^2_{c2}}$. With this expressions we are in conditions of efficiently computing the autocorrelation measures ξ and ℓ. This result is summarized in the following

Theorem 1 (Efficient computation of ξ and ℓ)**.** *In the QAP, the values of ξ and ℓ related to the swap neighborhood and defined by*

$$\xi = \frac{n(n-1)}{2n(1+W_1) + 2W_2(n-2)} \quad \text{[eq. (8)]}$$

$$\ell = \frac{W_1(1-n) + W_2(2-n) + 2(n-1)}{4} \quad \text{[eq. (9)]}$$

can be computed in polynomial time over the size of the problem n using equations (12), (13), (18), (21), (23), and (24).

Proof. After computing $\overline{f}$, $\overline{f_{c1}}$, $\overline{f_{c2}}$, $\overline{f^2}$, $\overline{f^2_{c1}}$, and $\overline{f^2_{c2}}$ using the equations (18), (12), (13), (21), (23), and (24) we should compute W_1 and W_2 using equation (11). Then, the autocorrelation coefficient ξ can be obtained with (8) and ℓ can be computed with (9). None of the previous equations requires more than eight nested summations over n and, thus, the computation can be done in $O(n^8)$. □

We have gone one step further and we have expanded the expressions for $\overline{f^2}$, $\overline{f^2_{c1}}$, and $\overline{f^2_{c2}}$ in order to make a more efficient computation. The result is a $O(n^2)$ algorithm (which we omit due to space constraints) to compute ℓ and ξ. It is not difficult to prove that such algorithm is optimal in complexity, since the data of a QAP instance is composed of $2n^2$ numbers which have to be taken into account in order to compute the autocorrelation measures.

3. Autocorrelation length conjecture

The autocorrelation length is specially important in optimization because of the *autocorrelation length conjecture*, which claims that in many landscapes the number of local optima M can be estimated by the expression $M \approx \frac{|X|}{|X(x_0,\ell)|}$ [8], where $X(x_0,\ell)$ is the set of solutions reachable from x_0 in ℓ (the autocorrelation length) or less local movements (jumps between neighbors). The previous expression is not an equation, but an approximation. It can be useful to compare the estimated number of local optima in two instances of the same problem. In effect, for a given problem in which the conjecture is applicable, the higher the value of ℓ (or ξ) the lower the number of local optima. In a landscape with a low number of local optima, a local search strategy can *a priori* find the global optimum using less steps. This phenomenon has been empirically observed for the Quadratic Assignment Problem (QAP) by Angel and Zissimopoulos in [14].

In order to check the autocorrelation length conjecture in the QAP we have generated 4000 random instances of QAP with sizes varying between $n = 4$ and $n = 11$ (500 for each value of n) using a random generator where the elements of the matrices are uniformly selected from the range [0,99]. For each instance we computed the autocorrelation length ℓ using (9) and the number of local optima (minima) by complete enumeration of the search space. We computed the Spearman correlation coefficient ρ of the number of local optima and ℓ for the instances of the same size. The results are shown in Table 2. We can observe an inverse correlation (around -0.3) between the number of local optima and the autocorrelation length. Although this fact is in agreement with the autocorrelation length conjecture, the correlation coefficient is low. However, Angel and Zissimopoulos [14] used a simulated annealing algorithm based on the swap neighborhood and reported a better performance of the algorithm as the autocorrelation length increased. Assuming that the number of local optima is a parameter with an important influence on the search, we conclude that even in problems in which the number of local optima is lowly correlated with ℓ (like QAP) the autocorrelation measures (ξ and ℓ) can be useful as estimators of the performance of local search algorithms.

n	4	5	6	7	8	9	10	11
ρ	-0.3256	-0.2317	-0.2126	-0.3195	-0.3032	-0.2943	-0.2131	-0.1640

Table 2: Spearman correlation coefficient ρ for the number of local optima and the autocorrelation length.

In Figure 1 we plot the number of local optima against the autocorrelation length ℓ for all the instances of size $n = 10$. We can observe a slight trend: as the autocorrelation length increases the number of local optima decreases. The trend is the same in all the instances with different sizes (we omit their plots).

In a second experiment we check that the autocorrelation measures provided by the elementary landscape decomposition are the same as the ones computed using statistical methods. For this experiment we have chosen six instances of the QAPLIB [9]: two small, two medium and two large instances. For each instance we have generated one random walk of length 1 000 000 and we have computed the $r(s)$ values for $s \in [0, 49]$. This process has been repeated 100 times and we have computed the average value for the 100 independent runs. The results empirically obtained and those theoretically predicted with (10) can be found in Table 3 (only for $s \in [1, 6]$). We can observe a great matching

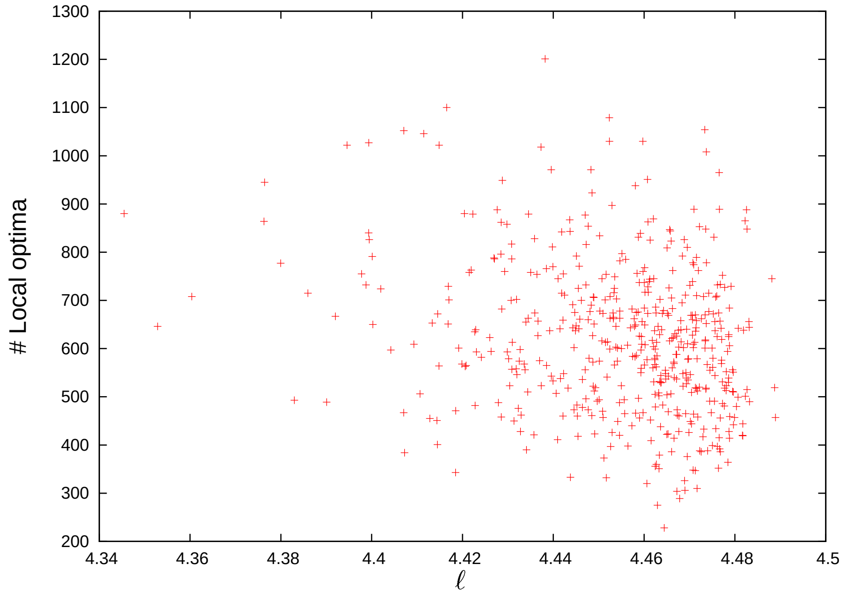

Figure 1: Number of local optima against the autocorrelation length ℓ for random instances of QAP with $n = 10$.

between the empirical and the theoretical value, as expected. The advantage of the theoretical approach is that it is much faster. The experimental results of Table 3 were obtained after 157 783 seconds of computation (more than 43 hours). However, the exact values were obtained evaluating Equation (10) in 0.4 seconds, near half a million times faster.

Finally, we have computed the values of ξ and ℓ for the 137 QAP instances found in the QAPLIB database [9]. The results, shown in Table 4 in alphabetical order, could be helpful for future investigations on the QAP. In the table we can observe some interesting behaviours, like that of the `esc` instances, which have always a value of $n/4$ for ξ and ℓ. This happens because in those instances $W_1 = W_3 = 0$ and $W_2 = 1$, that is, they are elementary landscapes with $k = 2(n-1)$. All the elementary landscapes have a value for the autocorrelation measures that does not depend on the instance data, but only on the problem size. In the case of `esc16f`, the objective function is a constant, that is, it takes the same value for every solution and the autocorrelation measures make no sense.

We should also notice that the value of ℓ and ξ depend on n, the size of the problem instance. In effect, the values are bounded (see [4]) by

$$\frac{n-1}{4} \leq \xi, \ell \leq \frac{n-1}{2} \tag{30}$$

Thus, the values of ξ and ℓ usually increase with the problem size n. As a consequence, the autocorrelation length conjecture can be applied only when the comparison is performed over instances with the same size n and, in general, it is not true that the higher the value of ℓ the easier to solve the instance, since the largest instances are usually the most difficult ones and have the highest value for ℓ (and ξ). A good indicator

Instances		$r(1)$	$r(2)$	$r(3)$	$r(4)$	$r(5)$	$r(6)$
tai10a	E	0.624255	0.393489	0.250810	0.161890	0.106102	0.070590
	T	0.624380	0.393590	0.250903	0.162013	0.106129	0.070617
esc16a	E	0.749984	0.562424	0.421759	0.316365	0.237300	0.177939
	T	0.750000	0.562500	0.421875	0.316406	0.237305	0.177979
esc64a	E	0.937402	0.878700	0.823668	0.772063	0.723672	0.678292
	T	0.937500	0.878906	0.823975	0.772476	0.724196	0.678934
lipa70a	E	0.943369	0.890041	0.839723	0.792267	0.747507	0.705296
	T	0.943479	0.890170	0.839890	0.792466	0.747735	0.705545
tho150	E	0.975680	0.951974	0.928863	0.906338	0.884384	0.862981
	T	0.975722	0.952060	0.928997	0.906518	0.884607	0.863251
tai256c	E	0.984364	0.968983	0.953843	0.938935	0.924256	0.909805
	T	0.984375	0.968994	0.953854	0.938950	0.924279	0.909837

Table 3: Experimental (E) and exact (T) values for the autocorrelation function $r(s)$ in six instances of the QAPLIB (s from 1 to 6).

Instance	ξ	ℓ	Instance	ξ	ℓ	Instance	ξ	ℓ	Instance	ξ	ℓ
bur26a	11.825	12.130	esc32b	8.000	8.000	nug16a	4.475	4.796	tai100b	35.472	39.613
bur26b	11.727	12.073	esc32c	8.000	8.000	nug16b	4.472	4.792	tai10a	2.662	2.774
bur26c	12.109	12.291	esc32d	8.000	8.000	nug17	4.836	5.220	tai10b	3.002	3.253
bur26d	12.050	12.258	esc32e	8.000	8.000	nug18	5.111	5.516	tai12a	3.419	3.674
bur26e	12.032	12.248	esc32f	8.000	8.000	nug20	5.800	6.311	tai12b	3.358	3.586
bur26f	11.962	12.208	esc32g	8.000	8.000	nug21	6.218	6.807	tai150b	40.458	42.947
bur26g	12.323	12.407	esc32h	8.000	8.000	nug22	6.751	7.446	tai15a	3.858	3.946
bur26h	12.296	12.392	esc64a	16.000	16.000	nug24	7.067	7.737	tai15b	7.000	7.000
chr12a	3.096	3.171	had12	3.743	4.092	nug25	7.308	7.987	tai17a	4.402	4.526
chr12b	3.201	3.346	had14	4.319	4.732	nug27	8.023	8.813	tai20a	5.211	5.385
chr12c	3.044	3.079	had16	4.405	4.690	nug28	8.181	8.949	tai20b	6.866	7.582
chr15a	3.917	4.049	had18	5.084	5.477	nug30	8.613	9.373	tai256c	64.000	64.000
chr15b	4.126	4.388	had20	5.830	6.352	rou12	3.158	3.275	tai25a	6.373	6.482
chr15c	3.843	3.920	kra30a	9.131	10.089	rou15	3.927	4.066	tai25b	6.896	7.374
chr18a	4.585	4.658	kra30b	9.086	10.031	rou20	5.354	5.628	tai30a	7.779	8.021
chr18b	4.632	4.742	kra32	9.848	10.908	scr12	3.407	3.657	tai30b	7.599	7.689
chr20a	5.105	5.195	lipa20a	5.072	5.135	scr15	4.303	4.650	tai35a	8.922	9.077
chr20b	5.035	5.067	lipa20b	5.196	5.358	scr20	5.514	5.885	tai35b	9.382	9.895
chr20c	5.260	5.469	lipa30a	7.622	7.732	sko100a	27.800	29.985	tai40a	10.216	10.413
chr22a	5.763	5.980	lipa30b	7.652	7.787	sko100b	28.106	30.470	tai40b	10.583	11.074
chr22b	5.672	5.819	lipa40a	10.154	10.295	sko100c	27.548	29.578	tai50a	12.675	12.839
chr25a	6.490	6.693	lipa40b	10.355	10.669	sko100d	27.535	29.557	tai50b	12.824	13.119
els19	5.178	5.494	lipa50a	12.684	12.855	sko100e	27.600	29.663	tai60a	15.292	15.563
esc128	32.000	32.000	lipa50b	12.854	13.174	sko100f	27.346	29.247	tai60b	17.837	19.691
esc16a	4.000	4.000	lipa60a	15.111	15.217	sko42	11.559	12.378	tai64c	16.000	16.000
esc16b	4.000	4.000	lipa60b	15.124	15.243	sko49	13.413	14.331	tai80a	20.214	20.419
esc16c	4.000	4.000	lipa70a	17.693	17.876	sko56	15.598	16.817	tai80b	24.021	26.612
esc16d	4.000	4.000	lipa70b	17.785	18.052	sko64	17.504	18.706	tho150	41.190	44.174
esc16e	4.000	4.000	lipa80a	20.102	20.201	sko72	19.929	21.436	tho30	8.326	8.938
esc16f	–	–	lipa80b	20.191	20.373	sko81	22.739	24.629	tho40	11.492	12.531
esc16g	4.000	4.000	lipa90a	22.610	22.716	sko90	25.046	27.024	wil100	28.362	30.868
esc16h	4.000	4.000	lipa90b	22.733	22.957	ste36a	10.954	12.122	wil50	13.832	14.860
esc16i	4.000	4.000	nug12	3.135	3.237	ste36b	11.821	13.177			
esc16j	4.000	4.000	nug14	3.892	4.155	ste36c	11.270	12.525			
esc32a	8.000	8.000	nug15	4.029	4.234	tai100a	25.195	25.383			

Table 4: Autocorrelation coefficient ξ and autocorrelation length ℓ for the 137 instances of the QAPLIB.

of the difficulty of an instance could be the pair (n, ℓ).

4. Conclusions

In this article we give an optimal way of exactly computing the autocorrelation measures ξ and ℓ for the QAP. These two parameters are important to better characterize QAP and to guide practitioners in the relative difficulty of the existing problem instances. These results can be automatically applied to all the subproblems of QAP, like de TSP. The main contributions of this work are:

- An exact expression for computing the autocorrelation coefficient ξ and the autocorrelation length ℓ of the QAP in polynomial time.
- Empirical evidence of the autocorrelation length conjecture in practice for the QAP, by using arbitrarily generated instances.
- The numerical value of ξ and ℓ for all the instances in the QAPLIB database.

As a future work we plan to obtain exact expressions for the autocorrelation measures in other problems, and study the actual practical applications of the information obtained from them.

Acknowledgements

This work has been partially funded by the Spanish Ministry of Science and Innovation and FEDER under contract TIN2008-06491-C04-01 (M* project) and the Andalusian Government under contract P07-TIC-03044 (DIRICOM project).

[1] L. K. Grover, Local search and the local structure of NP-complete problems, Operations Research Letters 12 (1992) 235–243.

[2] P. F. Stadler, Landscapes and their correlation functions, Journal of Mathematical Chemistry 20 (1996) 1–45.

[3] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, 1979.

[4] F. Chicano, G. Luque, E. Alba, Elementary landscapes decomposition of the quadratic assignment problem, in: Proceedings of GECCO 2010, ACM, Portland, OR, USA, 2010, pp. 1425–1432.

[5] F. Chicano, G. Luque, E. Alba, Elementary components of the quadratic assignment problem, arXiv:1109.4875v1, available from http://arxiv.org (september 2011).

[6] J. Barnes, B. Dimova, S. Dokov, A. Solomon, The theory of elementary landscapes, Applied Mathematics Letters 16 (3) (2003) 337–343.

[7] F. Chicano, L. D. Whitley, E. Alba, A methodology to find the elementary landscape decomposition of combinatorial optimization problems, Evolutionary Computation Journal 19 (4).

[8] P. F. Stadler, Biological Evolution and Statistical Physics, Springer, 2002, Ch. Fitness Landscapes, pp. 183–204.

[9] R. Burkard, S. Karisch, F. Rendl, QAPLIB - a quadratic assignment problem library, Journal of Global Optimization 10 (1997) 391–403.

[10] E. Weinberger, Correlated and uncorrelated fitness landscapes and how to tell the difference, Biological Cybernetics 63 (5) (1990) 325–336.

[11] E. Angel, V. Zissimopoulos, Autocorrelation coefficient for the graph bipartitioning problem, Theoretical Computer Science 191 (1998) 229–243.

[12] E. Angel, V. Zissimopoulos, On the classiffication of NP-complete problems in terms of their correlation coefficient, Discrete Applied Mathematics 99 (2000) 261–277.

[13] R. García-Pelayo, P. Stadler, Correlation length, isotropy and meta-stable states, Physica D: Nonlinear Phenomena 107 (2-4) (1997) 240–254.

[14] E. Angel, V. Zissimopoulos, On the landscape ruggedness of the quadratic assignment problem, Theoretical Computer Science 263 (2000) 159–172.