ELSEVIER

Theoretical Computer Science 191 (1998) 229–243

Theoretical Computer Science

Note

Autocorrelation coefficient for the graph bipartitioning problem

E. Angel*, V. Zissimopoulos

Laboratory Recherche en Informatique, Centre d'Orsay, Université de Paris Sud (Paris XI), CNRS-URA 410, Bat. 490, F-91405 Orsay, Cedex, France

Received November 1996
Communicated by M. Nivat

Abstract

Local search and its variants simulated annealing and tabu search are widely used heuristics to approximately solve NP-hard optimization problems. To use local search one "simply" has to specify a neighborhood structure and a cost function which has to be optimized. However, from a theoretical point of view, many questions remain unanswered, and one of the most important is: which neighborhood structure will provide the best quality solutions? The aim of this paper is to theoretically justify some results previously reported by Johnson et al. (1989, 1991) in their extended empirical study concerning simulated annealing and the graph bipartitioning problem, and to sharply tune the best landscape among the two reported in that study. Experimental results perfectly agree with the theoretical predictions.

Keywords: Local search; Simulated annealing; Autocorrelation length; Graph bipartitioning problem

1. Introduction

The graph bipartitioning problem is a well-known NP-complete combinatorial optimization problem [2]. Given an edge-weighted graph, the task is to find a partition of its vertices into two equal-sized subsets, such that the total weight of edges connecting the two subsets is minimum.

In [3] and the companion paper [4], Johnson et al. report an extensive empirical study for simulated annealing applied to this problem. They describe two neighborhoods. In the simpler one, only equal-sized partitions of the vertex set are considered, and two partitions are neighboring if one can be obtained from the other by performing an

* Corresponding author. E-mail: angel@lri.fr.

PII S0304-3975(97)00176-X

exchange of two vertices. In the other, any partition of the vertex set is a solution, and two partitions are neighboring if one can be obtained from the other by moving a single vertex from one of its sets to the other. To penalize non-equal-sized partitions, a penalty term is added to the cost function, which is function of a coefficient α called the imbalance factor.

We retain the three following aspects of their work: First, they have concluded a good behavior of simulated annealing when it is applied to random graphs, relatively to the far more elaborate standard Kernighan–Lin benchmark algorithm [5], even when the running time is taken into account. Second, they argued that the second neighborhood gives better results than the first one. And third, they have dressed the problem of choosing a value for the imbalance factor. They noticed that there was a safe range for it, and that either too small or too big values of this parameter lead to poor results.

The autocorrelation coefficient is a theoretical parameter, first introduced by Weinberger in [8], used to determine the degree of ruggedness of a landscape which is the neighborhood structure union of the cost function. In this paper, we study the autocorrelation coefficient for the two previous landscapes for the bipartitioning problem. Our study is a first step towards a theoretical justification of the previous observations by using the autocorrelation coefficient, and allows us to calculate the "optimum" value of the imbalance factor. Several experiments totally confirm our results. This research axis for evaluating different neighborhood structures is important, and in our knowledge, it is the first time that one has a theoretical justification of the behavior of local search with various landscapes.

The next of this paper is organized as follows: in Section 2 we define the autocorrelation coefficient of a landscape. In Section 3 we calculate its values for landscapes used in the graph bipartitioning problem. This enables us to say that one landscape is better than the other, and to sharply tune the best one. Section 4 is devoted to experimental evaluations. Finally, the conclusion appears in Section 5, where we also report some related studies concerning other optimization problems, and raise some open questions.

2. The autocorrelation coefficient of a landscape

Consider the problem of minimizing a real-valued function C, over a finite and discrete search space S. By definition, the cost of a solution $s \in S$ is $C(s)$. To use local search one simply has to specify a neighborhood structure which associates for each solution $s \in S$, a neighborhood $N(s) \subset S$. The association of a function C with a neighborhood structure forms what is called a landscape. Then, a local search algorithm consists in iterating the following instruction, which has to take a polynomial time in order to be useful in practice: substitute the current solution s for a best one in $N(s)$. The search will end to a local optimum, i.e., a solution for which none of its neighbors has a lower cost. Simulated annealing is a local search based heuristic, designed to avoid being trapped in poor local optima.

One of the most important characteristic of a landscape is its ruggedness. There is a strong link between this concept and the hardness of an optimization problem relatively to a local search-based algorithm. Intuitively, it is clear that the number of local minima depends on the link between the cost of a solution and the cost of its neighbors. If the cost difference between any two neighboring solutions is on average small (respectively important), then the landscape will be well (respectively bad) suited for a local search algorithm.

Let the distance between any two distinct solutions s and t, noted $d(s,t)$, be the smallest integer $k \geqslant 1$ such that there exists a sequence of solutions $s_0, \dots, s_k$ with $s_0 = s$, $\forall i \in \{0, \dots, k-1\}$, $s_{i+1} \in N(s_i)$ and $s_k = t$. In the sequel, we always have $d(s,t) = d(t,s)$. By definition, the landscape autocorrelation function [8] is

$$\rho(d) = \frac{\langle C(s)C(t)\rangle_{d(s,t)=d} - \langle C\rangle^2}{\langle C^2\rangle - \langle C\rangle^2}$$

with $\langle C\rangle$ (respectively $\langle C^2\rangle$) the average value of $C(s)$ (respectively $C^2(s)$) over S, and $\langle C(s)C(t)\rangle_{d(s,t)=d}$ the average value of the product $C(s)C(t)$ over all solutions pairs $\{s,t\}$ which are at distance d.

Function $\rho(d)$ shows the level of correlation between any two solutions which are at a distance d from each other. The most important value to know is $\rho(1)$, because the link between two adjacent solutions is of first importance for any local search based heuristic. A value close to 1 indicates that costs of any two neighboring solutions are (in average) very close. In contrary, a value close to 0 indicates that the cost of any two neighboring solutions are almost independent.

We define the autocorrelation coefficient λ by $\lambda = 1/(1 - \rho(1))$. The larger λ is, the more suited the landscape is for any based local search heuristic.

The following proposition is easy to obtain.

Proposition. *Let $Var(C) = \langle C^2\rangle - \langle C\rangle^2$ be the variance of the cost function C, then the autocorrelation function can be rewritten as*

$$\rho(d) = 1 - \frac{\langle (C(s) - C(t))^2\rangle_{d(s,t)=d}}{2Var(C)}.$$

Proof. We have for the numerator

$$\begin{aligned}\langle (C(s) - C(t))^2\rangle_{d(s,t)=d} &= \langle C^2(s) + C^2(t) - 2C(s)C(t)\rangle_{d(s,t)=d} \\ &= 2\langle C^2\rangle - 2\langle C(s)C(t)\rangle_{d(s,t)=d}.\end{aligned}$$

So, one obtains

$$\begin{aligned}\rho(d) &= 1 - \frac{2(\langle C^2\rangle - \langle C(s)C(t)\rangle_{d(s,t)=d})}{2(\langle C^2\rangle - \langle C\rangle^2)} \\ &= \frac{\langle C(s)C(t)\rangle_{d(s,t)=d} - \langle C\rangle^2}{\langle C^2\rangle - \langle C\rangle^2}. \qquad \square\end{aligned}$$

3. The graph bipartitioning problem

Given a graph and an associated matrix $X=(x_{ij})$ of edge weights, the graph bipartitioning problem asks to find a partition of its vertices V into two sets of same cardinality (more or less one) A and $V\backslash A$, such that the total edge weights $C(A)=\sum_{i\in A, j\notin A} x_{ij}$ is minimized.

We will consider the special case where the x_{ij} are random variables. Moreover, we suppose that either all the x_{ij} with $i\neq j$ are mutually independent with the same distribution, or only the x_{ij} with $i<j$ are mutually independent with the same distribution, with $x_{ij}=x_{ji}$ $\forall i\neq j$.

We consider the following two neighborhoods: In the first one, which we call SWAP, only equal-sized partitions of the vertex set are solutions. Two partitions will be neighboring if one can be obtained from the other by performing an exchange of two vertices. The union of this neighborhood structure with the above cost function forms what we call the SWAP-RGBP-landscape.

In the second one, which we call FLIP, any partition of the vertex set is a solution. Two partitions will be neighboring if one can be obtained from the other by moving a single vertex from one of its sets to the other. To penalize non-equal-sized partitions, we add a penalty term, function of a coefficient α, called the imbalance factor, to the cost function. For this neighborhood, the cost function which has to be minimized is therefore $C(A)=\sum_{i\in A, j\notin A} x_{ij} + \alpha(|A| - |V\backslash A|)^2$. The union of this neighborhood structure with this cost function forms what we call the α-FLIP-RGBP-landscape.

The first landscape has been previously studied in [6] where it was proved that for the SWAP-RGBP-landscape, the autocorrelation coefficient is $\lambda=n/8+\mathcal{O}(1/n)$, independently of the distribution of independent random variables x_{ij} (admitting finite expectation and non-null variance).

Let x be a random variable with the same distribution as the x_{ij}. We suppose that x has a non-null variance. The average (expectation) of x (respectively x^2) is noted $\langle x\rangle$ (respectively $\langle x^2\rangle$). We shall also use the notation $X_1, X_2, \ldots$ for a sequence of mutually independent and identically distributed random variables, each X_i distributed as x.

Recall that $\langle X_i X_j\rangle = \langle X_i\rangle\langle X_j\rangle$ for $i\neq j$. We are going to study the second neighborhood.

The following lemma will be extensively used.

Lemma 1. *Let S_l^n denotes the sum $\sum_{k=1}^{n} k^l \binom{n}{k}/2^n$, then we have the following recursion: $\forall l,n\geqslant 1$, $S_l^n = n(S_{l-1}^n - \frac{1}{2}S_{l-1}^{n-1})$. The four first values of S_l^n are given by*

$$S_1^n = \frac{n}{2},$$
$$S_2^n = \frac{n}{4} + \frac{n^2}{4},$$
$$S_3^n = \frac{3n^2}{8} + \frac{n^3}{8},$$
$$S_4^n = \frac{-n}{8} + \frac{3n^2}{16} + \frac{3n^3}{8} + \frac{n^4}{16}.$$

Proof. We have

$$S_l^n = \sum_{k=1}^{n} k^{l-1}\, k\binom{n}{k}\Big/ 2^n$$
$$= \sum_{k=1}^{n} k^{l-1}\, n\binom{n-1}{k-1}\Big/ 2^n.$$

By using the recursion $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$, we obtain

$$S_l^n = n\left(\sum_{k=1}^{n} k^{l-1}\binom{n}{k}\Big/ 2^n - \sum_{k=1}^{n-1} k^{l-1}\binom{n-1}{k}\Big/ 2^n\right)$$
$$= n\left(S_{l-1}^n - \frac{1}{2}S_{l-1}^{n-1}\right). \quad \square$$

Lemma 2. *We have the following combinatorial identities*:

$$\sum_{k=0}^{n} k(n-k)\binom{n}{k}\Big/ 2^n = \frac{n(n-1)}{4},$$
$$\sum_{k=0}^{n} k(n-k)(k(n-k)-1)\binom{n}{k}\Big/ 2^n = \frac{n}{16}(n^3 - 2n^2 - n + 2),$$
$$\sum_{k=0}^{n} (2k-n)^4\binom{n}{k}\Big/ 2^n = 3n^2 - 2n,$$
$$\sum_{k=0}^{n} k(n-k)(2k-n)^2\binom{n}{k}\Big/ 2^n = \frac{1}{4}(n^3 - 3n^2 + 2n).$$

Proof. For the first equality we have $k(n-k) = k^2 - nk$, therefore

$$\sum_{k=0}^{n} k(n-k)\binom{n}{k}\Big/ 2^n = S_2^n - nS_1^n$$
$$= \frac{n(n-1)}{4}.$$

For the second equality, we have $k(n-k)(k(n-k)-1) = k^4 - 2nk^3 + (n^2+1)k^2 - nk$, and so it yields

$$\sum_{k=0}^{n} k(n-k)(k(n-k)-1)\binom{n}{k}\Big/ 2^n = S_4^n - 2nS_3^n + (n^2+1)S_2^n - nS_1^n$$
$$= \frac{n}{16}(n^3 - 2n^2 - n + 2).$$

For the third equality, we have $(2k-n)^4 = 16k^4 - 32nk^3 + 24n^2k^2 - 8n^3k + n^4$. Therefore,

$$\sum_{k=0}^{n} (2k-n)^4\binom{n}{k}\Big/ 2^n = 16S_4^n - 32nS_3^n + 24n^2S_2^n - 8n^3S_1^n + n^4$$
$$= 3n^2 - 2n.$$

For the fourth equality, we have $k(n-k)(2k-n)^2 = -4k^4 + 8nk^3 - 5n^2k^2 + n^3k$. Thus,

$$\sum_{k=0}^{n} k(n-k)(2k-n)^2 \binom{n}{k} \Big/ 2^n = -4S_4^n + 8nS_3^n - 5n^2S_2^n + n^3S_1^n$$
$$= \tfrac{1}{4}(n^3 - 3n^2 + 2n). \qquad \square$$

Lemma 3. *The expectation of C is given by $\langle C\rangle = n(n-1)/4\langle x\rangle + \alpha n$.*

Proof. Using the equality $|A| + |V\backslash A| = n$, the cost of a solution A can be equivalently written

$$C(A) = \sum_{i\in A, j\notin A} x_{ij} + \alpha(2|A| - n)^2$$
$$= \sum_{i\in A, j\notin A} x_{ij} + \alpha(4|A|^2 - 4n|A| + n^2).$$

Therefore, we have

$$\langle C\rangle = \sum_{k=0}^{n} \langle C\rangle_{|A|=k} P(|A| = k)$$
$$= \sum_{k=0}^{n} (k(n-k)\langle x\rangle + \alpha(4k^2 - 4kn + n^2))P(|A| = k)$$
$$= \sum_{k=0}^{n} (k(n-k)\langle x\rangle + \alpha(4k^2 - 4kn + n^2))\binom{n}{k} \Big/ 2^n$$
$$= \frac{n(n-1)}{4}\langle x\rangle + \alpha(4S_2^n - 4nS_1^n + n^2) \text{ (using Lemma 2)}$$
$$= \frac{n(n-1)}{4}\langle x\rangle + \alpha n. \qquad \square$$

Lemma 4. *The expectation of C^2 is given by*

$$\langle C^2\rangle = \frac{n(n-1)}{4}\langle x^2\rangle + \frac{n}{16}(n^3 - 2n^2 - n + 2)\langle x\rangle^2 + \frac{\alpha}{2}(n^3 - 3n^2 + 2n)\langle x\rangle$$
$$+ \alpha^2(3n^2 - 2n).$$

Proof. The average cost of C^2 over all solutions of size k is given by

$$\langle C^2\rangle_{|A|=k} = \langle (X_1 + \cdots + X_{k(n-k)} + \alpha(2k-n)^2)^2\rangle$$
$$= \langle (X_1 + \cdots + X_{k(n-k)})^2\rangle + 2\alpha k(n-k)(2k-n)^2\langle x\rangle + \alpha^2(2k-n)^4.$$

Therefore,

$$\langle C^2\rangle = \sum_{k=0}^{n} (k(n-k)\langle x^2\rangle + k(n-k)(k(n-k)-1)\langle x\rangle^2$$
$$+ 2\alpha k(n-k)(2k-n)^2\langle x\rangle + \alpha^2(2k-n)^4)\binom{n}{k} \Big/ 2^n.$$

Finally, using Lemma 2 we obtain

$$\langle C^2\rangle = \frac{n(n-1)}{4}\langle x^2\rangle + \frac{n}{16}(n^3-2n^2-n+2)\langle x\rangle^2 + \frac{\alpha}{2}(n^3-3n^2+2n)\langle x\rangle + \alpha^2(3n^2-2n). \qquad \square$$

Lemma 5. *The variance of C is given by, $n(n-1)/4\langle x^2\rangle - n(n-1)/8\langle x\rangle^2 - \alpha n(n-1)\langle x\rangle + 2\alpha^2 n(n-1)$.*

Proof. By definition, $Var(C) = \langle C^2\rangle - \langle C\rangle^2$, and use Lemmas 3 and 4. $\square$

Lemma 6. *The average squared cost of a sum composed with m additions and $n-m$ subtractions over the random variables $X_1,\dots,X_n$ is given by*

$$\langle (X_1+\cdots+X_m-X_{m+1}-\cdots-X_n)^2\rangle = n\langle x^2\rangle + (4m^2-4mn+n^2-n)\langle x\rangle^2.$$

Proof. Just expand the expression. Recall that the X_i are mutually independent and identically distributed random variables, each X_i distributed as x. We have

$$\begin{aligned}\langle (X_1+\cdots+X_m-X_{m+1}-\cdots-X_n)^2\rangle &= n\langle X_1^2\rangle + (m(m-1)+(n-m)(n-m-1)) \\ &\quad \times\langle X_1X_2\rangle - 2m(n-m)\langle X_1X_2\rangle \\ &= n\langle x^2\rangle + (4m^2-4mn+n^2-n)\langle x\rangle^2. \qquad \square\end{aligned}$$

Lemma 7. *We have the following combinatorial identity $\sum_{k=0}^{n}((n-2k-1)^2 + (4k/n)(n-2k))\binom{n}{k}/2^n = n-1$.*

Proof. We have $((n-2k-1)^2+(4k/n)(n-2k)) = (4-\frac{8}{n})k^2+(8-4n)k+n^2-2n+1$. Therefore,

$$\begin{aligned}&\sum_{k=0}^{n}\left((n-2k-1)^2+\frac{4k}{n}(n-2k)\right)\binom{n}{k}\Big/ 2^n \\ &= \left(4-\frac{8}{n}\right)S_2^n + (8-4n)S_1^n + n^2-2n+1 \\ &= n-1. \qquad \square\end{aligned}$$

Given a solution A, we note A' a neighboring solution, that is to say a solution of the form $A\cup\{a\}$ with $a\notin A$, or $A\backslash\{a\}$ with $a\in A$.

Lemma 8. *The average squared cost difference between two neighboring solutions is given by $\langle (C(A)-C(A'))^2\rangle = (n-1)\langle x^2\rangle + 16\alpha^2(n-1) - 8\alpha(n-1)\langle x\rangle$.*

Proof. We distinguish two cases, and calculate the average cost for each of them.

For the first case, given a solution A and $a \notin A$, we make the move $A \to A' = A \cup \{a\}$. We make the assumption $|A| = k$, with $0 \leqslant k \leqslant n-1$. We have

$$\begin{aligned} C(A') - C(A) &= \sum_{i \notin A} x_{ai} - \sum_{i \in A} x_{ia} + \alpha((2|A'| - n)^2 - (2|A| - n)^2) \\ &= \sum_{i \notin A} x_{ai} - \sum_{i \in A} x_{ia} + \alpha((2|A| + 2 - n)^2 - (2|A| - n)^2) \\ &= \sum_{i \notin A} x_{ai} - \sum_{i \in A} x_{ia} + 4\alpha(2|A| + 1 - n). \end{aligned}$$

So,

$$\begin{aligned} \langle (C(A') - C(A))^2 \rangle_{|A|=k} &= \langle (X_1 + \cdots + X_{n-k-1} - X_{n-k} - \cdots - X_{n-1})^2 \rangle \\ &\quad + 16\alpha^2(n-2k-1)^2 + 2(n-k-1-k)4\alpha(2k+1-n)\langle x \rangle \\ &= \langle (X_1 + \cdots + X_{n-k-1} - X_{n-k} - \cdots - X_{n-1})^2 \rangle \\ &\quad + 16\alpha^2(n-2k-1)^2 - 8\alpha(n-2k-1)^2 \langle x \rangle, \end{aligned}$$

and using Lemma 6 we obtain

$$\begin{aligned} \langle (C(A') - C(A))^2 \rangle_{|A|=k} &= (n-1)\langle x^2 \rangle + (4k^2 - 4k(n-1) + (n-1)^2 - (n-1))\langle x \rangle^2 \\ &\quad + 16\alpha^2(n-2k-1)^2 - 8\alpha(n-2k-1)^2 \langle x \rangle \\ &= (n-1)\langle x^2 \rangle + (4k^2 - 4k(n-1) + n^2 - 3n + 2)\langle x \rangle^2 \\ &\quad + 16\alpha^2(n-2k-1)^2 - 8\alpha(n-2k-1)^2 \langle x \rangle. \end{aligned}$$

For the second case, given a solution A and $a \in A$, we make the move $A \to A' = A \backslash \{a\}$. We make the assumption $|A| = k$, with $1 \leqslant k \leqslant n$. We have

$$\begin{aligned} C(A') - C(A) &= \sum_{i \in A} x_{ia} - \sum_{i \notin A} x_{ai} + \alpha((2|A'| - n)^2 - (2|A| - n)^2) \\ &= \sum_{i \in A} x_{ia} - \sum_{i \notin A} x_{ai} + \alpha((2|A| - 2 - n)^2 - (2|A| - n)^2) \\ &= \sum_{i \in A} x_{ia} - \sum_{i \notin A} x_{ai} + 4\alpha(n - 2|A| + 1). \end{aligned}$$

So,

$$\begin{aligned} &\langle (C(A') - C(A))^2 \rangle_{|A|=k} \\ &= \langle (X_1 + \cdots + X_{k-1} - X_k - \cdots - X_{n-1})^2 \rangle + 16\alpha^2(n-2k+1)^2 \\ &\quad + 2(k-1-(n-k))4\alpha(n-2k+1)\langle x \rangle \\ &= \langle (X_1 + \cdots + X_{k-1} - X_k - \cdots - X_{n-1})^2 \rangle \\ &\quad + 16\alpha^2(n-2k+1)^2 - 8\alpha(n-2k+1)^2 \langle x \rangle \end{aligned}$$

and using Lemma 6 we obtain

$$\begin{aligned}
&\langle (C(A') - C(A))^2 \rangle_{|A|=k} \\
&= (n-1)\langle x^2 \rangle + (4(k-1)^2 - 4(k-1)(n-1) + (n-1)^2 \\
&\quad - (n-1))\langle x \rangle^2 + 16\alpha^2(n-2k+1)^2 - 8\alpha(n-2k+1)^2\langle x \rangle \\
&= (n-1)\langle x^2 \rangle + (4(k-1)^2 - 4(k-1)(n-1) + n^2 - 3n + 2)\langle x \rangle^2 \\
&\quad + 16\alpha^2(n-2k+1)^2 - 8\alpha(n-2k+1)^2\langle x \rangle.
\end{aligned}$$

Therefore, we have for $1 \leqslant k \leqslant n-1$

$$\begin{aligned}
&\langle (C(A) - C(A'))^2 \rangle_{|A|=k} \\
&= \frac{n-k}{n}\langle (C(A \cup \{a\}) - C(A))^2 \rangle_{|A|=k} + \frac{k}{n}\langle (C(A \backslash \{a\}) - C(A))^2 \rangle_{|A|=k} \\
&= (n-1)\langle x^2 \rangle + \left(k^2 \left(4 - \frac{8}{n} \right) + k(8-4n) + n^2 - 3n + 2 \right) \langle x \rangle^2 \\
&\quad + 16\alpha^2 \left((n-2k-1)^2 + \frac{4k}{n}(n-k) \right) \\
&\quad - 8\alpha \left((n-2k-1)^2 + \frac{4k}{n}(n-k) \right) \langle x \rangle.
\end{aligned}$$

When $k = 0$ or $k = n$ this expression is still valid, therefore

$$\begin{aligned}
&\langle (C(A) - C(A'))^2 \rangle \\
&= \sum_{k=0}^{n} \langle (C(A) - C(A'))^2 \rangle_{|A|=k} \; P(|A| = k) \\
&= \frac{1}{2^n} \sum_{k=0}^{n} \binom{n}{k} \left((n-1)\langle x^2 \rangle + \left(k^2 \left(4 - \frac{8}{n} \right) + k(8-4n) + n^2 - 3n + 2 \right) \langle x \rangle^2 \right. \\
&\quad - 8\alpha \left((n-2k-1)^2 + \frac{4k}{n}(n-k) \right) \langle x \rangle \\
&\quad \left. + 16\alpha^2((n-2k-1)^2 + \frac{4k}{n}(n-k)) \right) \\
&= (n-1)\langle x^2 \rangle + \left(\left(4 - \frac{8}{n} \right) S_2^n + (8-4n)S_1^n + n^2 - 3n + 2 \right) \langle x \rangle^2
\end{aligned}$$

$$- 8\alpha(n-1)\langle x\rangle + 16\alpha^2(n-1) \quad \text{(using Lemma 7)}$$

$$= (n-1)\langle x^2\rangle - 8\alpha(n-1)\langle x\rangle + 16\alpha^2(n-1). \qquad \square$$

Theorem 1. *For the α-FLIP-RGBP-landscape, the autocorrelation coefficient is given by*

$$\lambda(\alpha,x)=\frac{2\langle x^2\rangle - \langle x\rangle^2 - 8\alpha\langle x\rangle + 16\alpha^2}{4\langle x^2\rangle - 32\alpha\langle x\rangle + 64\alpha^2}\, n.$$

Proof. We have

$$\lambda(\alpha,x)=\frac{2\, Var(C)}{\langle (C(A)-C(A'))^2\rangle}.$$

Using Lemmas 5 and 8 it yields,

$$\lambda(\alpha,x)=\frac{n(n-1)/2\langle x^2\rangle - n(n-1)/4\langle x\rangle^2 - 2\alpha n(n-1)\langle x\rangle + 4\alpha^2 n(n-1)}{(n-1)\langle x^2\rangle - 8\alpha(n-1)\langle x\rangle + 16\alpha^2(n-1)}. \qquad \square$$

Theorem 2. *For all α, the autocorrelation coefficient of the α-FLIP-RGBP-landscape, is bounded below by $n/4$, and this bound is sharp. Moreover, the maximum autocorrelation coefficient is obtained for $\alpha=\langle x\rangle/4$, and its value is $n/2$ for all x with a non-null variance.*

Proof. We have the following equivalences:

$$\lambda(\alpha,x)\geqslant\frac{n}{4} \Leftrightarrow 2\langle x^2\rangle - \langle x\rangle^2 - 8\alpha\langle x\rangle + 16\alpha^2 \geqslant \langle x^2\rangle - 8\alpha\langle x\rangle + 16\alpha^2$$

$$\Leftrightarrow \langle x^2\rangle - \langle x\rangle^2 \geqslant 0.$$

Notice that the bound is attained when $\langle x^2\rangle \to 0$. Moreover,

$$\frac{\partial}{\partial\alpha}\lambda(\alpha,x)=\frac{2(\langle x^2\rangle - \langle x\rangle^2)(\langle x\rangle - 4\alpha)}{(16\alpha^2 - 8\alpha\langle x\rangle + \langle x^2\rangle)^2}.$$

The derivative becomes equal to zero for $\alpha=\langle x\rangle/4$, and some calculus show that $\lambda(\langle x\rangle/4,x)=n/2$. $\square$

As it was pointed out by Stadler [7], in order to compare various landscapes one has to take into account not only the autocorrelation coefficient, but also the diameter of the landscape (the maximum distance between two solutions). The size of the neighborhood is also important. Indeed, if two landscapes have the same autocorrelation coefficient, the one which has the smallest diameter will be "more flat" than the other, and therefore more suited for local search heuristics.

The α-FLIP-RGBP-landscape has an autocorrelation coefficient located between $n/4$ and $n/2$, whereas for the SWAP-RGBP-landscape it is only (asymptotically) $n/8$, and since the diameter of the α-FLIP-RGBP-landscape is n, and the diameter of the SWAP-RGBP-landscape is $n/2$, the overall winner is the α-FLIP-RGBP-landscape. It should also be noticed that for small values of the imbalance factor, local search stops with a partition that is far out-of-balance, and the greedy heuristic used to obtain a feasible solution (see the next section) is not sufficient to obtain good results. For large values of α, all non-balanced partitions are forbidden, and so the α-FLIP-RGBP-landscape has no more utility compared to the SWAP-RGBP-landscape. From the above considerations, we can now state the following two claims:

Claim 1. *There exists an interval of values for the parameter α, such that the α-FLIP-RGBP-landscape is more suited than the SWAP-RGBP-landscape, for a local search algorithm.*

Claim 2. *The "optimum value" of α is $\langle x \rangle/4$.*

The next section is devoted to an experimental evaluation of them.

4. Experimental results

In order to test the previous claims, a distribution has to be chosen for the random variable x. We have chosen the most widely used random graphs model. In the $G_{n,p}$ model, random graphs have n vertices, and each x_{ij} is a boolean independent random variable, which takes the value 1 with probability p, and the value 0 with probability $1-p$.

Under these statements $\langle x \rangle = \langle x^2 \rangle = p$. The autocorrelation coefficient is given by:

$$\lambda(\alpha, p) = \frac{16\alpha^2 - 8p\alpha + 2p - p^2}{64\alpha^2 - 32p\alpha + 4p} n.$$

Fig. 1 shows the ratio $R(\alpha, p) = \lambda(\alpha, p)/n$ for $0 \leqslant \alpha \leqslant 0.4$ and $0 < p \leqslant 1$. For larger values of α the ratio is almost a constant. We can see that the maximum is attained for $\alpha = p/4$.

We use the simulated annealing implementation of [3], to test our claims. Since all the procedure is based on the use of percentages, it allows us to concentrate on the quality of obtained solutions, and so the suitness of the landscape, rather than time-spent considerations.

Recall that in simulated annealing, a new solution is chosen randomly among the neighbors of the current solution, and improving moves are always accepted, whereas other moves are accepted with probability $e^{-\delta/T}$, where δ is the change in cost function, and T is a parameter, called temperature, which decreases every a fixed number (called the temperature length) of steps (usually in a geometric way, i.e. $T \leftarrow rT$, with r the geometric cooling ratio).

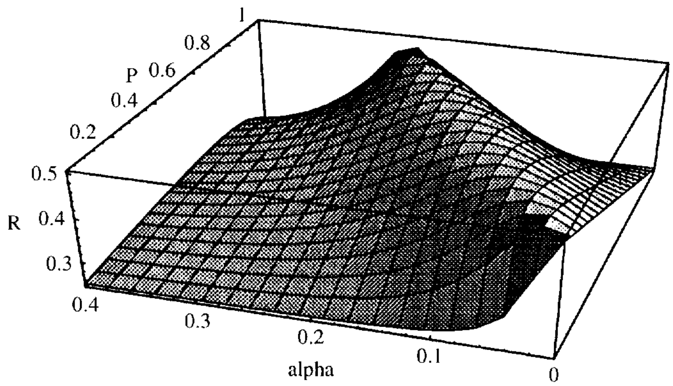

Fig. 1. The ratio $R(\alpha, p) = \lambda(\alpha, p)/n$ in relation to the imbalance factor α and the probability p.

The initial temperature is experimentally fixed in such a way that the fraction of accepted moves is approximately 40%. The temperature length is set to be 50 × instance size, and the geometric cooling ratio is 0.95. When at the end of a temperature the percentage of accepted moves is less than 2%, it means that the search is going to stop soon, because none moves will be accepted. If a such observation occurs five times, then we consider the search process as being "frozen", and the simulated annealing stops. There is an exception if a solution better than the previous best one is found, in that case we wait again for five new low-acceptance temperature completions, to stop the algorithm.

In order to obtain a feasible solution from the final one which is possibly unfeasible, the following heuristic is used at the end of the search: repeat until the partition is equal sized: find a vertex in the larger set that when it is moved to the opposite set it increases the less the cost function, and move it to the other set.

The final cost comes from the best feasible solution found, which can be the last encountered solution (possibly modified) or an earlier feasible one (non-modified).

In order to confirm experimentally the two claims of the previous section, we have applied the simulated annealing algorithm on six types of random graphs, $G_{100,0.1}$, $G_{100,0.8}$, $G_{300,0.1}$, $G_{300,0.8}$, and $G_{500,0.1}$, $G_{500,0.8}$, with imbalance factor varying between 0 and 1.

Fig. 2 resume the results we have obtained, averaged on 10 graphs for each type of random graphs. Plain lines are for the various α-FLIP-RGBP-landscapes, whereas dotted lines are for the SWAP-RGBP-landscape. One notices that they perfectly agree with the previously reported theoretical predictions. One observes, on all figures, for the cost obtained, an abrupt variation around the value $\alpha = p/4$. Moreover, the α-FLIP-RGBP-landscape, when $\alpha \geqslant p/4$, always gives better results than the SWAP-RGBP-landscape.

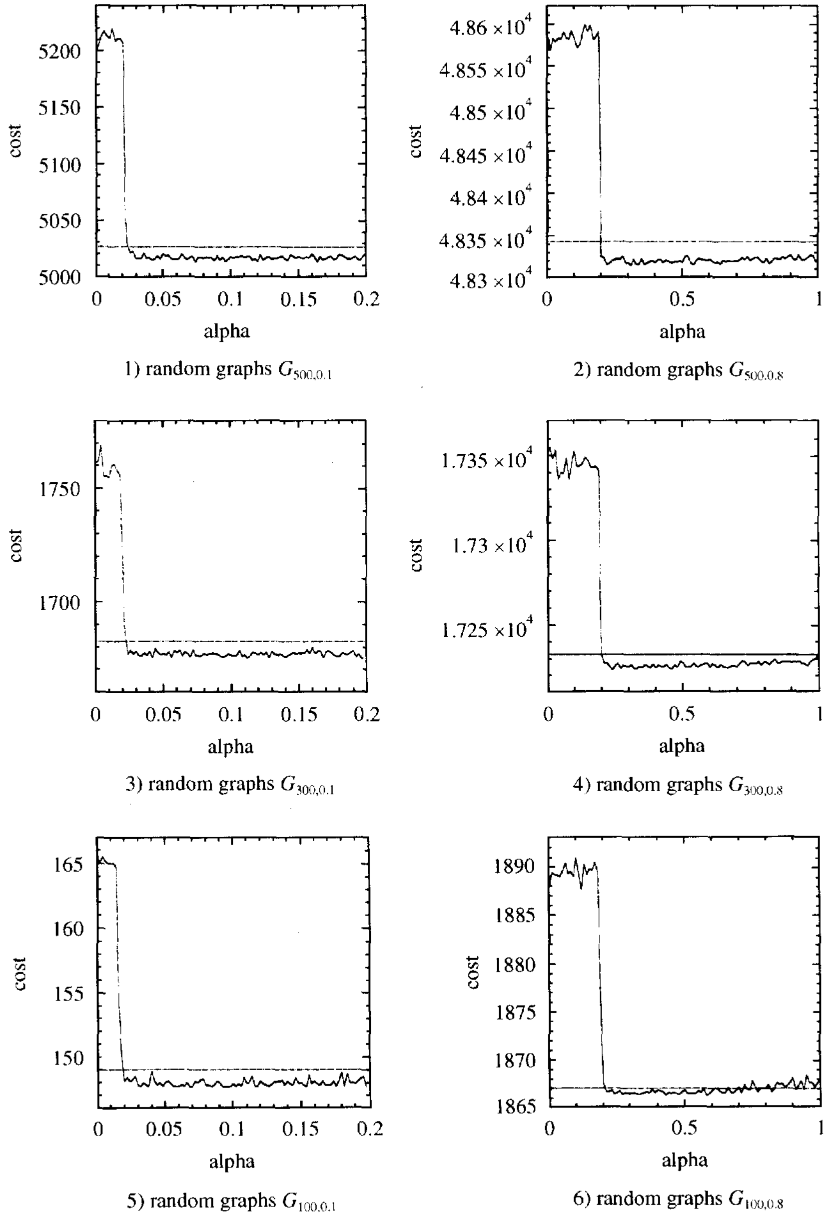

Fig. 2. The behavior of the simulated annealing with the SWAP-RGBP-landscape (dotted lines) and the α-FLIP-RGBP-landscape for different values of α.

5. Conclusions

We have studied the autocorrelation coefficient for two landscapes used in local search-based heuristics to approximatively solve the graph bipartitioning problem. This enabled us to say that one landscape was better than the other, and to sharply tune the best one. The experiments we have conducted, totally confirm these conclusions. We also think that the large value of the autocorrelation coefficient for the second studied neighborhood could explain the good behavior of simulated annealing relatively to the far more complex Kernighan–Lin heuristic.

Stadler in [7] has studied the traveling salesman problem. Two versions have been considered. Either the distance matrix is a random symmetric matrix, or it comes from euclidean distances for cities randomly distributed in a d-dimensional hypercube ($d \geqslant 2$). For the 2-SWAP-RSTSP-landscape (respectively k-opt-RSTSP-landscape ($k \geqslant 2$)) we have $\lim_{n\to\infty} \lambda = n/4$ (respectively n/k). For asymmetric TSP the situation is more involved. Stadler deduced that the 2-opt-RSTSP-landscape was better than the 2-SWAP-RSTSP-landscape. He also noticed that one cannot conclude, from the above result, that the 2-opt-RSTSP-landscape is better than 3-opt-RSTSP-landscape (which is experimentally wrong), because the landscapes have not the same diameter.

Weinberger [8] also suggested to use random walks to investigate the correlation structure of a landscape. Consider the sequence of costs generated by a random walk (t_i), which at each step moves to a new solution chosen randomly among the neighbors of the current solution. A landscape is said to be statistically isotropic if the statistics of this sequence of costs are the same, regardless of the starting point chosen for the random walk. Under this assumption, we can define an another autocorrelation function by putting

$$r(s) = \frac{\langle C(t_i)C(t_{i+s})\rangle - \langle C\rangle^2}{\langle C^2\rangle - \langle C\rangle^2}.$$

Notice the equality $\rho(1) = r(1)$.

If the landscape is Gaussian, then $r(s)$ can easily be computed from $r(1)$. A landscape is said to be Gaussian if the cost function C has a normal distribution $\mathrm{N}(\mu, \sigma^2)$, i.e.

$$\Pr(C \leqslant c) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(c-\mu)^2}{2\sigma^2}\right)$$

(μ is the expectation of C, σ^2 is the variance of C).

A random walk which is isotropic and Gaussian inevitably leads [1] to an autocorrelation function of the form $r(s) = r(1)^s$ (but the reciprocal is not always true).

In that case, one defines the autocorrelation length [8] λ by $r(s) = \rho(1)^s = \mathrm{e}^{-s/\lambda}$. Hence, $\lambda = -1/(\ln \rho(1))$, and it means that the larger is λ the closer to one is $\rho(1)$, and therefore, the more suited for a local search is the landscape. Intuitively, the autocorrelation length λ, indicates the minimum distance between any two solutions for them to have a non-correlated cost. Therefore, when one compares various landscapes it is more rigorous to compare the ratios λ/D, where D denotes the diameter of the landscape, than the values λ. But the size of the neighbor should also be taken into account.

Our autocorrelation coefficient is asymptotically equal with the autocorrelation length (when $\rho(1) \to 1$), hence the same notation. But, we have employed two terms because we cannot interpretate our autocorrelation coefficient as being an autocorrelation length, as we conjecture that the α-FLIP-RGBP-landscape is not (asymptotically) Gaussian. This is an open question, especially when $\alpha = 0$ (under this case, the maximization version of the problem we consider is the well known MAX CUT problem). For this case one has to study asymptotically the behavior when $n \to \infty$ of a random sum of random variables: $\sum_{i=0}^{N_n(n-N_n)} X_i$, with $\Pr(N_n = k) = \binom{n}{k}/2^n$.

In contrary, notice that the k-opt-RSTSP-landscape ($k \geqslant 2$) and the SWAP-RGBP-landscape are (asymptotically) Gaussian, due to a direct application of the central limit theorem.

References

[1] W. Feller, An Introduction to Probability Theory and its Applications, Wiley, New York, 1972.

[2] M.R. Garey, D.S. Johnson, Computers and Intractability – A Guide to the Theory of NP-Completeness, Freeman, New York, 1979.

[3] D.S. Johnson, C.R. Aragon, L.A. McGeoch, C. Schevon, Optimization by simulated annealing: an experimental evaluation; Part I, graph partitioning, Oper. Res. 37(6) (1989) 865–892.

[4] D.S. Johnson, C.R. Aragon, L.A. McGeoch, C. Schevon, Optimization by simulated annealing: an experimental evaluation; Part II, graph coloring and number partitioning, Oper. Res. 39 (1991) 378–406.

[5] B.W. Kernighan, S. Lin, An efficient heuristic procedure for partitioning graphs, Bell Systems Tech. J. 49 (1970) 291–307.

[6] P.F. Stadler, R. Happel, Correlation structure of the landscape of the graph-bipartitioning problem, J. Phys. A 25 (1992) 3103–3110.

[7] P.F. Stadler, W. Schnabl, The landscape of the traveling salesman problem, Phys. Lett. A 161 (1992) 337–344.

[8] E.D. Weinberger, Correlated and uncorrelated fitness landscapes and how to tell the difference, Biol. Cybernet. 63 (1990) 325–336.