Swarm and Evolutionary Computation 22 (2015) 47–65

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Regular Paper

# Anatomy of the fitness landscape for dense graph-colouring problem

M.-H. Tayarani-N. *, Adam Prügel-Bennett

*University of Southampton, SO17 1BJ Southampton, UK*

ARTICLE INFO

*Article history:*
Received 3 April 2014
Received in revised form
26 November 2014
Accepted 14 January 2015
Available online 9 March 2015

*Keywords:*
Optimization problems
Fitness landscape analysis
Graph-colouring problem

ABSTRACT

Graph-colouring is one of the best-known combinatorial optimisation problems. This paper provides a systematic analysis of many properties of the fitness landscape for random instances as a function of both the problem size and the number of colours used. The properties studied include both statistical properties of the bulk of the states, such as the distribution of fitnesses and the auto-correlation, but also properties related to the local optima of the problem. These properties include the mean time to reach the local optima, the number of local optima and the probability of reaching local optima of a given cost, the average distance between global optima and between local optima of a given cost and the closest local optimum, the expected cost as a function of the distance from a configuration and the fitness–distance correlation. Finally, an analysis of how a successful algorithm exploits the fitness distance correlation is carried out.



## 1. Introduction

This paper investigates the fitness landscape of graph-colouring for dense random graphs, concentrating in particular on the structure of local and global optima as a function of the problem size and number of colours. This allows us to characterise the behaviour of this problem as we move through the chromatic phase transition which is widely regarded as marking the transition between easy and hard problem instances. To undertake this investigation we have concentrated on problem instance up to around 100 vertices where it is possible to find the majority of low cost solutions. We have tended to concentrate on properties that we believe may be important for deciding between search algorithms and for designing new search strategies. There is a large number of such properties so as a result this paper is long. We believe that this reflects the difficulty of combinatorial search, since so many factors may be influential. In a previous paper, we performed a similar analysis on the maximum-satisfiability (MAX-SAT) problem [1]. The current paper is intended to be independent of the MAX-SAT paper, although interestingly there is a great similarity in many of the properties (as well as important differences), which we comment on in the conclusion. In one of our previous works we studied the landscape of different problems [2] and pointed out their differences and similarities. This paper in an extension of that paper which describes the graph-colouring problem in considerably more detail.

The landscape of the graph-colouring problem has attracted the attention of many researchers. In the first major effort in understanding the landscape of the graph-colouring problem, [3] uses a branch and bound search algorithm to find the local optima, and studies some properties of the solutions (note that their notion of a local optimum differs from ours). The major obstacle to this method is the problem size, where they can study graphs with up to 20 vertices. Hamiez et al. [4] have studied some properties of the solutions in the graph-colouring problem including the diversity of the configurations in a population of solutions. Culberson et al. [5] study a property of the landscape that is called the frozen set. The frozen set is a particular set of vertices that in every globally optimal configuration of the problem has the same colour class. Bouziri et al. [6] study the statistical properties of some benchmark graph-colouring problems. There are several properties of the fitness landscape that can be studied, such as the distribution of the fitness function, the number and distribution of local optima, the structure of the basins of attraction, the presence and structure of neutral networks, correlation between the quality of a solution and its distance to a local or global optimum and landscape ruggedness. Porumbel et al. [7] provide some evidence for the existence of clustering of good solutions in some graph-colouring problems, and describe an algorithm that might exploit this. Researchers have tried to propose some measures, representing the hardness of the problems. It is believed that presenting some measures for local ruggedness of a problem provides good indications about the problem difficulty. Some examples of such measures include auto-correlation [8] and fitness distance correlation [9,10]. Fitness distance correlation describes the relationship between the fitness functions and heuristic functions. These characteristics have been studied and several methods have been proposed to measure some of these properties [11,9]. However, it was quickly realised that it is easy to

* Corresponding author at: School of Electronics and Computer Science, University of Southampton, Southampton, UK.

http://dx.doi.org/10.1016/j.swevo.2015.01.005
2210-6502/

build problems where these measures do not reflect the problem difficulty [12]. Another active line of research has been to look at algebraic properties of landscapes and particularly elementary landscapes [13,14], which are shared by many well known NP-hard problems. Graph-colouring is an elementary landscape although MAX-SAT is not. Unfortunately these properties do not correlate with problem difficulty and therefore are not studied in this paper.

The rest of this paper is organised as follows. In the next section we introduce the graph-colouring problem and describe some of the features required for interpreting the results in later sections. We also describe the local search algorithm we use to find the optima in the landscape. Section 3 describes some statistical properties of random configurations, auto-correlation, and some properties of global and local optima, including the number of steps a local-search algorithm takes to get to a local optimum, number of local and global optima and distance between the optima. In Section 4, we examine the expected fitness of configurations in Hamming spheres of different radii from a local optimum. We also consider the probability of returning to a local optimum starting from a randomly chosen configuration in the Hamming sphere. We draw conclusions including making comparisons with MAX-SAT in Section 5.

## 2. Graph-colouring problem

In this section, we describe the graph-colouring problem and how we make the problem instances. We finish this section with a discussion about the local-search algorithm we use to find the local optima, and the way we distinguish different local optima from each other.

### 2.1. Problem definition

The graph-colouring problem is a combinatorial optimisation problem which belongs to the class of NP-hard problems. Given an undirected graph $G(\mathcal{V}, \mathcal{E})$, with a vertex (node) set $\mathcal{V}$ and edge set $\mathcal{E}$, and $k$ different colours, the graph-colouring problem is defined as finding a colouring of the vertices to minimise the number of edges whose vertices share the same colour. We denote a configuration of the graph-colouring problem with $k$ colours as a vector $\boldsymbol{x}$ of size $n = |\mathcal{V}|$, with elements $x_i \in \{1, 2, \ldots, k\}$ representing the colour of the $i$th node. The cost of a configuration $\boldsymbol{x}$ is defined as the number of *colour conflicts* in the graph, i.e., the number of edges whose vertices have identical colours. That is

$$c(G, \boldsymbol{x}) = \sum_{(i,j) \in \mathcal{E}} [\![ x_i = x_j ]\!], \tag{1}$$

where $[\![\text{ predicate }]\!]$ denotes the indicator function that is equal to 1 if the predicate is true and 0 otherwise. The chromatic number, $\chi(G)$ of a graph, $G$, is defined to be the smallest number of colours $k$ such that a configuration exists that has no colour conflicts (i.e. a cost of zero). We consider the problem of finding low cost configurations (i.e. try to minimise the number of colour conflicts).

In this paper, we concentrate on instances drawn from the ensemble of random graphs $\mathcal{G}(n, p)$, consisting of graphs with $n$ vertices where each edge is drawn with a probability $p$. A graph is represented by an $n \times n$ adjacency matrix, $G$. We generate these problem instances randomly, where the probability of two nodes being connected (the probability of an edge existing) is $p$. Thus a graph is generated as $G(i,j) = [R(0,1) < p]$, for $i, j = 1 \ldots n$, $i \neq j$, where $R( .,. )$ is a uniform random number generator and $[\text{predicate}] = 1$ if predicate=true and $[\text{predicate}] = 0$ if predicate=false. We focus on the case $p=0.5$, so that the graphs are dense (a dense graph is one where $p$ remains fixed as $n$ increases so the number of edges grow is of order $n^2$. In contrast, in sparse graph $p = 1/n$, so that the number of edges per vertex remains fixed). From preliminary investigations, we found that the greatest determiner of the structure of the fitness landscape is its proximity to the phase-transition. By investigating the landscape as a function of $n$ and the number of colours, $k$, we are able to characterise the behaviour around the phase-transition. Thus, we believe that the behaviour we report is typical of dense graphs at other values of $p$.

### 2.2. Colour symmetry and distance measures

An important feature of graph-colouring is that if we permute all the colours, then the cost is unchanged. As there are $k!$ permutations of the colours, there is a $k!$-fold symmetry in the search space. Graph-colouring can also be viewed as a partitioning problem, where we try to partition the vertices into $k$ partitions so as to minimise the number of edges with vertices in the same partition. In this partition view of the problem we eliminate the $k!$ symmetry of the problem. Although it is more logical to view graph-colouring as a partitioning problem, most algorithms treat the problem as a colouring problem. This reflects the fact that partitions are difficult to treat (e.g. it is non-trivial to determine whether two partitions are identical). In this paper, we have tried to accommodate both views of the problem: either as a colouring problem with a $k!$-fold symmetry or as a partition problem.

As a consequence of these two views of the problem we consider two distance measures between configurations. The first is the Hamming distance defined as

$$D_h(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n} [\![ x_i \neq y_i ]\!]. \tag{2}$$

The second measure is a measure of the 'partition distance' defined as

$$D_p(\boldsymbol{x}, \boldsymbol{y}) = \min_{\pi} D_h(\boldsymbol{x}, \pi(\boldsymbol{y})), \tag{3}$$

where $\pi(\cdot)$ is a permutation operator that permutes the colours. The minimisation is over all possible permutations of the $k$ colours. The partition distance measures the smallest number of reallocations of partition membership to make the partition represented by $\boldsymbol{x}$ into the partition represented by $\boldsymbol{y}$. When the Hamming distance is small, it is often the same as the partition distance. In practice, we can compute the partition distance in $O(k^3)$ by representing the colour matching as a linear assignment problem and using the Hungarian algorithm [15].

It is useful to understand the distribution of distances between random configurations. Note that this property depends only on the number of vertices, $n$, and number of colours, $k$, but is otherwise independent of the problem instances. For two randomly generated configurations the probability that the Hamming distance is equal to $h$ is given by a binomial distribution

$$\mathbb{P}(D_h(\boldsymbol{x}, \boldsymbol{y}) = h) = \binom{n}{h} \left(1 - \frac{1}{k}\right)^h \left(\frac{1}{k}\right)^{n-h}, \tag{4}$$

so that the expected Hamming distance between randomly chosen configurations is

$$E(D_h(\boldsymbol{x}, \boldsymbol{y})) = n\left(1 - \frac{1}{k}\right). \tag{5}$$

We are not aware of an analytic formula for the probability distribution of partition distances between randomly drawn configurations. In Fig. 1b we show the probability distribution of distances between randomly chosen pairs of configurations. The partition distance is measured empirically by sampling.

The average partition distances between the solutions in the search space for different values of $n$ and $k$ are shown in Fig. 1b. In this figure, the horizontal axis is $k$ and the vertical axis is the