COMBINATORICA 7 (3) (1987) 275—284

EXPOSE-AND-MERGE EXPLORATION AND THE CHROMATIC NUMBER OF A RANDOM GRAPH

D. W. MATULA

Received 28 August 1986

Revised 20 March 1987

The expose-and-merge paradigm for exploring random graphs is presented. An algorithm of complexity $n^{O(\log n)}$ is described and used to show that the chromatic number of a random graph for any edge probability $0<p<1$ falls in the interval

$$\left[\left(\frac{1}{2}-\varepsilon\right)\log(1/(1-p))\frac{n}{\log n},\ \left(\frac{2}{3}+\varepsilon\right)\log(1/(1-p))\frac{n}{\log n}\right]$$

with probability approaching unity as $n\to\infty$.

I. Introduction and Summary

A random graph is a pair $(\mathcal{G}, \mathcal{P})$ where $\mathcal{G}$ is a class of graphs and $\mathcal{P}$ is a probability distribution over $\mathcal{G}$. $K_{np}=(\mathcal{G}, \mathcal{P})$ shall denote the random graph where $\mathcal{G}$ is the set of all graphs on n vertices with any particular m-edge graph occuring with probability $p^m(1-p)^{\binom{n}{2}-m}$. K_{np} characterizes the edge probability model of random graphs in that each edge of the complete graph K_n can be considered to be chosen independently with probability p in creating K_{np}.

$\beta(K_{np})$ shall denote the size of the largest independent set in K_{np}, and $\chi(K_{np})$ shall denote the chromatic number of K_{np}. Both of the random variables $\beta(K_{np})$ and $\chi(K_{np})$ have their distribution functions concentrated over a small range for any $0<p<1$ and sufficiently large n, and have been the subject of numerous investigations in the literature (for summaries see [11], [12], [2], and for a more recent result [13]).

We have previously shown the distribution of the independence number $\beta(K_{np})$ to form a spike at just one or two values [8] (see also [9], [12], [1], [2]).

Theorem 1. *For any* $0<p<1$ *and* $\varepsilon>0$, *with* $b=1/(1-p)$,

$$\lim_{n\to\infty} \text{Prob}\,\{\lfloor z(n,p)-\varepsilon\rfloor \leqq \beta(K_{np}) \leqq \lfloor z(n,p)+\varepsilon\rfloor\} = 1, \tag{1}$$

AMS subject classification (1980): 05 c 80 *where*

$$z(n,p) = 2\log_b n - 2\log_b \log_b n + 1 + 2\log_b (e/2). \tag{2}$$

Throughout this paper we shall assume logarithms are to the base $b=1/(1-p)$, and that p is a fixed constant independent of n.

Grimmett and McDiarmid have shown the distribution of the chromatic number $\chi(K_{np})$ to be concentrated in a narrow region not quite as sharply defined as for the independence number [5].

Theorem 2. *For any* $0<p<1$ *and* $\varepsilon>0$,

$$\lim_{n\to\infty} \text{Prob}\left\{\left(\frac{1}{2}-\varepsilon\right)\frac{n}{\log n} \leqq \chi(K_{np}) \leqq (1+\varepsilon)\frac{n}{\log n}\right\} = 1. \tag{3}$$

The upper bound in (3) is obtained constructivly in [5] by analysis of the greedy (or sequential) coloring algorithm [10]. The greedy coloring of K_{np} primarily selects independent sets for each color of only about 1/2 the maximum available size indicated by Theorem 1. It is conjectured that the factor $(1+\varepsilon)$ of the upper bound in (3) can be reduced to $(1/2+\varepsilon)$, so then a minimal coloring of K_{np} would necessarily employ primarily independent sets of essentially optimal size $(2+o(1))\log n$. Employing the second moment method [3], which may be used to obtain the lower bound on $\beta(K_{np})$ in Theorem 1, Korshunov [7] reports the validity of this conjecture in the case corresponding to $p=1/2$ for the constant density model [12] of random graphs. More recently, Shamir and Spencer [13] use martingales to show that the distribution of $\chi(K_{np})$ must be sharply concentrated somewhere within the interval indicated in Theorem 2.

In this paper we introduce a new algorithmic paradigm for exploring random graphs, and provide algorithmic verification that the factor $(1+\varepsilon)$ of the upper bound in (3) can be reduced to $(2/3+\varepsilon)$ for any $0<p<1$. Our approach incorporates the strong lower bound on $\beta(K_{np})$ of Theorem 1 in the following specific form [9], [2].

Theorem 3. *For any* $0<p<1$, *sufficiently large* n, *with* $z(n,p)$ *given by* (2),

$$\text{Prob}\{\beta(K_{np}) \geqq \lfloor z(n,p)-3\rfloor\} \geqq 1-1/n, \tag{4}$$

so also for any $\varepsilon>0$,

$$\text{Prob}\{\beta(K_{np}) \geqq 2\log n-(2+\varepsilon)\log\log n\} \geqq 1-1/n. \tag{5}$$

In Section II we introduce our expose-and-merge paradigm for exploring random graphs. This procedure provides the feature of allowing separate exploration of independently generated random subgraphs, followed by appropriate merger of the random subgraphs to form the random graph K_{np}. In contrast to the sequential greedy coloring algorithm, subsequent selection of independent sets for coloring by this paradigm can implicitly involve exploration over portions of the random graph previously visited and exposed.

The expose-and-merge paradigm is illustrated by two coloring algorithms in Section III. Analysis of the first algorithm provides an alternative proof of the upper bound of Theorem 2. Analysis of the latter algorithm in Section IV yields the improved upper bound that for any $0<p<1$, $0<\varepsilon<1/9$, and sufficiently large n,

$$\text{Prob}\left\{\chi(K_{np}) \leqq \left(\frac{2}{3}+\varepsilon\right)\frac{n}{\log n}\right\} \geqq 1-\frac{1}{n^{\varepsilon/4}}. \tag{6}$$

As a corollary to this result we note in Section V that the greedy coloring algorithm is then not asymptotically optimal, as it requires at least $((3/2)-\varepsilon)\chi(G)$ colors for almost every graph G.

II. Expose-and-Merge Exploration of Random Graphs

K_{np} may be considered to be formed by sequentially determining if each edge of K_n is in K_{np}, such determination for each edge termed the *exposure* of the edge in K_{np}. In our algorithms we shall illustrate this process by labeling each successively exposed edge of K_n "**in**" corresponding to its determined presence in K_{np}, or "**out**" corresponding to its determined absence in K_{np}. Observe that the selection of the k^{th} edge of K_n to be exposed (labeled) *can* depend on the labeling (**in** or out) of the preceding $k-1$ selected and exposed edges, it being sufficient that the exposure (labeling) of the edge itself be independent. It is equivalent to consider "exposure" either to be the random creation of an edge, or the revelation by a referee of the existence or non existence of a particular edge of a previously determined random graph.

In the balance of this section the expose-and-merge process is briefly formalized in more generality than required for our coloring applications, for the purpose of suggesting further applications. The reader may wish to pass on directly to the coloring algorithms of the next section as they are sufficiently self contained and serve to introduce the paradigm by example.

A *sequential exposure rule* for determining the random graph K_{np} is characterized by a complete out-directed binary tree $T=(N, A)$ of depth $\binom{n}{2}$ with labeled node set N and arc set A. Each internal node of N is labeled with an edge of K_n distinct from all other edges of K_n occurring as labels on the path from the root to the node. Traversing to a right child from a node exposes the edge at that node to be in the graph being determined, traversing to the left child exposes the edge to be out of the resulting graph. Each leaf is then labeled with the unique subgraph of K_n whose edges are the labels of nodes whose out-directed arcs on the path to the leaf are directed to the right child. Note then, that for any sequential exposure rule, traversing from the root to a leaf of T by selecting the right child with probability p (and left child with probability $1-p$) at each node reaches a leaf containing a particular m-edge subgraph of K_n with probability $p^m(1-p)^{\binom{n}{2}-m}$, thus realizing the random graph K_{np}.

The expose-and-merge paradigm for exploring random graphs is an extension of the notion of a sequential exposure rule involving phases of independent subgraph exposure followed by merger into a larger random subgraph. Specifically, an *expose-and-merge rule* is a sequential exposure rule $T=(N, A)$, where the root and certain internal nodes of T are also labeled with subgraphs $(V, E)\subset K_n$. Each edge occuring as a label at an internal node of T must then be a member of the edge set E' of the last subgraph (V', E') occurring as a label on the preceding internal nodes of the path from the root of T to that node, unless all members of E' have occurred as labels, in which case the internal node of T is labeled both with a subgraph $(V, E) \subset K_n$ and an edge $e \in E$.

For any subgraph $(V, E) \subset K_n$, let $(V, E)^p$ denote the *independent random subgraph* exposed on vertex set V determined by including each edge of E independently with probability p. Now we shall interpret encountering a subgraph label (V, E) on a particular node in an expose-and-merge rule T as dictating a two step process. First, an independent random subgraph $(V, E)^p$ is *exposed*. In the subsequent merge step, only the edges of E not previously occuring as labels on the internal nodes of the path from the root of T to the particular node are assigned **in** or **out** labels consistent with their membership in $(V, E)^p$, this process leading to a subsequent node of T which is either a leaf or another internal node with a subgraph label. The merger phase of this process may be summarized as follows.

Random Subgraph Merger Principal: Let $E_{in} \subset E$ and $E'_{in} \subset E'$ denote the corresponding member edges of the distinct independent random subgraphs $(V, E)^p$ and $(V', E')^p$ of K_n. Then the graph Merge $((V, E)^p, (V', E')^p)$ formed having vertex set $V \cup V'$ and edge set $E_{in} \cup (E'_{in} - E)$ yields the independent random subgraph denoted by $(V \cup V', E \cup E')^p$, so explicitly

$$(V \cup V', E \cup E')^p = \mathrm{Merge}\big((V, E)^p, (V', E')^p\big). \tag{7}$$

The outcome of a particular expose-and-merge rule T for K_n may then be characterized as the successive merger of a sequence of $k \leq \binom{n}{2}$ independent random subgraphs $(V_1, E_1)^p, (V_2, E_2)^p, \ldots, (V_k, E_k)^p$, where

$$\bigcup_1^k E_i = E(K_n).$$

The evolving random subgraph is iterativly determined by first setting $(V, E)^p \leftarrow (V_1, E_1)^p$, and then setting $(V, E)^p \leftarrow \mathrm{Merge}\big((V, E)^p, (V_i, E_i)^p\big)$ for $i = 2, 3, \ldots, k$, and finally $K_{np} \leftarrow (V, E)^p$. Note in this process that, without bias, each subgraph (V_j, E_j) may be chosen depending on the revealed structure of the preceding random subgraphs $(V_i, E_i)^p$, $i < j$.

Our coloring algorithms in particular shall further choose (V_j, E_j) in an appropriate random manner so that the expected size of the overlap with previously exposed edges,

$$E_j \cap \Big(\bigcup_1^{j-1} E_i\Big),$$

will be readily computable and suitably small. Each $(V_i, E_i)^p$ will be separately explored for a subset of V_i to color, the selection made such that the subset will, with high probability, be an independent set *after* merger of $(V_i, E_i)^p$ into the evolving random graph $(V, E)^p$, and ultimately independent in K_{np}.

The expose-and-merge coloring algorithms of the next section effectivly illustrate the power of this paradigm.

III. Two Expose-and-Merge Graph Coloring Algorithms

Our first graph coloring algorithm provides a straightforward instance of the expose-and-merge paradigm and yields an alternative proof of the upper bound in Theorem 2.

Algorithm A: Given $0<p<1$, $0<\varepsilon<1/2$, and any sufficiently large n, this algorithm generates an instance of the random graph K_{np} and a k-coloring of the vertices of K_{np} where $\text{Prob}\{k\leqq(1+\varepsilon)\,n/\log n\}>1-1/n^{\varepsilon/4}$.

1. [Initialize] $V\leftarrow\{1,2,\ldots,n\}$, $U\leftarrow V$, $E\leftarrow\emptyset$, $k\leftarrow 0$, $m\leftarrow\lfloor(1-\varepsilon/2)\log n\rfloor$. (All n vertices of V are assigned to the uncolored vertex set U, the set of exposed edges E is initialized to empty, the number k of colors employed is initialized to zero. The parameter m is fixed at $\lfloor(1-\varepsilon/2)\log n\rfloor$, and distinct color values will subsequently be assigned to vertex sets only of size m or unity.)

2. [Expose random subgraph] Select an $\lceil n^{1/2-\varepsilon/5}\rceil$-vertex set $S\subset U$ at random and expose a random graph $K_{|S|p}$ on vertex set S. (At this point $K_{|S|p}$ is created ignoring any edges of E that may join vertices of S.)

3. [Determine m-vertex set to color] Select the m-vertex set P at random from the m-membered independent sets of $K_{|S|p}$ if any exist, otherwise select P at random from all m-membered subsets of S. (The procedure of Steps 2 and 3 assures that $P\subset U$ is implicitly chosen uniformly from all m-membered sets of U.)

4. [Color vertices of P] If P is an independent set in $K_{|S|p}$ and if no edge of E joins two vertices of P, assign all vertices of P color value $k+1$, and set $k\leftarrow k+1$. Otherwise assign the m-vertices of P distinct color values $k+1, k+2, \ldots, k+m$. (This assigns a coloring to P consistent both with $K_{|S|p}$ and the previously exposed edges E.)

5. [Merge random subgraph] Any edge $e\in\bar{E}$ joining two vertices of S is added to the exposed set E, and labeled **in** if e is an edge of $K_{|S|p}$, and labeled **out** if e is not an edge of $K_{|\hat{S}|p}$. Also set $U\leftarrow U-P$. (At this point it remains true that any two vertices of $V-U$ assigned the same color value must be joined by an exposed edge of E labeled **out**.)

6. [Uncolored vertex set large?] If $|U|\geqq n/\log^2 n$, return to Step 2. (The cycle of Steps 2—6 will be executed exactly $1+\lfloor n(1-1/\log^2 n)/m\rfloor$ times.)

7. [Complete random graph and vertex coloring.] Assign vertices of U the colors $k+1, k+2, \ldots, k+|U|$, and set $k\leftarrow k+|U|$. Independently label each edge of $\bar{E}$ **in** with probability p, and **out** with probability $(1-p)$. Set $E\leftarrow E\cup\bar{E}$ and $K_{np}\leftarrow(V, E_{in})$, where E_{in} denotes the set of edges of E labeled **in**, and stop. (We terminate with the random graph K_{np} and a k-coloring of the vertices of K_{np}.)

Theorem 4. *For* $0<p<1$, $0<\varepsilon<1/2$, *and any sufficiently large* n, Algorithm A *determines a k-coloring of the random graph* K_{np} *where*

$$\text{Prob}\left\{k \leqq (1+\varepsilon)\frac{n}{\log n}\right\} > 1-1/n^{\varepsilon/4},$$

where the log *is to the base* $b=1/1-p$.

Proof. Although Algorithm A chooses the order in which edges of K_{np} are exposed based on the results of previously exposed edges, each edge exposed is labeled **in**, and thus is contained in the final graph K_{np}, independently with probability p. Thus the algorithm generates a random graph K_{np} as per the edge probability model.

Steps 2—7 of Algorithm A determine a partition of V into sets of size $m=\lfloor(1-\varepsilon/2)\log n\rfloor$ and sets of size 1. For sufficiently large n, the partition contains, in total, at most $(1+2\varepsilon/3)n/\log n$ sets. A set P of the partition when colored with m colors by Algorithm A is termed *bad*; otherwise, then colored with a single color, is termed *good*. To verify the theorem it is sufficient to show that the partition contains only $o(n/\log^2 n)$ bad sets with probability at least $1-1/n^{\varepsilon/4}$.

A set P of the partition will be bad for either of two reasons. Initially, P will be bad if there is no m-membered independent set in the corresponding random graph $K_{|S|p}$. For sufficiently large n, $m<z(\lceil n^{1/2-\varepsilon/5}\rceil, p)-4$, so by Theorem 3 this occurs with probability at most $1/n^{1/3}$. Secondly, P will be bad if it contains an edge of E. Now when P is chosen, $|E|\leqq(n^{1/2-\varepsilon/5})^2 n=n^{2-2\varepsilon/5}$, and $|U|\geqq n/\log^2 n$. Note that each edge of E is in at most $\binom{|U|-2}{m-2}$ m-tuples of vertices of U, so the probability that any randomly chosen m-tuple of U contains an edge of E is, for sufficiently large n, at most

$$\frac{\binom{|U|-2}{m-2}|E|}{\binom{|U|}{m}} \leqq \frac{m^2|E|}{|U|^2} \leqq \frac{\log^6 n}{n^{2\varepsilon/5}} \leqq \frac{1}{2n^{\varepsilon/3}}.$$

The procedures of Steps 2 and 3 of Algorithm A assure that $P\subset U$ is chosen uniformly over all m-membered sets of U, so P contains an edge of U with probability at most $1/2n^{\varepsilon/3}$.

Hence for sufficiently large n, P is bad with probability at most $1/n^{1/3}+1/2n^{\varepsilon/3}<1/n^{\varepsilon/3}$. Then, by Markov's Theorem, the probability that more than

$$n^{\varepsilon/4}\left[\left((1+2\varepsilon/3)\frac{n}{\log n}\right)\frac{1}{n^{\varepsilon/3}}\right] = o\left(\frac{n}{\log^2 n}\right)$$

sets of the partition are bad is at most $1/n^{\varepsilon/4}$, proving the theorem. ∎

Our second graph coloring algorithm involves a more sophisticated application of the expose-and-merge exploration paradigm. The result yields an improvement in the upper bound on the chromatic number of a random graph reducing the range of uncertainty for the concentrated distribution of the random variable $\chi(K_{np})$ to a size 1/3 that of Theorem 2.

Algorithm B: Given $0<p<1$, $0<\varepsilon<1/9$, and any sufficiently large n, this algorithm generates an instance of the random graph K_{np} and a k-coloring of the vertices of K_{np} where

$$\mathrm{Prob}\left\{k \leqq \left(\frac{2}{3}+\varepsilon\right)\frac{n}{\log n}\right\} > 1-\frac{1}{n^{\varepsilon/4}}.$$

1. [Initialize] $V\leftarrow\{1, 2, \ldots, n\}$, $U\leftarrow V$, $E\leftarrow\emptyset$, $k\leftarrow 0$, $m\leftarrow\lfloor(1/2)(1-\varepsilon)\log n\rfloor$. (All n vertices of V are assigned to the uncolored vertex set U, the set of exposed edges E is initialized to empty, and the number of colors employed k is set to zero.

The parameter m is fixed at $\lfloor(1/2)(1-\varepsilon)\log n\rfloor$, and distinct color values will subsequently be assigned to vertex sets of size either $3m$ or unity.)

2. [Create partition S, $\bar{S}$ of U] Select an $\lceil n^{1/4-\varepsilon/5}\rceil$-vertex set $S\subset U$ at random yielding the partition S, $\bar{S}=U-S$. (The cycle of Steps 2—8 will assign colors to m vertices of S and $2m$ vertices of $\bar{S}$.)

3. [Expose $K_{|S|p}$ and select m vertices to color from S] Expose a random graph $K_{|S|p}$ on vertex set S and select the m-vertex set P at random from all m-membered independent sets of $K_{|S|p}$ if any exist, otherwise select P at random from all m-membered sets of S.

4. [Expose $K_{|P||\bar{S}|p}$ and determine admissible vertices of $\bar{S}$] Expose a random bipartite graph $K_{|P||\bar{S}|p}$ on sets P and $\bar{S}$, and let $A\subset\bar{S}$ be the set of all vertices of $\bar{S}$ non-adjacent to all vertices of P in $K_{|P||\bar{S}|p}$. (Note that if P is an independent set of $K_{|S|p}$, then $P\cup A$ is an independent set in the graph $K_{|S|p}\cup K_{|P||\bar{S}|p}$.)

5. [Expose random subgraph in $\bar{S}$ and select $2m$ vertices to color from $\bar{S}$] Select $R\subset\bar{S}$ at random over all $\lceil n^{1/2-\varepsilon/5}\rceil$-vertex sets of A if $|A|\geqq\lceil n^{1/2-\varepsilon/5}\rceil$, otherwise at random over all $\lceil n^{1/2-\varepsilon/5}\rceil$-vertex sets of $\bar{S}$. Expose a random graph $K_{|R|p}$ on vertex set R. Select the $2m$-vertex set $\bar{P}$ at random from all $2m$-membered independent sets of $K_{|R|p}$ if any exist, otherwise select $\bar{P}$ at random from all $2m$-membered sets of R. (The procedure of Steps 2—5 assures that $P\cup\bar{P}$ is implicitly chosen uniformly from all $3m$-membered sets of U.)

6. [Color selected vertices] If $P\cup\bar{P}$ is an independent set of the graph $K_{|S|p}\cup K_{|P||\bar{S}|p}\cup K_{|R|p}$ and if no edge of E joins two vertices of $P\cup\bar{P}$, assign all vertices of $P\cup\bar{P}$ color value $k+1$ and set $k\leftarrow k+1$. Otherwise, assign the $3m$ vertices of $P\cup\bar{P}$ distinct color values $k+1, k+2, \ldots, k+3m$ and set $k\leftarrow k+3m$. (This assigns a coloring to $P\cup\bar{P}$ consistent with $K_{|S|p}$, $K_{|P||\bar{S}|p}$, $K_{|R|p}$ and the previously exposed edges E.)

7. [Merge random subgraphs] Any edge $e\in\bar{E}$ joining (i) two vertices of S, or (ii) a vertex of P and a vertex of $\bar{S}$, or (iii) two vertices of R, is added to the exposed set E and labeled in if it is an edge of $K_{|S|p}\cup K_{|P||\bar{S}|p}\cup K_{|R|p}$, and is labeled **out** otherwise. Set $U\leftarrow U-P-\bar{P}$. (At this point it remains true that any two vertices of $V-U$ assigned the same color value must be joined by an exposed edge of E labeled **out**.)

8. [Uncolored vertex set large?] If $|U|\geqq n/\log^2 n$, return to Step 2. (The cycle of Steps 2—8 will be executed exactly $1+\lceil n(1-1/\log^2 n)/3m\rceil$ times.)

9. [Complete random graph and vertex coloring] Assign vertices of U the colors $k+1, k+2, \ldots, k+|U|$, and set $k\leftarrow k+|U|$. Independently label each remaining edge of $\bar{E}$ **in** with probability p and **out** with probability $(1-p)$, and set $E\leftarrow E\cup\bar{E}$. Set $K_{np}\leftarrow(V, E_{in})$, where E_{in} denotes all exposed edges of E labeled **in**, and stop. (Termination thus yields the random graph K_{np} and a k-coloring of the vertices of K_{np}.)

IV. Chromatic Number of a Random Graph

Theorem 5. *For* $0<p<1$, $0<\varepsilon<1/9$, *and any sufficiently large* n, Algorithm B *determines a k-coloring of the random graph* K_{np} *where*

$$\mathrm{Prob}\left\{k \leqq \left(\frac{2}{3}+\varepsilon\right)\frac{n}{\log n}\right\} > 1-1/n^{\varepsilon/4},$$

where the log *is to the base* $b=1/1-p$.

Proof. Although Algorithm B chooses the order in which edges of K_{np} are exposed based on the results of previously exposed edges, each edge exposed is labeled **in**, and thus is contained in the final graph K_{np}, independently with probability p. Thus the algorithm generates a random graph K_{np} as per the edge probability model.

Steps 2—9 of Algorithm B partition V into at most $((2/3)+(3\varepsilon/4))n/\log n$ sets for sufficiently large n, each set of size $3m$ or 1, where $m=\lfloor(1/2)(1-\varepsilon)\log n\rfloor$. A set $P\cup\bar{P}$ of the partition colored with $3m$ colors by Algorithm A is termed *bad*; otherwise, colored with a single color, is termed *good*. We shall show that the partition contains only $o(n/\log^2 n)$ bad sets with probability at least $1-1/n^{\varepsilon/4}$.

A set $P\cup\bar{P}$ of the partition will be bad if either there is no m-membered independent set in the corresponding $K_{|S|p}$ or no $2m$-membered independent set in the corresponding $K_{|R|p}$. By Theorem 3 these events occur with probabilities at most $1/n^{1/5}$ and $1/n^{1/3}$, respectively. Also $P\cup\bar{P}$ will be bad if $|A|<n^{1/2-\varepsilon/5}$. Consider that each of the at least $n/\log^2 n-n^{1/4}$ vertices of $\bar{S}$ will be in A independently with probability $(1-p)^m\geqq 1/n^{1/2-\varepsilon/2}$. For sufficiently large n, the expected size of $|A|$ is then greater than $n^{1/2}$. Since A is formed from independent Bernoulli trials, it follows [4] that prob $\{|A|\geqq n^{1/2-\varepsilon/5}\}>1-1/n$. Thus $P\cup\bar{P}$ will be bad for the reason $|A|<n^{1/2-\varepsilon/5}$ with probability at most $1/n$. Finally, $P\cup\bar{P}$ will be bad if it contains an edge of E, i.e. a previously exposed edge. Since vertices of P are deleted from U, the exposed edges remaining in U created during each cycle come only from the $\binom{|S|}{2}+\binom{|R|}{2}$ vertex pairs of the graphs $K_{|S|p}$ and $K_{|R|p}$. For sufficiently large n, when each set $P\cup\bar{P}$ is chosen, $|E_u|\leqq(n^{1/2-\varepsilon/5})^2 n=n^{2-2\varepsilon/5}$, and $|U|\geqq n/\log^2 n$, where E_u denotes the edges of E in U before merger (step 7).

Now each edge of E_u is in at most $\binom{|U|-2}{3m-2}$ $3m$-tuples of vertices of U, so the probability that any randomly chosen $3m$-tuple of U contains an edge of E_u is, for sufficiently large n, at most

$$\frac{\binom{|U|-2}{3m-2}|E_u|}{\binom{|U|}{3m}} \leqq \frac{9m^2|E_u|}{|U|^2} \leqq \frac{3\log^6 n}{n^{2\varepsilon/5}} \leqq \frac{1}{2n^{\varepsilon/3}}.$$

The procedures of Steps 2—8 of Algorithm B assure that $(P\cup\bar{P})\subset U$ is chosen uniformly over all $3m$-membered sets of U, so $P\cup\bar{P}$ contains an edge of E_u with probability at most $1/2n^{\varepsilon/3}$.

Hence, for sufficiently large n, $P\cup\bar{P}$ is bad with probability at most $1/n^{1/5}+1/n^{1/3}+1/n+1/2n^{\varepsilon/3}<1/n^{\varepsilon/3}$. Then, by Markov's Theorem, the probability that more than

$$n^{\varepsilon/4}\left[\left(\frac{2}{3}+\frac{3\varepsilon}{4}\right)\left(\frac{n}{\log n}\right)\frac{1}{n^{\varepsilon/3}}\right] = o\left(\frac{n}{\log^2 n}\right)$$

sets of the partition are bad is at most $1/n^{\varepsilon/4}$, proving the theorem. ∎

V. Average Case Complexity of Graph Coloring

Algorithm B can clearly be implemented in time $O(n^{(1/2)\log n})$, the dominant time coming from exhaustive inspection of all $2m$-membered sets of R in Step 5. So we obtain the following.

Corollary 5.1. *For* $\varepsilon>0$, *almost every graph* $G=(V, E)$ *can be colored in at most* $(4/3+\varepsilon)\chi(G)$ *colors in time complexity* $O(|V|^{1/2\log_2|V|})$.

Although the complexity indicated in Corollary 5.1 is subexponential, it is not polynomial. This leaves open the question of whether we can color almost every graph G in at most $c\chi(G)$ colors for any $c\leqq 2$ by a polynomially bounded algorithm.

The methodology illustrated by Algorithms A and B guided the design of Algorithm GE1 in [6]. Algorithm GE1 was shown to be practically efficient for approximate coloring of random graphs on up to 1000 vertices and (corresponding to $p=1/2$) some quarter million edges, and GE1 was observed to require about 20% fewer colors than the greedy algorithm on the 1000-vertex random graphs studied in [6].

As previously noted [5], the greedy algorithm does color almost every graph in $(2+\varepsilon)\chi(G)$ colors for any $\varepsilon>0$, and can be implemented in time $O(|V|+|E|)$. The greedy algorithm must, however, utilize $(1-\varepsilon)n/\log n$ colors in coloring $K_{\widehat{np}}$ with probability going to one, and so from Theorem 5 can not be asymptotically optimal on almost every graph.

Corollary 5.2. *For coloring almost every graph* G, *the greedy coloring algorithm requires at least* $((3/2)-\varepsilon)\chi(G)$ *and at most* $(2+\varepsilon)\chi(G)$ *colors for any* $\varepsilon>0$.

References

[1] B. Bollobás and P. Erdős, Cliques in random graphs, *Math. Proc. Camb. Phil. Soc.* **80**, (1976), 419—427.
[2] B. Bollobás, *Random Graphs*, Academic Press, London, 1985.
[3] P. Erdős and T. Spencer, *Probabilistic Methods in Combinatorics*, Academic Press, New York, 1974.
[4] W. Feller, *An Introduction to Probability Theory and its Applications*, 3rd Ed., Wiley, New York, 1968.
[5] G. R. Grimmett and C. J. H. McDiarmid, On coloring random graphs, *Math. Proc. Camb. Phil. Soc.* **77**, (1975), 313—324.
[6] A. Johri and D. W. Matula, Probabilistic bounds and heuristic algorithms for coloring large random graphs, *Tech. Rep.* 82-CSE-6, Southern Meth. Univ., Dallas, (1982).
[7] A. D. Korshunov, The chromatic number of n-vertex graphs, *Metody Diskret. Analiz. No.* **35** (1980), 14—44 (in Russian).
[8] D. W. Matula, The employee party problem, *Not. A. M. S.* **19**, (1972), A—382.

[9] D. W. MATULA, The largest clique size in a random graph, *Tech. Rep.* CS7608, Southern Meth. Univ., Dallas, (1976).
[10] D. W. MATULA, G. MARBLE and J. D. ISAACSON, Graph Coloring Algorithms, *in Graph Theory and Computing,* Read, R. C., ed., Academic Press, New York, 1972, 109—122.
[11] C. J. H. MCDIARMID, Colouring random graphs, *Ann. Op. Res.* **1**, (1984), 183—200.
[12] E. M. PALMER, *Graphical Evolution — An Introduction to the Theory of Random Graphs,* Wiely Interscience, New York, 1985.
[13] E. SHAMIR and J. SPENCER, Sharp Concentration of the Chromatic Number on Random Graphs $G_{n,p}$, *Combinatorica* 7 (1987), 121—129.

D. W. Matula

Dept. of Computer Science
and Engineering
Southern Methodist University
Dallas, Texas 75275