Improved bounds on the complexity of graph coloring

This paper appeared in: *Proceedings of the 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, track „Advances in the Theory of Computing”, Timisoara (Romania), 2010, pp. 347-354, IEEE Computer Society*

Zoltán Ádám Mann
Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Budapest, Hungary
e-mail: zoltan.mann@cs.bme.hu

Anikó Szajkó
Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Budapest, Hungary
e-mail: szajko.aniko@gmail.com

***Abstract*—The coloring of random graphs has been the subject of intensive research in the last decades. As a result, the asymptotic behaviour of both the chromatic number and the complexity of the colorability problem are quite well understood. However, the asymptotic results give limited help in predicting the behaviour in specific finite cases.**

In this paper, we consider the application of the usual backtrack algorithm to random graphs, and analyze the expected size of the search tree as a machine-independent measure of algorithm complexity. With a combination of combinatorial, probabilistic and analytical methods, we derive upper and lower bounds for the expected size of the search tree. Our bounds are much tighter than previous results and thus enable accurate prediction of algorithm runtime.

I. Introduction and previous work

Graph coloring is one of the most fundamental problems in algorithmic graph theory, with many practical applications such as register allocation, frequency assignment, pattern matching, and scheduling [16], [6], [15]. Unfortunately, graph coloring is NP-complete [9].

Although graph coloring is hard in the worst case, it is easier in the average case [19]. The probabilistic analysis of the coloring of random graphs was first suggested in [8]. Subsequent work [10], [4], [12] uncovered the order of magnitude of the expected chromatic number of random graphs. Through more recent work [2], [1], we can determine almost exactly the expected chromatic number of a random graph in the limit: with probability tending to 1 when the size of the graph tends to infinity, the expected chromatic number of a random graph is one of two possible values.

Empirical study of the behaviour of search algorithms and the complexity of graph coloring problem instances [14], [18] has lead to the discovery of a phase transition phenomenon with an accompanying easy-hard-easy pattern [7], [11]. Briefly, this means that for small values of the edges/vertices ratio (underconstrained case), almost all random graphs are colorable. When the connectivity of the graph is increased, the ratio of colorable graphs abruptly drops from almost 1 to almost 0 (phase transition). After this critical regime, almost all graphs are uncolorable (overconstrained case). In the underconstrained case, coloring is easy: even the simplest heuristics usually find a proper coloring [19], [5]. In the overconstrained case, it is easy for backtracking algorithms to prove uncolorability because they quickly reach contradiction [17]. The hardest instances lie in the critical regime [7].

Summarizing these results, one can state that we have a good *quantitative* understanding of graph coloring *in the limit* (when the size of the graph tends to infinity) and a good *qualitative* understanding of it in the finite case. Our aim in this paper is to study the hardness of graph coloring *quantitatively* with accurate results for *finite* graphs.

Specifically, we consider the application of the usual backtrack search to the coloring of random graphs. We restrict ourselves to the non-colorable case; extension of our model to the colorable case remains as future work. We use the size of the search tree as a measure of complexity and analyze its expected value as a function of input parameters.

Lower and upper bounds for the expected size of the search tree in a similar model have been presented by Bender and Wilf [3]. Their main focus was on the study of the asymptotic behavior of the search tree. In finite cases, the difference between their lower and upper bounds can be quite large (several orders of magnitude), as shown in Table I.

TABLE I
EXAMPLES OF THE BOUNDS BY BENDER AND WILF ($k = 7$)

	$n = 30$, $p = 0.5$	$n = 50$, $p = 0.5$	$n = 50$, $p = 0.4$	$n = 30$, $p = 0.7$
lower bound	$6.41 \cdot 10^9$	$6.45 \cdot 10^9$	$3.26 \cdot 10^{12}$	$4.94 \cdot 10^6$
upper bound	$1.81 \cdot 10^{12}$	$1.83 \cdot 10^{12}$	$1.84 \cdot 10^{15}$	$5.27 \cdot 10^8$

Therefore, our aim is to significantly improve these bounds, in order to enable accurate prediction of the runtime of the algorithm on specific graphs. This is beneficial for example for random restart algorithms to decide when to perform the restart. Also, runtime prediction can be used to decide whether it is at all feasible to solve a problem instance with such an exact algorithm.

We use a combination of combinatorial, probabilistic and analytical methods. We show that a simple probabilistic model and some combinatorial considerations yield a first pair of non-trivial upper and lower bounds. As a by-product of our first upper bound, we also obtain a short proof for a theorem of Wilf [20]. We then use Jensen's inequality to significantly improve our lower bound. In the second half of the paper, we perform a detailed – and quite technical – case analysis to obtain a series of ever sharper (but also increasingly complicated) lower and upper bounds. At the end we show empirically how the bounds are getting closer to each other and how much they improve the bounds of Bender and Wilf.

II. Preliminaries

We consider the decision version of the graph coloring problem, in which the input consists of an undirected graph $G = (V, E)$ and a number k, and the task is to decide whether the vertices of G can be colored with k colors such that adjacent vertices are not assigned the same color. The input graph is a random graph taken from $G_{n,p}$, meaning that it has n vertices and each pair of vertices is connected by an edge with probability p independently from each other. The vertices of the graph will be denoted by $v_1, \ldots, v_n$, the colors by $1, \ldots, k$. A *coloring* assigns a color to each vertex; a *partial coloring* assigns a color to some of the vertices. A (partial) coloring is *invalid* if there is a pair of adjacent vertices with the same color, otherwise the (partial) coloring is *valid*.

The backtrack algorithm considers partial colorings. It starts with the empty partial coloring, in which no vertex has a color. This is the root – that is, the single node on level 0 – of the search tree. Level t of the search tree contains the k^t possible partial colorings of $v_1, \ldots, v_t$. The search tree, denoted by T, has n levels, with the last level containing the colorings of the graph. Let T_t denote the set of partial colorings on level t. If $t < n$ and $w \in T_t$, then w has k children in the search tree: those partial colorings of $v_1, \ldots, v_{t+1}$ that assign to the first t vertices the same colors as w.

In each partial coloring w, the backtrack algorithm considers the children of w and visits only those that are valid. Note that T depends only on n and k, not on the specific input graph. However, the algorithm visits only a subset of the nodes of T, depending on which vertices of G are actually connected. The number of actually visited nodes of T will be used to measure the complexity of the given problem instance.

III. The expected number of visited nodes of T

For each $w \in T$, we define the following random variable (the value of which depends on the choice of G):

$$Y_w = \begin{cases} 1 & \text{if } w \text{ is valid,} \\ 0 & \text{else.} \end{cases}$$

Let $p_w = Pr(Y_w = 1)$. Moreover, we define one more random variable (whose value also depends on the choice of G): $Y =$ the number of visited nodes of T.

Since the algorithm visits exactly the valid partial colorings, it follows that $Y = \sum_{w \in T} Y_w$, and thus $E(Y) = \sum_{w \in T} E(Y_w)$. Moreover, it is clear that $E(Y_w) = p_w$. It follows that the expected number of visited nodes in T is: $E(Y) = \sum_{w \in T} p_w$.

Let $Q(w) := \{\{x, y\} \in V^2 : x \neq y, color(x) = color(y)\}$, where V^2 is the set of unordered pairs of elements of V. Let $q(w) := |Q(w)|$. Clearly, w is valid if and only if, for all $\{x, y\} \in Q(w)$, x and y are not adjacent. It follows that $p_w = (1 - p)^{q(w)}$ and thus the expected number of visited nodes of T is:

$$E(Y) = \sum_{w \in T} (1 - p)^{q(w)}.$$

Note that computing $E(Y)$ through this formula is not tractable since $|T|$ is exponentially large in n.

IV. Simple lower and upper bounds

In the following, we denote by $s(w, i)$ (or simply s_i if it is clear which partial coloring is considered) the number of vertices of G that are assigned color i in partial coloring w.

Proposition 1. *For all* $w \in T_t$, $q(w) \leq \binom{t}{2}$.

Proof:

$$\begin{aligned} q(w) &= \sum_{i=1}^{k} \binom{s_i}{2} = \frac{1}{2} \left(\sum_{i=1}^{k} s_i^2 - \sum_{i=1}^{k} s_i \right) \leq \\ &\leq \frac{1}{2} \left(\left(\sum_{i=1}^{k} s_i \right)^2 - \sum_{i=1}^{k} s_i \right) = \frac{1}{2} \left(t^2 - t \right) = \binom{t}{2}. \end{aligned}$$

■

As a consequence, $\sum_{w \in T_t} (1-p)^{q(w)} \geq \sum_{w \in T_t} (1-p)^{\binom{t}{2}} = k^t \cdot (1-p)^{\binom{t}{2}}$, and thus we obtain the following – easily computable – lower bound:

$$E(Y) = \sum_{w \in T} (1 - p)^{q(w)} \geq \sum_{t=0}^{n} k^t \cdot (1 - p)^{\binom{t}{2}}. \qquad (1)$$

Proposition 2. *For all* $w \in T_t$, $q(w) \geq \frac{1}{2} \left(\frac{t^2}{k} - t \right)$.

Proof: Since

$$\frac{\sum_{i=1}^{k} s_i^2}{k} \geq \left(\frac{\sum_{i=1}^{k} s_i}{k} \right)^2 = \frac{t^2}{k^2},$$

it follows that

$$q(w) = \frac{1}{2} \left(\sum_{i=1}^{k} s_i^2 - \sum_{i=1}^{k} s_i \right) \geq \frac{1}{2} \left(\frac{t^2}{k} - t \right).$$

■

As a consequence, $\sum_{w \in T_t} (1 - p)^{q(w)} \leq \sum_{w \in T_t} (1 - p)^{\frac{1}{2}\left(\frac{t^2}{k} - t\right)} = k^t \cdot (1 - p)^{\frac{1}{2}\left(\frac{t^2}{k} - t\right)}$, and thus we obtain the following – easily computable – upper bound:

$$E(Y) = \sum_{w \in T} (1 - p)^{q(w)} \leq \sum_{t=0}^{n} k^t \cdot (1 - p)^{\frac{1}{2}\left(\frac{t^2}{k} - t\right)}. \qquad (2)$$

As a by-product, we obtain a simple proof for a theorem of Wilf [20]:

Corollary 3 (Wilf, 1984)**.** *The average-case complexity of coloring a random graph with a constant number of colors is $O(1)$.*

Proof: According to (2), the complexity of the backtracking algorithm is not more than $\sum_{t=0}^{\infty} k^t \cdot (1-p)^{\frac{1}{2}\left(\frac{t^2}{k}-t\right)} = \sum_{t=0}^{\infty} A^t \cdot B^{t^2}$, where $A = \frac{k}{\sqrt{1-p}}$ and $B = \sqrt[2k]{1-p}$. Since $0 < B < 1$, the root test shows that $\sum_{t=0}^{\infty} A^t \cdot B^{t^2}$ is convergent. This upper bound is independent of n. ■

Numerical comparison of the lower bound (1) and the upper bound (2) has shown that their difference is quite large in practice (see Section X). This motivates the quest for better lower and upper bounds.

V. Refined lower bound using Jensen's inequality

Let $\bar{q} := \frac{1}{|T_t|} \sum_{w \in T_t} q(w)$ denote the mean of the $q(w)$ values in T_t.

Lemma 4. $\bar{q} = \frac{t^2-t}{2k}$.

Proof: Since the role of the colors is symmetric, it is easy to see that

$$\begin{aligned} \sum_{w \in T_t} q(w) &= \sum_{w \in T_t} \sum_{i=1}^{k} \binom{s(w,i)}{2} = \\ &= \sum_{i=1}^{k} \sum_{w \in T_t} \binom{s(w,i)}{2} = k \sum_{w \in T_t} \binom{s(w,1)}{2}. \end{aligned}$$

In order to compute this sum, we should examine for how many $w \in T_t$ we have $s(w,1) = j$. In other words, how many colorings exist for the first t vertices, in which exactly j vertices receive color 1. Since the j vertices can be chosen in $\binom{t}{j}$ ways and the remaining $t-j$ vertices must receive a color from the remaining $k-1$ colors, there are $\binom{t}{j}(k-1)^{t-j}$ such colorings. Hence, the above sum can be written as follows:

$$\sum_{w \in T_t} q(w) = k \sum_{j=0}^{t} \binom{j}{2} \binom{t}{j} (k-1)^{t-j}.$$

The members of the sum corresponding to $j = 0$ and $j = 1$ are 0, thus it is enough to start with $j = 2$. Using that $\binom{j}{2}\binom{t}{j} = \binom{t}{2}\binom{t-2}{j-2}$, we have:

$$\begin{aligned} \sum_{w \in T_t} q(w) &= k \binom{t}{2} \sum_{j=2}^{t} \binom{t-2}{j-2} (k-1)^{t-j} = \\ &= k \binom{t}{2} \sum_{\ell=0}^{t-2} \binom{t-2}{\ell} (k-1)^{t-2-\ell}. \end{aligned}$$

Using the binomial theorem for $((k-1)+1)^{t-2}$, this can be written as

$$\sum_{w \in T_t} q(w) = k \binom{t}{2} k^{t-2} = k^{t-1} \binom{t}{2}.$$

Dividing this by $|T_t| = k^t$ leads to the stated formula for $\bar{q}$. ■

Theorem 5. $E(Y) = \sum_{w \in T} (1-p)^{q(w)} \geq \sum_{t=0}^{n} k^t (1-p)^{\frac{t^2-t}{2k}}$.

Proof: Since $x \mapsto (1-p)^x$ is convex, thus Jensen's inequality gives

$$\frac{1}{|T_t|} \sum_{w \in T_t} (1-p)^{q(w)} \geq (1-p)^{\frac{1}{|T_t|} \sum_{w \in T_t} q(w)} = (1-p)^{\frac{t^2-t}{2k}},$$

yielding exactly the stated bound. (In the last equation, we used Lemma 4.) ■

Comparing the lower bound of Theorem 5 and the upper bound (2), it can be seen that both have the form $\sum_{t=0}^{n} k^t \cdot (1-p)^{\frac{t^2}{2k}+\Theta(t)}$. Numerical comparison has shown that they are indeed closer to each other than the bounds (1) and (2), but there is still room for improvement (see Section X).

VI. Calculating with q_{min} term separately

In order to improve the bounds, we look at the distribution of $q(w)$ in more detail. Since $x \mapsto (1-p)^x$ is monotonously decreasing, smaller values of $q(w)$ are more significant than higher values. Moreover, the results of Proposition 1, Proposition 2 and Lemma 4 show that the mean of the $q(w)$ values is closer to the minimum than to the maximum, suggesting that small values of $q(w)$ have a high frequency. This is also justified by empirical results, see Fig. 1 for an example. Therefore, we investigate the smallest values of $q(w)$.

Proposition 6. *Moving a vertex from a color class with A vertices to a color class with B vertices decreases $q(w)$ by $A-B-1$ (if this is negative, then $q(w)$ is increased).*

Proof: The change in $q(w)$ is $\binom{A}{2} + \binom{B}{2} - \left(\binom{A-1}{2} + \binom{B+1}{2}\right) = \frac{A-1}{2}(A-(A-2)) + \frac{B}{2}(B-1-(B+1)) = A-1-B$. ■

We call such a move a *correction move*, if $A > B$. During a correction move, q either decreases or remains constant.

Proposition 7. *If $q(w)$ is minimal in T_t, then in the partial coloring w each color class contains either $\left\lceil \frac{t}{k} \right\rceil$ or $\left\lfloor \frac{t}{k} \right\rfloor$ vertices.*

Proof: Since the average size of a color class is $\frac{t}{k}$, the biggest color class has at least $\left\lceil \frac{t}{k} \right\rceil$ vertices, and the smallest color class has at most $\left\lfloor \frac{t}{k} \right\rfloor$ elements. Using proof by contradiction, we assume that the sizes of the biggest and smallest color classes differ by at least 2. Then, it follows from Proposition 6 that moving a vertex from the biggest color class to the smallest color class decreases $q(w)$ by at least 1. This contradicts the minimality of $q(w)$. ■

As can be seen, an arbitrary partial coloring w can be turned into a partial coloring w' with $q(w') = q_{min}$ by using a sequence of correction moves.

Let $t = ck + d$ where $0 \leq d \leq k-1$. Then, according to Proposition 7, colorings with minimum $q(w)$ have d color classes of size $c+1$ and $k-d$ color classes of size c. Thus, the

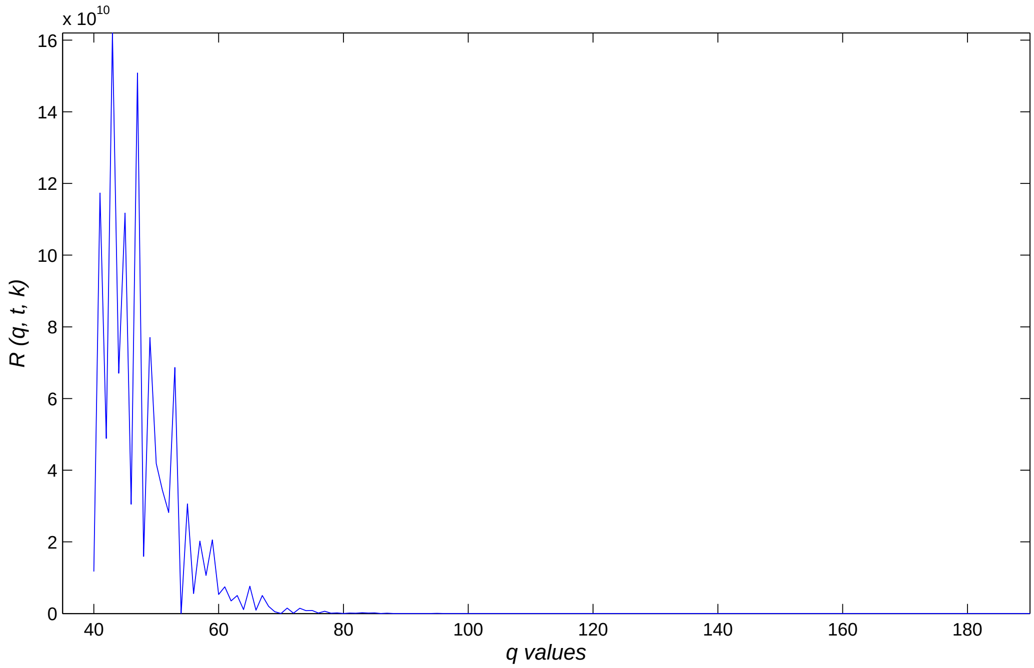

Fig. 1. The frequency of different $q(w)$ values for $t = 20$ and $k = 4$. Here, $q_{min} = 40$, $\bar{q} = 47.5$ and $q_{max} = 190$. It can be seen that the distribution is concentrated in the lower region of the possible q values.

minimum value of $q(w)$ is: $q_{min} = d\binom{c+1}{2} + (k-d)\binom{c}{2}$. This is sharp for each t and k, and thus a slightly more accurate bound than the one of Proposition 2.

Let $R(q,t,k) := |\{w \in T_t : q(w) = q\}|$ denote the frequency of value q among the $q(w)$ values of nodes in T_t.

Proposition 8. $R(q_{min},t,k) = \binom{k}{d} \cdot \frac{t!}{((c+1)!)^d (c!)^{(k-d)}}$.

Proof: There are $\binom{k}{d}$ possibilities to choose the d color classes whose size should be $c+1$. Given the size of each color class as $s_1, s_2, \ldots, s_k$, there are $\frac{t!}{s_1! \cdot s_2! \cdot \ldots \cdot s_k!}$ possibilities to distribute the t vertices among the color classes. ■

Using $R_{min} := R(q_{min},t,k)$, this leads to a more accurate upper bound:

$$\sum_{w \in T_t} (1-p)^{q(w)} \leq R_{min}(1-p)^{q_{min}} + (k^t - R_{min})(1-p)^{q_{min}+1}$$

and thus

$$E(Y) \leq \sum_{t=0}^{n} R_{min}(1-p)^{q_{min}} + (k^t - R_{min})(1-p)^{q_{min}+1}. \tag{3}$$

The lower bound can also be improved by separating the term corresponding to q_{min}:

Theorem 9.

$$E(Y) \geq \sum_{t=0}^{n} R_{min}(1-p)^{q_{min}} + (k^t - R_{min})(1-p)^{\widehat{q_1}},$$

where $\widehat{q_1} = \frac{k^t \bar{q} - R_{min} \cdot q_{min}}{k^t - R_{min}}$.

Proof: Let $T_t^{(1)} := \{w \in T_t : q(w) = q_{min}\}$ and $T_t^{(1+)} := \{w \in T_t : q(w) > q_{min}\}$. Clearly, $|T_t^{(1)}| = R_{min}$ and $|T_t^{(1+)}| = k^t - R_{min}$. Moreover, $\sum_{w \in T_t^{(1)}} q(w) = R_{min} q_{min}$ and $\sum_{w \in T_t^{(1+)}} q(w) = k^t \bar{q} - R_{min} q_{min}$. Using Jensen's inequality,

$$\begin{aligned} \sum_{w \in T_t^{(1+)}} (1-p)^{q(w)} &\geq \left|T_t^{(1+)}\right| (1-p)^{\frac{1}{\left|T_t^{(1+)}\right|} \sum_{w \in T_t^{(1+)}} q(w)} = \\ &= (k^t - R_{min})(1-p)^{\frac{k^t \bar{q} - R_{min} q_{min}}{k^t - R_{min}}}. \end{aligned}$$

Together with $\sum_{w \in T_t^{(1)}} (1-p)^{q(w)} = R_{min}(1-p)^{q_{min}}$, this yields the stated bound. ■

VII. Frequency of $q_{min} + 1$

In order to further improve our bounds in an analogous way, the frequency of $q_{min} + 1$ should be calculated.

Consider a partial coloring w with $q(w) = q_{min} + 1$. Since $q(w) > q_{min}$, we can perform a correction move: we move a vertex from the biggest color class (containing A vertices) to the smallest color class (containing B vertices). We thus obtain a new partial coloring w' with $q(w') < q(w)$, see Fig. 2. It follows that $q(w') = q_{min}$ and the decrease is $A - B - 1 = 1$, hence in w' the two color classes contain the same number of vertices ($A - 1 = B + 1$). Moreover, since $q(w') = q_{min}$, all color classes in w' contain c or $c+1$ vertices. From these facts, we can deduce the possible sizes of color classes in w.

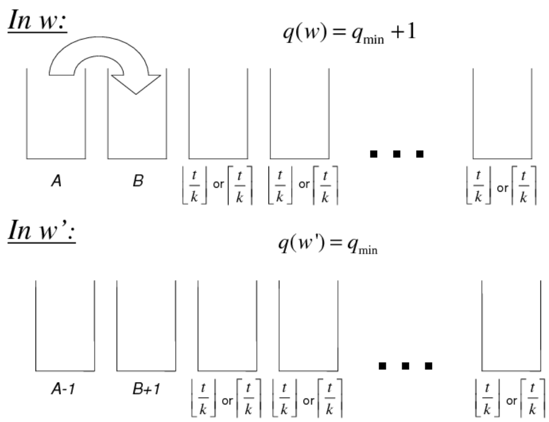

Fig. 2. Number of elements in different color classes.

A. *Case $d \neq 0$, $d \neq 1$ and $d \neq k-1$:*

In w', there are $k-d$ color classes with c elements and d color classes with $c+1$ elements. The new color classes with $A-1$ and $B+1$ elements in w' contain either c or $c+1$ elements.

1) If $A-1 = B+1 = c$ and $c \geq 1$: In this case, in w:

- one color class contains $c-1$ elements
- $k-d-2$ color classes contain c elements
- $d+1$ color classes contain $c+1$ elements

Hence, the frequency of this case is:

$$\binom{k}{1}\binom{k-1}{d+1}\frac{t!}{((c+1)!)^{d+1}(c!)^{k-d-2}(c-1)!} =$$
$$= \frac{k!t!}{(d+1)!\,(k-d-2)!(c+1)^{d+1}(c)^{k-1}((c-1)!)^{k}}.$$

2) If $A-1 = B+1 = c+1$ and $d \geq 2$: Then in w:

- $k-d+1$ color classes contain c elements
- $d-2$ color classes contain $c+1$ elements (thus $d \geq 2$)
- 1 color class contains $c+2$ elements

The frequency of this case:

$$\binom{k}{1}\binom{k-1}{d-2}\frac{t!}{(c+2)!\,((c+1)!)^{d-2}(c!)^{k-d+1}} =$$
$$= \frac{k!t!}{(d-2)!\,(k-d+1)!(c+2)\,(c+1)^{d-1}(c!)^{k}}.$$

B. *Case $d = 0$ and $c \geq 1$:*

In w', there are exactly c elements in all color classes. Thus in w:

- 1 color class contains $c-1$ elements
- $k-2$ color classes contain c elements
- 1 color class contains $c+1$ elements

The frequency of this case:

$$\binom{k}{1}\binom{k-1}{1}\frac{t!}{(c-1)!(c+1)!\,(c!)^{k-2}} =$$
$$= \frac{k(k-1)t!}{((c-1)!)^{k}(c+1)c^{k-1}}.$$

C. *Case $d = 1$ and $c \geq 1$:*

In w:

- 1 color class contains $c-1$ elements
- $k-3$ color classes contain c elements
- 2 color classes contain $c+1$ elements

The frequency of this case is:

$$k\binom{k-1}{2}\frac{t!}{(c-1)!\,((c+1)!)^{2}(c!)^{k-3}}$$

D. *Case $d = k-1$:*

In w:

- 2 color classes contain c elements
- $k-3$ color classes contain $c+1$ elements
- 1 color class contains $c+2$ elements

The frequency of this case is:

$$k\binom{k-1}{2}\frac{t!}{(c+2)!\,((c+1)!)^{k-3}(c!)^{2}}$$

As a consequence, the frequency of $q_{min}+1$ can be calculated as a function of t and k (using the proper case).

VIII. Frequency of $q_{min}+2$

The bounds can be further improved by calculating the value and the frequency of the third smallest q. Similarly to the previous section, we start from a partial coloring w with $q(w) = q_{min}+2$, and we move to another partial coloring w' with $q(w') = q_{min}$. There are two different ways: by using either one or two correction moves.

A. *Using one correction move*

In this case, in accordance with Proposition 6, $q(w) - q(w') = A - B - 1 = 2$, and with Proposition 7, in w $A_{max} = c+2$ and $B_{min} = c-1$. Therefore, in w:

- 1 color class contains $c-1$ elements (thus $c-1 \geq 0$)
- $k-d-1$ color classes contain c elements
- $d-1$ color classes contain $c+1$ elements (thus $d \geq 1$)
- 1 color class contains $c+2$ elements

The frequency of this case is:

$$k\binom{k-1}{1}\binom{k-2}{d-1}\frac{t!}{(c-1)!\,(c!)^{k-d-1}((c+1)!)^{d-1}(c+2)!}$$

B. *Using two correction moves*

After the first correction move $q(w'') = q_{min}+1$. In this case $q(w) - q(w'') = q(w'') - q(w') = 1$. Hence, after each correction move, the color classes with the changed number of elements contain equal number of elements.

Proposition 10. *In w, there is no color class with more than $c+2$ elements.*

Proof: Using contradiction we assume, that there is a color class with at least $c+3$ elements. Hence in both correction moves a vertex should be moved from this color class to another. Meanwhile there should not arise a color class with more than $c+1$ elements. Then in the first correction move $q(w) - q(w'') > 1$. ∎

Proposition 11. *In w, there are at most two color classes with $c+2$ elements.*

Proof: Similarly, at least three correction moves would be needed otherwise. ■

We further split this case by the number of color classes containing $c+2$ elements.

1) If there are two color classes with $c+2$ elements: In w:

- $k-d+2$ color classes contain c elements
- $d-4$ color classes contain $c+1$elements (thus $d \geq 4$)
- 2 color classes contain $c+2$ elements

The frequency of this case is:

$$\binom{k}{2}\binom{k-2}{d-4}\frac{t!}{((c+2)!)^2\,((c+1)!)^{d-4}\,(c!)^{k-d+2}}$$

2) If there is one color class with $c+2$ elements: The same way as earlier, in w:

- 1 color class contains $c-1$ elements (thus $c \geq 1$)
- $k-d-1$ color classes contain c elements
- $d-1$ color classes contain $c+1$elements (thus $d \neq 0$)
- 1 color class contains $c+2$ elements

The frequency of this case is:

$$k\binom{k-1}{1}\binom{k-2}{d-1}\frac{t!}{(c-1)!\,(c!)^{k-d-1}\,((c+1)!)^{d-1}\,(c+2)!}$$

3) If there is no color class with $c+2$ elements: In w:

- 2 color classes contain $c-1$ elements (thus $c \geq 1$)
- $k-d-4$ color classes contain c elements (thus $d \leq k-4$)
- $d+2$ color classes contain $c+1$elements

The frequency of this case is:

$$\binom{k}{2}\binom{k-2}{d+2}\frac{t!}{((c+1)!)^{d+2}\,(c!)^{k-d-4}\,((c-1)!)^2}$$

Using the proper case, the value of R_{min+2} can always be calculated. Care needs to be taken though as two correction moves might be substituted with a single one. Specifically, the case in Subsection VIII-A is equivalent to the case of Subsubsection VIII-B2. Otherwise, the cases are disjoint.

IX. Putting the pieces together

Let $R_{min+1} := R(q_{min+1}, t, k)$ and $R_{min+2} := R(q_{min+2}, t, k)$. The best lower and upper bounds are:

$$\begin{aligned}E(Y) \leq \sum_{t=0}^{n} & R_{min}(1-p)^{q_{min}} + \\ &+ R_{min+1}(1-p)^{q_{min+1}} + R_{min+2}(1-p)^{q_{min+2}} + \\ &+ (k^t - R_{min} - R_{min+1} - R_{min+2})(1-p)^{q_{min}+3}.\end{aligned}$$

and

Theorem 12.

$$\begin{aligned}E(Y) \geq \sum_{t=0}^{n} & R_{min}(1-p)^{q_{min}} + \\ &+ R_{min+1}(1-p)^{q_{min+1}} + R_{min+2}(1-p)^{q_{min+2}} + \\ &+ (k^t - R_{min} - R_{min+1} - R_{min+2})(1-p)^{\widehat{q_3}}\end{aligned}$$

where $\widehat{q_3} = \frac{k^t\bar{q} - R_{min}q_{min} - R_{min+1}(q_{min}+1) - R_{min+2}(q_{min}+2)}{k^t - R_{min} - R_{min+1} - R_{min+2}}$.

Proof: Let $T_t^{(1)} := \{w \in T_t : q(w) = q_{min}\}$, $T_t^{(2)} := \{w \in T_t : q(w) = q_{min}+1\}$, $T_t^{(3)} := \{w \in T_t : q(w) = q_{min}+2\}$ and $T_t^{(3+)} := \{w \in T_t : q(w) > q_{min}+2\}$ on the analogy of Theorem 9. Hence, $|T_t^{(1)}| = R_{min}$, $|T_t^{(2)}| = R_{min+1}$, $|T_t^{(3)}| = R_{min+2}$ and $|T_t^{(3+)}| = k^t - R_{min} - R_{min+1} - R_{min+2}$. Clearly,

$$\sum_{w \in T_t^{(1)} \cup T_t^{(2)} \cup T_t^{(3)}} q(w) =$$
$$R_{min}q_{min} + R_{min+1}(q_{min}+1) + R_{min+2}(q_{min}+2)$$

and

$$\sum_{w \in T_t^{(3+)}} q(w) =$$
$$k^t\bar{q} - R_{min}q_{min} - R_{min+1}(q_{min}+1) - R_{min+2}(q_{min}+2).$$

Using Jensen's inequality,

$$\sum_{w \in T_t^{(3+)}} (1-p)^{q(w)} \geq \left|T_t^{(3+)}\right| (1-p)^{\frac{1}{\left|T_t^{(3+)}\right|}\sum_{w \in T_t^{(3+)}} q(w)} =$$
$$(k^t - R_{min} - R_{min+1} - R_{min+2})\cdot$$
$$\cdot (1-p)^{\frac{k^t\bar{q} - R_{min}q_{min} - R_{min+1}(q_{min}+1) - R_{min+2}(q_{min}+2)}{k^t - R_{min} - R_{min+1} - R_{min+2}}}.$$

Together with the other terms, we get the stated bound. ■

Because of space constraints, we do not include the calculation of the bounds determined by $q_{min}+1$ (without the help of $q_{min}+2$) and for the inherent $\widehat{q_2} = \frac{k^t\bar{q} - R_{min}q_{min} - R_{min+1}(q_{min}+1)}{k^t - R_{min} - R_{min+1}}$.

Clearly, we could continue the above procedure and further improve the bounds by also calculating the term of $q_{min}+3$, then $q_{min}+4$ etc. However, it is also clear from the above that the calculation becomes significantly more complex with each step, and on the other hand, the gain is decreasing with every step (see Section X).

X. Numerical comparison of the bounds

In order to assess how good the different lower and upper bounds are, we compared them numerically for different values of the control parameters n, k, p. Here, we show the comparison for fix values of n and k, as a function of p. In order to enhance visibility, we include two figures (note the exponential scale on the y axis in both cases): one for small values of p (Fig. 3) and one for high values of p (Fig. 4). As can be seen, both the upper bounds and the lower bounds are becoming better and better.

The shown bounds are as follows:

- 1st upper bound: bounding q_{min}
- 2nd upper bound: calculating q_{min} term separately
- 3rd upper bound: calculating $q_{min}+1$ term separately
- 4th upper bound: calculating $q_{min}+2$ term separately
- 5th lower bound: calculating $q_{min}+2$ term separately
- 4th lower bound: calculating $q_{min}+1$ term separately

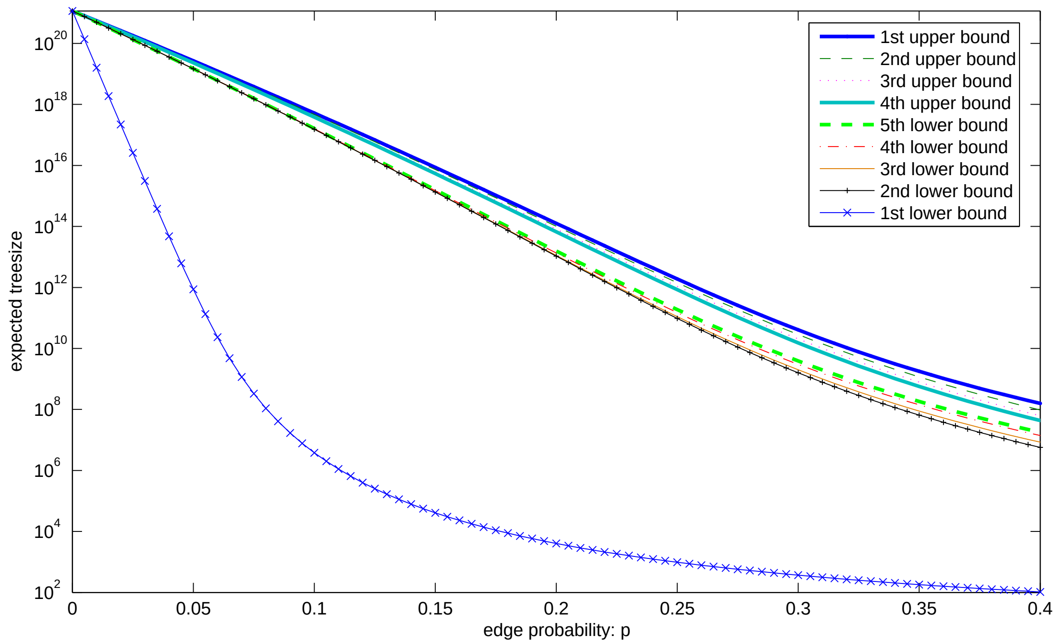

Fig. 3. Comparison of the presented lower and upper bounds for small values of p, with $n = 30$ and $k = 5$.

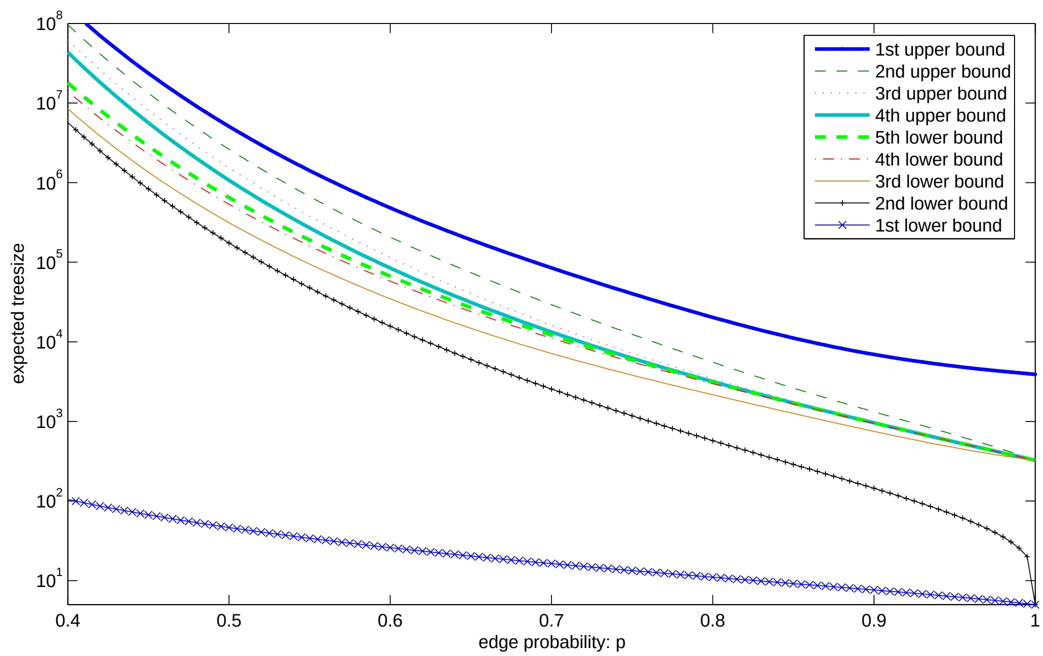

Fig. 4. Comparison of the presented lower bounds and upper bounds for high values of p, with $n = 30$ and $k = 5$.

- 3rd lower bound: calculating q_{min} term separately
- 2nd lower bound: using Jensen's inequality
- 1st lower bound: bounding q_{max}

Fig. 5 presents only the best bounds, together with the bounds of Bender and Wilf [3]. As can be seen, the new bounds are much closer to each other than the original bounds. (The shape and relative position of the curves are similar for other values of n and k as well.) The exact location of the true expected tree size is currently not known, but a method for determining it is presented in [13].

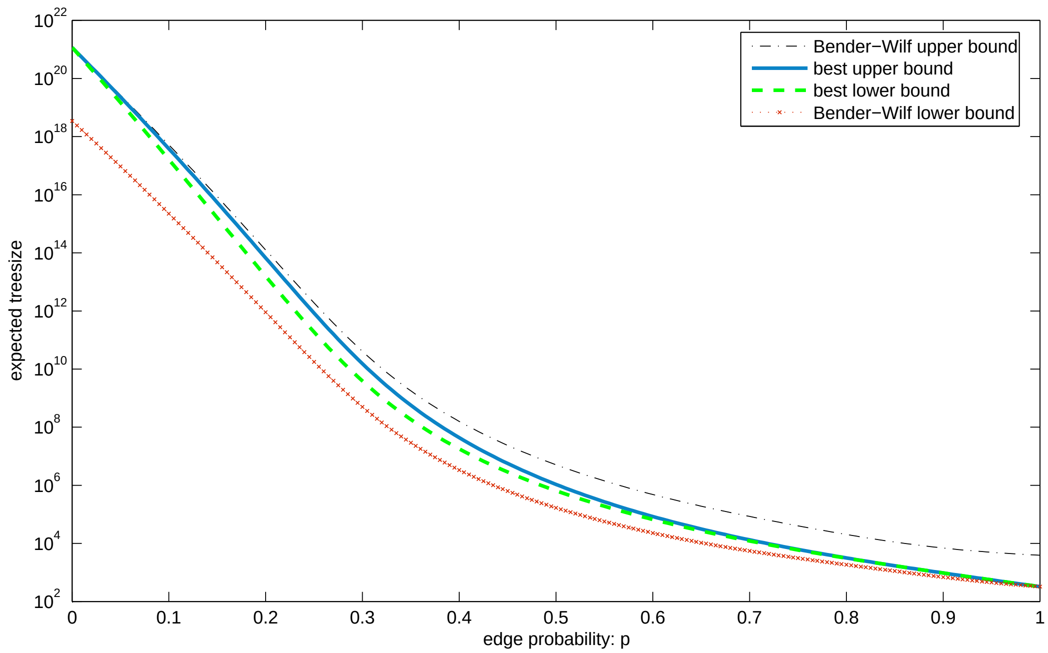

Fig. 5. Comparison of the presented best lower bound and best upper bound with the bounds of Bender and Wilf [3] for $n = 30$ and $k = 5$.

XI. Conclusion and Future Work

We have investigated the complexity of a typical backtrack search for coloring random graphs with k colors. Using the expected size of the search tree as the measure of complexity, we derived lower and upper bounds for the complexity. We showed empirical evidence that these bounds are much closer to each other than previously known bounds.

In this paper, we only dealt with uncolorable problem instances. Our future work will focus on extending the presented results to colorable problem instances.

Bender and Wilf [3] also presented lower and upper bounds on the jth moment of the number of visited nodes in the search tree. The variance is particularly interesting to better judge the algorithm's performance. It remains a future research direction to investigate how the methods presented in this paper can be used to improve Bender and Wilf's bounds on higher moments.

Acknowledgements

This work was partially supported by the Hungarian National Research Fund and the National Office for Research and Technology (Grant Nr. OTKA 67651).

References

[1] Dimitris Achlioptas and Assaf Naor. The two possible values of the chromatic number of a random graph. In *36th ACM Symposium on Theory of Computing (STOC '04)*, pages 587–593, 2004.

[2] Noga Alon and Michael Krivelevich. The concentration of the chromatic number of random graphs. *Combinatorica*, 17(3):303–313, 1997.

[3] Edward A. Bender and Herbert S. Wilf. A theoretical analysis of backtracking in the graph coloring problem. *Journal of Algorithms*, 6(2):275–282, 1985.

[4] Béla Bollobás. The chromatic number of random graphs. *Combinatorica*, 8(1):49–55, 1988.

[5] Daniel Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):251–256, 1979.

[6] Preston Briggs, Keith D. Cooper, and Linda Torczon. Improvements to graph coloring register allocation. *ACM Transactions on Programming Languages and Systems*, 16(3):428–455, 1994.

[7] Peter Cheeseman, Bob Kanefsky, and William M. Taylor. Where the really hard problems are. In *12th International Joint Conference on Artificial Intelligence (IJCAI '91)*, pages 331–337, 1991.

[8] Pál Erdős and Alfréd Rényi. On the evolution of random graphs. *Magyar Tud. Akad. Mat. Kutató Int. Közl*, 5:17–61, 1960.

[9] Michael R. Garey, David S. Johnson, and L. J. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1:237–267, 1976.

[10] G. R. Grimmett and C. J. H. McDiarmid. On colouring random graphs. *Mathematical Proceedings of the Cambridge Philosophical Society*, 77(2):313–324, 1975.

[11] Tad Hogg. Refining the phase transition in combinatorial search. *Artificial Intelligence*, 81(1-2):127 – 154, 1996.

[12] Tomasz Luczak. The chromatic number of random graphs. *Combinatorica*, 11(1):45–54, 1991.

[13] Zoltán Á. Mann and Anikó Szajkó. Determining the expected runtime of exact graph coloring. In *Mini-conference on Applied Theoretical Computer Science (MATCOS)*, 2010.

[14] Zoltán Á. Mann and Tamás Szép. BCAT: A framework for analyzing the complexity of algorithms. In *8th IEEE International Symposium on Intelligent Systems and Informatics*, pages 297–302, 2010.

[15] Zoltán Ádám Mann and András Orbán. Optimization problems in system-level synthesis. In *3rd Hungarian-Japanese Symposium on Discrete Mathematics and Its Applications*, pages 222–231, 2003.

[16] Nirbhay K. Mehta. The application of a graph coloring method to an examination scheduling problem. *Interfaces*, 11(5):57–65, 1981.

[17] Rémi Monasson. On the analysis of backtrack procedures for the coloring of random graphs. In E. Ben-Naim, H. Frauenfelder, and Z. Toroczkai, editors, *Complex Networks*, pages 235–254. Springer, 2004.

[18] Tamás Szép and Zoltán Á. Mann. Graph coloring: the more colors,

the better? In *11th IEEE International Symposium on Computational Intelligence and Informatics*, pages 119–124, 2010.

[19] Jonathan S. Turner. Almost all k-colorable graphs are easy to color. *Journal of Algorithms*, 9(1):63–82, 1988.

[20] Herbert S. Wilf. Backtrack: an O(1) expected time algorithm for the graph coloring problem. *Information Processing Letters*, 18:119–121, 1984.