Asymptotic behaviour of the complexity of coloring sparse random graphs*

ZOLTÁN ÁDÁM MANN

Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Magyar tudósok körútja 2., 1117 Budapest, Hungary
e-mail: zoltan.mann@gmail.com

ANIKÓ SZAJKÓ

Department of Computer Science and Information Theory
Budapest University of Technology and Economics
Magyar tudósok körútja 2., 1117 Budapest, Hungary
e-mail: szajko.aniko@gmail.com

Abstract: The behaviour of a backtrack algorithm for graph coloring is well understood for large random graphs with constant edge density. However, sparse graphs, in which the edge density decreases with increasing graph size, are more common in practice. Therefore, in this paper we analyze the expected runtime of a usual backtrack search to color such random graphs, when the size of the graph tends to infinity. Contrary to the case of constant edge density, where the expected runtime is known to be $O(1)$, here we prove that the expected runtime tends to infinity in this case. We also examine when the expected runtime grows polynomially or exponentially, depending on the edge density function. Besides, we also investigate the asymptotic behaviour of the expected number of solutions in this model.

Keywords: graph coloring, average-case complexity, search tree, random graphs, backtracking

1 Introduction

Graph coloring is an important combinatorial optimization problem with many applications in engineering, such as register allocation, frequency assignment, pattern matching and scheduling [11, 4, 7]. Accordingly, graph coloring has been intensively researched.

One of the main tools to mathematically investigate graph coloring is to study the coloring of random graphs. Usually, the $G_{n,p}$ random graph model is used [5]. Through the research results of the last couple of decades, we can almost exactly determine the chromatic number of random graphs when the size of the graph tends to infinity [12, 6, 2, 1].

Another related question is the performance of graph coloring algorithms on random graphs. In 1984, Wilf proved the surprising result that the expected runtime of a standard backtrack algorithm is bounded even if the size of the graph tends to infinity [13]. That is, the average-case complexity of this algorithm is $O(1)$, although its worst-case complexity is exponential in the size

*This paper was published in: *Proceedings of the 7th Hungarian-Japanese Symposium on Discrete Mathematics and Its Applications*, pages 399-408, 2011.

of the graph. Bender and Wilf provided a more detailed analysis of the asymptotic distribution of the algorithm's runtime [3]. In our recent research, we refined the results of Bender and Wilf: with detailed examinations, we can quite precisely predict the expected runtime of the usual backtrack algorithm for a random graph, as a function of the number of vertices, the number of colors, and the edge density [9, 10].

The above results apply to random graphs where the edge density p is constant. Note that such graphs are with high probability very dense with $\Theta(n^2)$ edges. However, sparse graphs with varying edge density $p = p(n)$ depending on their size are often a subject of research work, since they are more common in practice [8]. Therefore, in this paper, we investigate the asymptotic behavior of the expected runtime of the backtrack algorithm in cases of different $p(n)$ functions tending to 0. As a machine independent measure of complexity, we estimate the expected number of visited nodes in the algorithm's search tree. Our main results are:

- We prove that, in contrast to Wilf's Theorem [13], the expected size of the search tree tends to infinity in case of any arbitrary sequence $p(n) \to 0$.
- We determine how rapidly the expected size of the search tree tends to infinity. In particular, it is exponential for $p(n) = 1/n$, but polynomial for $p(n) = 1/\log n$. That is, for the latter case, the algorithm's average-case complexity is polynomial.
- As a by-product, we also obtained the asymptotic behaviour of the expected number of solutions for different $p(n)$ sequences.

2 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, 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.

As in [3, 10, 9], we assume that the algorithm doesn't stop even if it found a proper solution. Therefore, our results are accurate only for uncolorable graphs; for colorable graphs, they are just upper estimates.

3 Notations and previous results

We define a random variable Y to be the number of visited nodes in T. In [10], we proved the following lower bound:

$$E(Y) \geq \sum_{t=0}^{n} k^t (1-p)^{\frac{t^2-t}{2k}}, \tag{1}$$

and an upper bound:

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

Moreover, the number of solutions (S) is equivalent with the number of visited nodes in the last level of the search tree. Accordingly,

$$k^n (1-p)^{\frac{n^2-n}{2k}} \leq E(S) \leq k^n \cdot (1-p)^{\frac{1}{2}\left(\frac{n^2}{k}-n\right)}.$$

4 Expected size of the search tree

The following two lemmas are a refinement of Lemma 3 in [3].

Lemma 1. *For any $a, b > 0$*

$$\sum_{t=0}^{n} e^{-at^2} e^{bt} > \begin{cases} \frac{1}{\sqrt{a}} e^{\frac{b^2}{4a}} \left(\int_{\frac{-b-2a}{2\sqrt{a}}}^{\frac{2a(n+1)-b}{2\sqrt{a}}} e^{-u^2}\, du - \sqrt{a} \right) > \frac{b}{2a} e^{-b-a} & \text{if } 2an - b > 0, \\ \frac{(n+1)}{2} \left(e^{-\frac{a+an^2+2b-2nb-2an}{4}} + e^{-b-a} \right) > (n+1) e^{-b-a} & \text{if } 2an - b \leq 0. \end{cases}$$

Proof. Let $x = t - \frac{b}{2a}$, hence $-ax^2 = -at^2 + bt - \frac{b^2}{4a}$. Besides, let $u = \sqrt{a}x$, thus $u^2 = ax^2$. Accordingly:

$$\sqrt{a} e^{\frac{-b^2}{4a}} \sum_{t=0}^{n} e^{-at^2} e^{bt} = \sqrt{a} \sum_{t=0}^{n} e^{-ax^2(t)} = \sqrt{a} \sum_{x=-\frac{b}{2a}}^{n-\frac{b}{2a}} e^{-ax^2} = \sqrt{a} \sum_{u=-\frac{b}{2\sqrt{a}}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2},$$

since $\frac{-b}{2a} \leq x \leq n - \frac{b}{2a} \iff -\frac{b}{2\sqrt{a}} \leq \sqrt{a}x \leq \sqrt{a}n - \frac{b}{2\sqrt{a}}$. x and u might denote fractions too, the summations range over all x and u for which $x = i - \frac{b}{2a}$, $u = i - \frac{b}{2\sqrt{a}}$, where i is an integer between 0 and n. The received sum might be regarded as an upper estimation of an integral by step $\sqrt{a}$ and an optional rest term. Moreover, the area under the integral curve is greater than the area of one or two rectangles under that.

If $\frac{2an-b}{2\sqrt{a}} > 0:$

$$\sqrt{a}\sum_{u=-\frac{b}{2\sqrt{a}}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2} > \int_{\frac{-b}{2\sqrt{a}}-\sqrt{a}}^{\frac{2an-b}{2\sqrt{a}}+\sqrt{a}} e^{-u^2}\,du - 1\cdot\sqrt{a} >$$

$$> \left(\frac{b}{2\sqrt{a}}+\sqrt{a}-\sqrt{a}\right)e^{-\left(\frac{-b}{2\sqrt{a}}-\sqrt{a}\right)^2} = \frac{b}{2\sqrt{a}}e^{-\frac{b^2+4ab+4a^2}{4a}} = \frac{b}{2\sqrt{a}}e^{-\frac{b^2}{4a}-b-a}.$$

If $\frac{2an-b}{2\sqrt{a}} \leq 0:$

$$\sqrt{a}\sum_{u=-\frac{b}{2\sqrt{a}}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2} > \int_{\frac{-b}{2\sqrt{a}}-\sqrt{a}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2}\,du >$$

$$> \frac{(n+1)}{2}\sqrt{a}\left(e^{-\left(\frac{-b-2a}{4\sqrt{a}}+\frac{2an-b}{4\sqrt{a}}\right)^2}+e^{-\left(\frac{-b-2a}{2\sqrt{a}}\right)^2}\right) =$$

$$= \frac{(n+1)}{2}\sqrt{a}\left(e^{-\left(\frac{-b-a+an}{2\sqrt{a}}\right)^2}+e^{-\left(\frac{-b-2a}{2\sqrt{a}}\right)^2}\right) =$$

$$= \frac{(n+1)}{2}\sqrt{a}\left(e^{-\frac{b^2+a^2+a^2n^2+2ab-2anb-2a^2n}{4a}}+e^{-\frac{b^2+4ab+4a^2}{4a}}\right) >$$

$$> (n+1)\sqrt{a}e^{-\left(\frac{-b-2a}{2\sqrt{a}}\right)^2} = (n+1)\sqrt{a}e^{-\frac{b^2+4ab+4a^2}{4a}} = (n+1)\sqrt{a}e^{-\frac{b^2}{4a}-b-a}.$$

□

Lemma 2. *For any $a, b > 0$*

$$\sum_{t=0}^{n} e^{-at^2}e^{bt} < \frac{1}{\sqrt{a}}e^{\frac{b^2}{4a}}\left(\int_{\frac{-b}{2\sqrt{a}}}^{\frac{2an-b}{2\sqrt{a}}} e^{-u^2}\,du + \sqrt{a}\right).$$

Proof. Similar to the proof of Lemma 1 and using its notations, the received sum is a lower estimation of the summation of integrals by step $\sqrt{a}$ and a rest term.

$$\sqrt{a}e^{\frac{-b^2}{4a}}\sum_{t=0}^{n} e^{-at^2}e^{bt} = \sqrt{a}\sum_{u=-\frac{b}{2\sqrt{a}}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2} < \int_{\frac{-b}{2\sqrt{a}}}^{\sqrt{a}n-\frac{b}{2\sqrt{a}}} e^{-u^2}\,du + 1\cdot\sqrt{a}$$

□

Theorem 3. *In case of any sequence $0 \leq p(n) = p_n \leq 1$ tending to 0, the expected size of the search tree tends to infinity when $n \to \infty$.*

Proof. From inequality (1),

$$E(Y) \geq \lim_{n\to\infty}\sum_{t=0}^{n} k^t\cdot(1-p_n)^{\frac{t^2-t}{2k}} = \lim_{n\to\infty}\sum_{t=0}^{n}\left((1-p_n)^{\frac{1}{2k}}\right)^{t^2}\cdot\left(k\,(1-p_n)^{\frac{-1}{2k}}\right)^t.$$

In this formula, $(1-p_n)^{\frac{1}{2k}} < 1$ and $k(1-p_n)^{\frac{-1}{2k}} > 1$. Therefore, $\exists a, b > 0$, so that $(1-p_n)^{\frac{1}{2k}} = e^{-a}$ and $k(1-p_n)^{\frac{-1}{2k}} = e^{b}$. In this way, $a = -\ln(1-p_n)^{\frac{1}{2k}}$, $b = \ln k(1-p_n)^{\frac{-1}{2k}}$. It follows that $\lim_{n\to\infty} a = \lim_{n\to\infty} -\ln(1-p_n)^{\frac{1}{2k}} = +0$ and $\lim_{n\to\infty} b = \lim_{n\to\infty} \ln k + \ln(1-p_n)^{\frac{-1}{2k}} = \ln k$.

Applying Lemma 1, we obtain

$$\sum_{t=0}^{n}\left((1-p_n)^{\frac{1}{2k}}\right)^{t^2}\cdot\left(k(1-p_n)^{\frac{-1}{2k}}\right)^{t} = \sum_{t=0}^{n} e^{-at^2}e^{bt} > \begin{cases} \frac{b}{2a}e^{-b-a} & \text{if } \frac{2an-b}{2\sqrt{a}} > 0, \\ (n+1)e^{-b-a} & \text{if } \frac{2an-b}{2\sqrt{a}} \le 0. \end{cases}$$

Therefore,

$$\lim_{n\to\infty} E(Y) > \begin{cases} \lim_{n\to\infty} \frac{b}{2a}e^{-b-a} = \infty & \text{if } \lim_{n\to\infty} \frac{2an-b}{2\sqrt{a}} > 0, \\ \lim_{n\to\infty}(n+1)e^{-b-a} = \infty & \text{if } \lim_{n\to\infty} \frac{2an-b}{2\sqrt{a}} \le 0. \end{cases}$$

$\square$

In the next theorem, we examine the rate by which the expected number of visited nodes of the search tree tends to infinity.

Theorem 4.

$$E(Y) = \begin{cases} \Theta\left(\frac{1}{\sqrt{p_n}}(c)^{\frac{1}{p_n}}\right) & \textit{if } \lim_{n\to\infty} np_n > k\ln k \ (\textit{where } c = k^{\frac{k\ln k}{2}}), \\ O(nk^n) \ \textit{and}\ \Omega(nc^n) & \textit{if } \lim_{n\to\infty} np_n \le k\ln k \ (\textit{where } c = k^{\frac{3}{8}}). \end{cases}$$

Proof. $\lim_{n\to\infty} 2an-b = \lim_{n\to\infty} -2n\ln(1-p_n)^{\frac{1}{2k}} - \ln k = \lim_{n\to\infty} \frac{-np_n}{k}\ln(1-p_n)^{\frac{1}{p_n}} - \ln k = \lim_{n\to\infty} \frac{np_n}{k} - \ln k > 0 \Leftrightarrow \lim_{n\to\infty} np_n > k\ln k$.

1. Case $2an-b > 0$:

   From Lemma 1 and Theorem 3,

$$E(Y) > \frac{1}{\sqrt{a}}e^{\frac{b^2}{4a}}\left(\int_{\frac{-b-2a}{2\sqrt{a}}}^{\frac{2a(n+1)-b}{2\sqrt{a}}} e^{-u^2}\,du - \sqrt{a}\right).$$

   In view of $\lim_{n\to\infty} \frac{-b-2a}{2\sqrt{a}} = -\infty$ and $\frac{2a(n+1)-b}{2\sqrt{a}} > 0$,

$$\frac{\sqrt{\pi}}{2} = \lim_{n\to\infty}\int_{-\infty}^{0} e^{-u^2}\,du - 0 < \lim_{n\to\infty}\int_{\frac{-b-2a}{2\sqrt{a}}}^{\frac{2a(n+1)-b}{2\sqrt{a}}} e^{-u^2}\,du - \sqrt{a} \le \lim_{n\to\infty}\int_{-\infty}^{\infty} e^{-u^2}\,du = \sqrt{\pi}.$$

   Thus,

$$E(Y) = \Omega\left(\frac{1}{\sqrt{a}}\left(e^{b^2}\right)^{\frac{1}{4a}}\right) = \Omega\left(\frac{1}{\sqrt{-\frac{p_n}{2k}\ln(1-p_n)^{\frac{1}{p_n}}}}\left(k^{\ln k}\right)^{\frac{2k}{-4p_n\ln(1-p_n)^{\frac{1}{p_n}}}}\right) =$$
$$= \Omega\left(\sqrt{\frac{2k}{p_n}}\left(k^{\frac{k\ln k}{2}}\right)^{\frac{1}{p_n}}\right) = \Omega\left(\frac{1}{\sqrt{p_n}}(c)^{\frac{1}{p_n}}\right).$$

   In a similar way, from Lemma 2, we get $E(Y) = O\left(\frac{1}{\sqrt{p_n}}(c)^{\frac{1}{p_n}}\right)$.

2. Case $2an - b \leq 0$:

   Applying Lemma 1, $E(Y) > \frac{(n+1)}{2}\left(e^{-\frac{a+an^2+2b-2nb-2an}{4}} + e^{-b-a}\right)$.

   As $0 < np_n \leq k \ln k \Leftrightarrow 0 > \frac{-n^2 p_n}{8k} \geq \frac{-n \ln k}{8}$,

$$E(Y) = \Omega\left(n\left(e^{-\frac{a+an^2-2nb-2an}{4}} + e^{-b-a}\right)\right) = \Omega\left(ne^{-\frac{a+an^2-2nb-2an}{4}}\right) + \Omega(n) =$$

$$= \Omega\left(ne^{\frac{-p_n}{8k} - \frac{p_n}{8k}n^2 + \frac{n \ln k}{2} + \frac{p_n}{4k}n}\right) + \Omega(n) = \Omega\left(ne^{\frac{-n^2 p_n}{8k}} k^{\frac{n}{2}}\right) + \Omega(n) =$$

$$= \Omega\left(ne^{\frac{-n \ln k}{8}} k^{\frac{n}{2}}\right) + \Omega(n) = \Omega\left(nk^{\frac{-n}{8}} k^{\frac{n}{2}}\right) + \Omega(n) = \Omega\left(n\left(k^{\frac{3}{8}}\right)^n\right) = \Omega(nc^n).$$

In addition, $E(Y) = O(nk^n)$, since the search tree has $n+1$ levels and at most k^n nodes on each level. □

As a consequence, the complexity of the algorithm is exponential invariably in the second case, but can be polynomial in the first case.

E. g. assuming $p_n = \frac{d}{n^\alpha}$, where d and α are positive constants: $\lim_{n\to\infty} \frac{d}{n^{\alpha-1}} > k \ln k \Leftrightarrow \frac{d}{k \ln k} > \lim_{n\to\infty} n^{\alpha-1} \Leftrightarrow 0 < \alpha < 1$, or $\alpha = 1$ and $d > k \ln k$. Therefore,

$$E(Y) = \begin{cases} \Theta\left(\sqrt{\frac{n^\alpha}{d}}\left(k^{\frac{k \ln k}{2}}\right)^{\frac{n^\alpha}{d}}\right) & \text{if } 0 < \alpha < 1, \text{ or } \alpha = 1 \text{ and } d > k \ln k, \\ O(nk^n) \text{ and } \Omega\left(nk^{\frac{3n}{8}}\right) & \text{if } 1 < \alpha, \text{ or } \alpha = 1 \text{ and } d \leq k \ln k. \end{cases}$$

An example for the polynomial case is $p_n = \frac{d}{\ln n}$. Here, we have $\lim_{n\to\infty} \frac{d}{\ln n} n = \lim_{n\to\infty} \frac{d}{\ln \sqrt[n]{n}} = \infty$. Thus,

$$E(Y) = \Theta\left(\sqrt{\frac{\ln n}{d}}\left(k^{\frac{k \ln k}{2}}\right)^{\frac{\ln n}{d}}\right) = \Theta\left(\sqrt{\frac{\ln n}{d}} n^{\frac{k \ln^2 k}{2d}}\right),$$

which is indeed polynomial in n.

5 Expected number of solutions

We can also use the presented machinery to estimate the asymptotic number of expected solutions:

Proposition 5.

$$\lim_{n\to\infty} E(S) = \begin{cases} \infty & \text{if } p_n < \frac{2k \ln k}{n-1} \\ 0 & \text{if } p_n > \frac{2k \ln k}{n-k} \end{cases} \quad (\textit{for all sufficiently large } n).$$

Proof. Applying the results of Section 3, $E(S) \geq k^n (1-p_n)^{\frac{n^2-n}{2k}}$. Therefore,

$$\lim_{n\to\infty} E(S) \geq \lim_{n\to\infty} k^n (1-p_n)^{p_n \frac{n^2-n}{2kp_n}} = \lim_{n\to\infty} k^n (e)^{-p_n \frac{n^2-n}{2k}} = \lim_{n\to\infty} \left(\frac{k}{e^{p_n \frac{n-1}{2k}}}\right)^n.$$

$\lim_{n\to\infty} \frac{k}{e^{p_n \frac{n-1}{2k}}} > 1 \Leftrightarrow \ln k > p_n \frac{n-1}{2k} \Leftrightarrow \frac{2k \ln k}{n-1} > p_n$ as $n \to \infty$.

Analogously,

$$\lim_{n\to\infty} E(S) \leq \lim_{n\to\infty} k^n (1-p_n)^{p_n \frac{n^2-nk}{2kp_n}} = \lim_{n\to\infty} k^n e^{-p_n \frac{n^2-nk}{2k}} = \lim_{n\to\infty} \left(\frac{k}{e^{p_n \frac{n-k}{2k}}} \right)^n .$$

$\lim_{n\to\infty} \frac{k}{e^{p_n \frac{n-k}{2k}}} < 1 \Leftrightarrow \frac{2k \ln k}{n-k} < p_n$ as $n \to \infty$. □

For a given p_n, the $\frac{2k \ln k}{n-1} \leq p_n \leq \frac{2k \ln k}{n-k}$ (for all sufficiently large n) case might also be estimated in a similar way.

E.g., let $p_n = \frac{d}{n^\alpha}$, where d and α are positive constants. Assuming $n \to \infty$,
$\frac{d}{n^\alpha} < \frac{2k \ln k}{n-1} \Leftrightarrow n^{1-\alpha} - n^{-\alpha} < \frac{2k \ln k}{d}$ is valid, if and only if $\alpha > 1$, or $\alpha = 1$ and $d < 2k \ln k$,
$\frac{d}{n^\alpha} > \frac{2k \ln k}{n-k} \Leftrightarrow n^{1-\alpha} - kn^{-\alpha} > \frac{2k \ln k}{d}$ is valid, if and only if $0 < \alpha < 1$, or $\alpha = 1$ and $d > 2k \ln k$.
Analyzing the $\alpha = 1$, $d = 2k \ln k$ case separately:
$\lim_{n\to\infty} E(S) \geq \lim_{n\to\infty} k^n \left(1 - \frac{d}{n}\right)^{n \frac{n-1}{2k}} = \lim_{n\to\infty} \left(\frac{k}{\sqrt[2k]{e^d}} \right)^n \sqrt[2k]{e^d} = \sqrt[2k]{e^d} = k$ and
$\lim_{n\to\infty} E(S) \leq \lim_{n\to\infty} k^n \left(1 - \frac{d}{n}\right)^{n \frac{n-k}{2k}} = \left(\frac{k}{\sqrt[2k]{e^d}} \right)^n \sqrt{e^d} = \sqrt{e^d} = k^k$.

To sum up:

$$\lim_{n\to\infty} E(S) = \begin{cases} \infty & \text{if } \alpha > 1, \text{ or } \alpha = 1 \text{ and } d < 2k \ln k, \\ 0 & \text{if } 0 < \alpha < 1, \text{ or } \alpha = 1 \text{ and } d > 2k \ln k. \end{cases}$$

If $\alpha = 1$ and $d = 2k \ln k$, then we have $k \leq E(S) \leq k^k$.

6 Uncolorability and the chromatic number

In this section, we mention some implications of the second part of Proposition 5. Let us assume that $p_n > \frac{2k \ln k}{n-k}$ for all sufficiently large n. Then, by Proposition 5, $\lim_{n\to\infty} E(S) = 0$. Applying Markov's inequality, $\lim_{n\to\infty} Pr(\exists \text{ solution}) = \lim_{n\to\infty} Pr(S \geq 1) \leq \lim_{n\to\infty} E(S) = 0$. In other words, such graphs are uncolorable with probability tending to 1.

As mentioned earlier, our model is precise only for uncolorable graphs. We can now conclude that in this case, our results are accurate.

The second implication is that, with probability tending to 1, the chromatic number must be higher than any k for which $p_n > \frac{2k \ln k}{n-k}$ holds. In the case $p_n = \frac{d}{n}$, this condition reduces to $d > 2k \ln k$. This is perfectly in line with Achlioptas and Naor's result [1]: the chromatic number of a graph with edge density $\frac{d}{n}$ is either k or $k+1$, where k is the smallest integer such that $d < 2k \ln k$, with probability tending to 1 as $n \to \infty$.

7 Numerical examinations

Using the presented approach and the technique for efficiently computing $E(Y)$ and $E(S)$ values that we developed in [9], we can also show the behaviour of these quantities for some representative p_n functions. See Figure 1 for the behaviour of $E(Y)$ and Figure 2 for the behaviour of $E(S)$. Please note the exponential scale on the vertical axis in both figures.

As can be seen, for $p_n = 1/n^5$ and $p_n = 1/n$, both $E(Y)$ and $E(S)$ tend rapidly to infinity. For $p_n = 1/n^{0.5}$, $E(Y)$ grows significantly more slowly, but as we know, still exponentially.

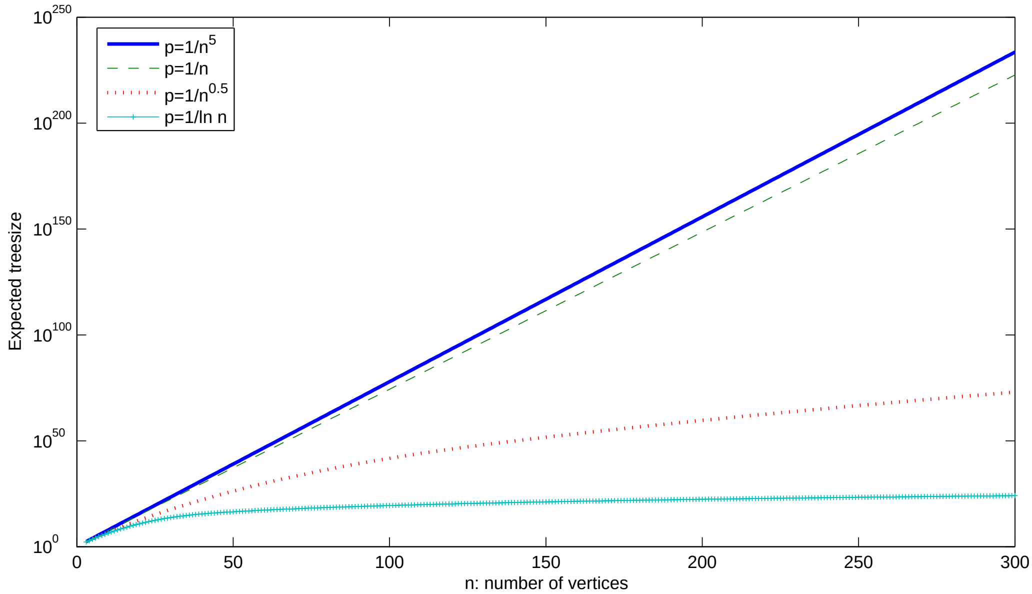

Figure 1: Expected search tree size for different edge density functions ($k = 6$).

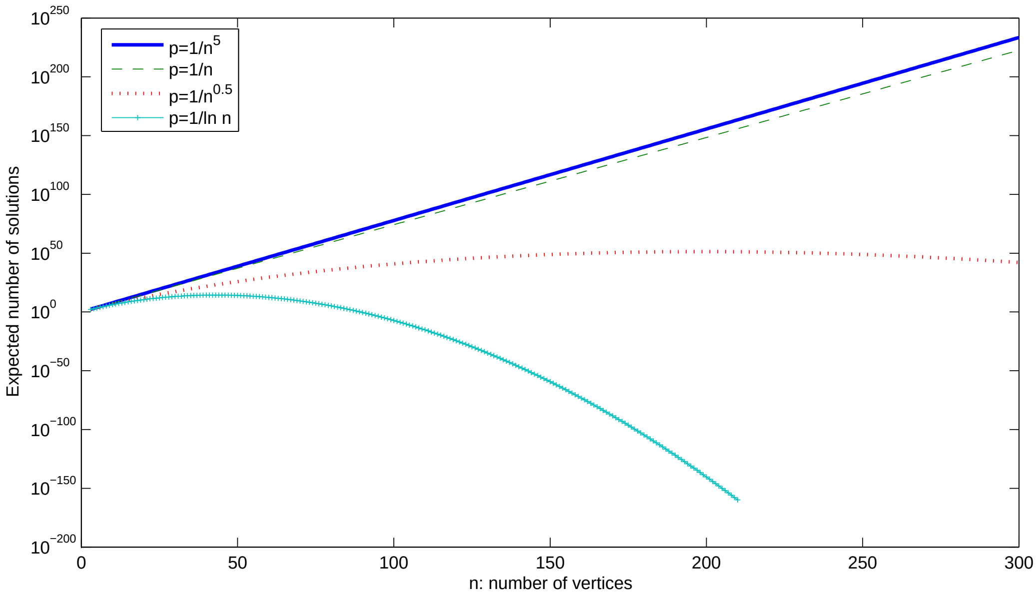

Figure 2: Expected number of solutions for different edge density functions ($k = 6$).

$E(S)$ starts as a monotonously increasing function, but has its maximum at around $n = 200$ and decreases afterwards. As we know, $E(S)$ tends to 0 in this case, but it is interesting to note that $E(S)$ is quite high for graphs with approximately 200 nodes. Finally, when $p_n = 1/\ln n$, then $E(S)$ tends to 0 in a much quicker manner. Also the growth of $E(Y)$ is quite moderate in this case – as we know, it is polynomial in n.

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] 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.

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

[6] Tomasz Luczak. A note on the sharp concentration of the chromatic number of random graphs. *Combinatorica*, 11(3):295–297, 1991.

[7] 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.

[8] Zoltán Ádám Mann, András Orbán, and Viktor Farkas. Evaluating the Kernighan-Lin heuristic for hardware/software partitioning. *International Journal of Applied Mathematics and Computer Science*, 17(2):249–267, 2007.

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

[10] Zoltán Ádám Mann and Anikó Szajkó. Improved bounds on the complexity of graph coloring. In *Proceedings of the 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 347–354, 2010.

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

[12] Eli Shamir and Joel Spencer. Sharp concentration of the chromatic number on random graphs $G_{n,p}$. *Combinatorica*, 7(1):121–129, 1987.

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