Coloring with Defects

by

C. E. Jesurum

Submitted to the Department of Applied Mathematics
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1995

Author ....
Department of Applied Mathematics
May 12, 1995

Certified by ....
Daniel Kleitman
Associate Professor
Thesis Supervisor

Accepted by ....
David Vogan
Chairman, Departmental Committee on Graduate Students

Coloring with Defects

by

C. E. Jesurum

Submitted to the Department of Applied Mathematics
on May 12, 1995, in partial fulfillment of the
requirements for the degree of
MASTER OF SCIENCE

Abstract

This paper is concerned with algorithms and complexity results for defective coloring, where a defective (k, d)-coloring is a k coloring of the vertices of a graph such that each vertex is adjacent to at most d-self-colored neighbors. First, $(2, d)$ coloring is shown NP-complete for $d \geq 1$, even for planar graphs, and $(3, 1)$ coloring is also shown NP-complete for planar graphs (while there exists a quadratic algorithm to (3,2)-color any planar graph). A reduction from ordinary vertex coloring then shows (χ, d) coloring NP-complete for any $\chi \geq 3, d \geq 0$, as well as hardness of approximation results.

Second, a generalization of $\Delta + 1$ coloring to defects is explored for graphs of maximum degree Δ. Based on a theorem of Lovasz, we obtain an $O(E)$ algorithm to $(k, \lfloor \frac{\Delta}{k} \rfloor)$ color any graph; this yields an $O(E)$ algorithm to (2,1)-color 3-regular graphs, and (3,2)-color 6-regular graphs.

The generalization of $\Delta + 1$ coloring is used in turn to generalize the polynomial-time approximate 3- and k-coloring algorithms of Widgerson and Karger-Motwani-Sudan to allow defects. For approximate 3-coloring, we obtain a linear time algorithm to $(\lceil (\frac{8n}{d})^{.5} \rceil, d)$ color, and a polynomial time algorithm to $(O((\frac{n}{d})^{.387}), d)$ color any 3-colorable graph.

Thesis Supervisor: Daniel Kleitman

Title: Associate Professor

Acknowledgments

This thesis represents joint work with Lenore Cowen. Without her guidance and constant reassurances it would not have been possible. We wish to thank DIMACS: the author did this work while visiting DIMACS in the Summer of 1994. Thanks to Martin Farach, Wayne Goddard, Steve Mahaney, and Mike Saks for helpful discussions. Thanks to Peter Shor for helpful comments.

0.1 Introduction

A proper (vertex) coloring of a graph is an assignment of colors to its vertices such that no two adjacent vertices receive the same color. Determining the chromatic number of G, the minimum number of colors needed to properly color G, is NP-hard. It remains NP-hard even to determine if a planar graph is 3-colorable [15]. Even the relaxation of the problem to approximate coloring is hard, in the sense that [22] using results of [2, 12] showed that there exists an $\epsilon > 0$ such that no polynomial time algorithm can n^{ϵ}-approximate the chromatic number unless $P = NP$. (in [3] the value of ϵ is improved under different hardness assumptions). In the special case of 3-colorable graphs, [19] showed that it is not possible to 4-color a 3-colorable graph unless $P = NP$.

This paper is concerned with relaxing the coloring problem in an additional way, namely we relax the requirement that each color class be an independent set. Cowen, Cowen and Woodall [9] first introduced the notion of coloring with defects as follows.

Definition 0.1.1 *A (k, d) coloring of a graph G, is a coloring of the vertices of G with k colors, such that each vertex is adjacent to at most d vertices of the same color.*

This has been an active area of current research in graph theory [27, 14]. Notice that $(k, 0)$ coloring corresponds to proper coloring. Notice also, that the defect requirement in the above definition is a *local* requirement, for example, in a legal $(3, 2)$ coloring, monochromatic paths and cycles are permitted. (Other definitions of defective coloring are of course possible, see Section 0.4 for discussion.)

For the scheduling problem where nodes represent jobs (say users on a computer system), and edges represent conflicts (needing to access one or more of the same files), allowing a defect means tolerating some threshold of conflict: each user may find the max slowdown incurred for retrieval of data with 2 conflicting other users on the system acceptable, and with more than 2 unacceptable.

Previous work. Cowen, Cowen and Woodall [9] give a complete characterization of all k and d such that every planar graph is (k, d)-colorable. Namely, there does not exist a d such that every planar graph is $(1, d)$- or $(2, d)$-colorable; there exist planar graphs which are not $(3, 1)$-colorable, but every planar graph is $(3, 2)$-colorable. Together with the $(4, 0)$ coloring implied by the 4-color Theorem, this solves defective chromatic number for the plane. We remark that the (3,2) coloring proof is constructive, and immediately implies an n^2 algorithm for (3,2) coloring planar graphs (just as the proof of the 5-color theorem is constructive, and immediately implies an n^2 algorithm— improved by [8, 13, 23, 26] to a linear-time algorithm.)

Recently, [10] have extended this to toroidal graphs, and show that every toroidal graph is $(3, 2)$ and $(5, 1)$-colorable. As above, these proofs are constructive and give quadratic-time algorithms. It is still an open question whether every graph embeddable in the torus is $(4, 1)$-colorable.

Finally, Archdeacon [1], showed that for every genus g, there always exists a $d = d(g)$, such that every graph embeddable on a surface of genus g is $(3, d)$-colorable. (The value of d has since been improved by [10].)

Our results. In this paper we prove hardness results and give approximation algorithms for defective coloring. We show, perhaps surprisingly, that determining if a graph is $(2, d)$ colorable is NP-complete for $d \geq 1$, even for planar graphs. We show determining if a planar graph is $(3, 1)$ colorable is also NP-Complete, and in general graphs we show that (k, d) coloring is NP-Complete for all $k \geq 3$, and all $d \geq 0$. A simple reduction from proper coloring and the result of [22] shows that for any constant defect d, there exists an $\epsilon > 0$ such that χ (the chromatic number) cannot be approximated within a factor of n^{ϵ} unless $P = NP$.

These impossibility results for general graphs do not, of course, rule out good algorithms for defective coloring of bounded-degree graphs. In Section 0.2 we present a generalization of $\Delta + 1$ coloring to defects, that follows from results of Lovasz and others [21, 7, 20, 4, 27]. As a corollary, we show that any cubic graph can be $(2, 1)$-colored in linear time, as well as obtaining other degree-based results.

This generalization of $\Delta + 1$ coloring to defect then allows us to give polynomial-time approximation algorithms for defective coloring, in the spirit of Widgerson [25], and others who improved his bounds [5, 17, 18]. We show how to generalize both Widgerson's original algorithm, and the recent algorithms of Karger, Motwani and Sudan [18] to defects, and achieve a tradeoff between the defect and number of colors used.

The paper concludes with some open problems.

0.2 Hardness results

We show in this section that determining whether or not a graph is $(2, d)$ colorable is NP-complete. even for planar graphs (extending a result of [11] that first showed $(2, 1)$ coloring is NP-complete in general graphs). However, we show that the $(2, 1)$-colorable planar graphs are easy to 4-color in quadratic time. We show determining if a planar graph is $(3, 1)$ colorable is also NP-Complete, and in general graphs we show that (k, d) coloring is NP-Complete for all $k \geq 3$, and all $d \geq 0$. A simple reduction from proper coloring and the result of [22] shows that for any constant defect k, there exists an $\epsilon > 0$ such that χ cannot be approximated within a factor of n^ϵ unless $P = NP$.

0.2.1 Defective coloring in the plane

It is easy to $(2, 0)$-color any (planar) graph in linear time. Determining whether a planar graph is 3-colorable is NP-complete. In theory, the 4-color theorem gives a quadratic time algorithm for 4-coloring planar graphs (although the constants are worse than terrible). Since, as was remarked in the introduction, [9] give a quadratic-time algorithm to $(3, 2)$-color any planar graph, together with the results of this section, this characterizes the complexity of defective coloring in the plane.

Theorem 0.2.1 *To determine whether or not a planar graph is* $(2, d)$*-colorable is* NP*-complete.*

Proof: We first show $(2, d)$-colorability NP-complete for general graphs by reduction from 3-SAT, and then use an idea similar to [24] to planarize the structure. The fact that $(2, d)$-colorability is in NP should be obvious. We will show for any 3-CNF ϕ, there exists a graph G_ϕ constructible in polynomial time such that ϕ is satisfiable if and only if G_ϕ is $(2, d)$-colorable.

We first introduce the graph A in Figure 0-1. This graph has the useful property that in order to be validly $(2, d)$-colored, it must color a and b the same.

We next use A to construct a subgraph which serves as a logical OR (B of Figure 0-1). In particular this graph specifies two "input" nodes and one "output" node such that the output node can be colored with the color of either of the inputs. The node labeled v is the output node and does indeed have to be colored the same as one of the inputs (p and q in the figure). Notice u is forced to have $d-1$ defects, so that if p and q are the same color, then u must be the opposite color. And because w is forced to have d defects, v must have the opposite color from that of w (and therefore, u), that is, the color of p and q. If p and q have opposite colors, then we may color u arbitrarily, which is what we desire for OR. A clause is simply constructed by identifying one OR output to another OR input.

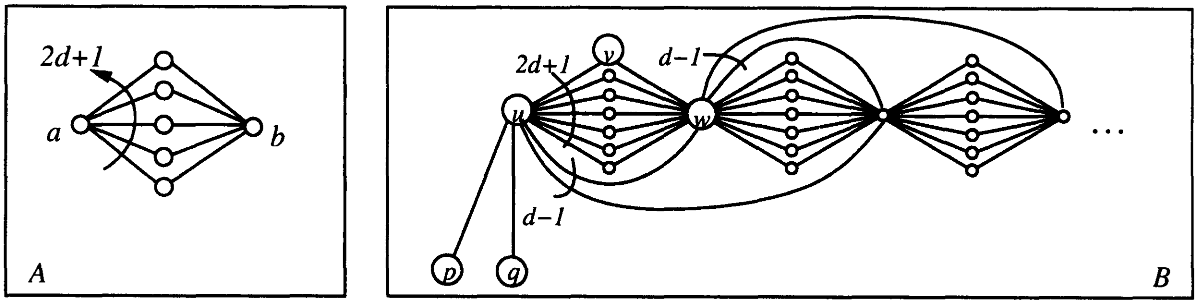

Figure 0-1: A is the forcing graph. Since there are $2d+1$ nodes in the middle, at least $d+1$ are colored, say, black. If this is the case, both a and b must be colored white. B has the property that node v must be colored the color of node p or node q if this is to be validly $(2,d)$-colored.

To each variable we associate a *variable-chain* depicted in Figure 0-2. In any valid $(2,d)$-coloring, all nodes labeled p have opposite color from those labeled $\neg p$. To avoid creating any unnecessary defects, the boldface $\mathbf{p}$ in the graph C of Figure 0-2 is not used in the clauses, but instead the italicized ones (p and $\neg p$) are used. The variable-chain is a chain of the graph C with enough copies so that there is a variable-node for each clause containing that literal in ϕ.

Finally, we force the output node of each clause-subgraph to have the same color as follows: A chain composed of multiple copies of the graph A in Figure 0-1 is added to the structure so that the output nodes take the place of the a and b nodes. The graph that results is G_ϕ.

Suppose we have a valid $(2,d)$-coloring of G_ϕ. Without loss of generality, assume that the output of each clause is colored white. By construction, at least one of the inputs to each clause is colored white also. If we associate white with TRUE

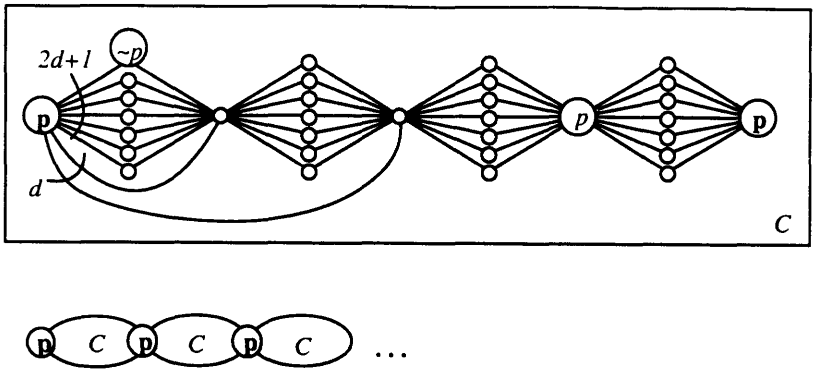

Figure 0-2: The variable-chain.

and black as FALSE, this coloring yields a satisfying assignment for ϕ. Also, if ϕ is satisfiable, the truth assignment yields a $(2, d)$-coloring for G_ϕ in the natural way (color the nodes associated with true variables white and the others black and the rest of the graph is forced accordingly). It is easy to see that this must yield a valid $(2, d)$-coloring, so we are done.

The graph constructed for the reduction above is not planar since the variable-chains may have to cross one another (and these are the only crossings). However, it can be made planar as follows: Whenever two edges cross, we can uncross them as in Figure 0-3. The graph D is used to force every node on the 8-cycle to have $d - 1$ defects. In this way we are assured that x and x' have the same color and y and y' have the same color. We add the graph A of Figure 0-1 so that these labeled nodes are not forced to have any defects as a result of making the graph planar.

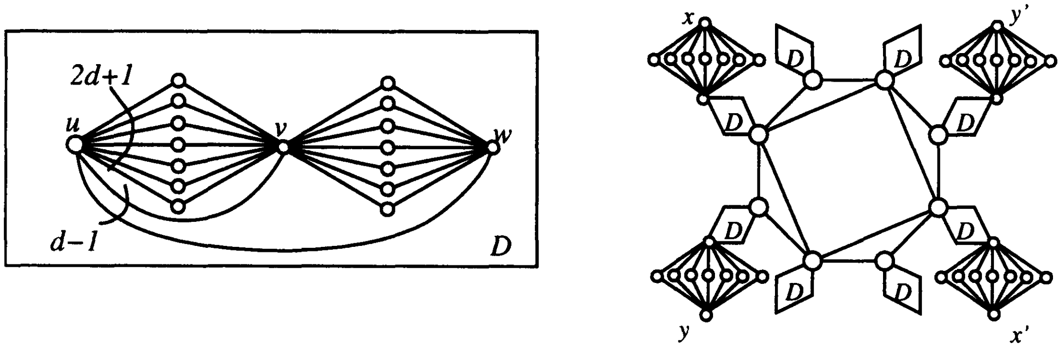

Figure 0-3: A graph used to uncross edges in a $(2, d)$-colored graph.

The number of times we might need to add this planarizing subgraph is at most the number of pairs of edges in G_ϕ, so the resulting graph has $O(m^4)$ nodes — and is planar. □

We remark that even though $(2,1)$-coloring planar graphs is NP-complete, nonetheless these graphs belong to a class of planar graphs that are easy to 4-color in practice.

Theorem 0.2.2 *Any* $(2,1)$*-colorable planar graph can be 4-colored in* n^2 *time.*

Proof. It follows easily from Euler's formula that the average degree of any planar bipartite graph will be less than 4. Since a $(2,1)$-colorable graph is the union of a bipartite graph plus a matching, the average degree of any $(2,1)$-colorable planar graph will be less than 5, and hence will always have a vertex of degree 4 or less. But then G can be colored by induction, using a Kempe chain argument to re-color in linear time, in the case when v's four neighbors have different colors. □

Theorem 0.2.3 $(3,1)$ *coloring planar graphs is NP-complete.*

Proof. The reduction is from planar 3-coloring. For any graph G in the plane, form the graph G' by joining to each vertex of G the 6 node subgraph, H (depicted in Figure 0-4). Since H is outerplanar, all its vertices can be joined to a single vertex of G and the resulting graph will still be planar. Furthermore, it is simple to check that H is not (2,1)-colorable, so in any legal (3,1)-coloring, all 3 colors must appear among the vertices of each copy of H (but H can be (3,1)-colored so that a specified color appears only once thereby giving each vertex in G exactly one new defect). Thus G' will be (3,1) colorable if and only if G was 3-colorable. □

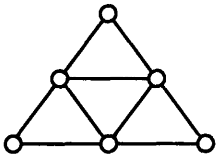

Figure 0-4: The outerplanar graph, H, which is not (2,1)-colorable.

0.2.2 Hardness of defective coloring for general graphs

Theorem 0.2.4 *(k, d) colorability is NP-complete for $k \geq 3$ and $d \geq 0$.*

Proof. We reduce k-colorability to (k, d)-colorability as follows. First consider the complete k-partite graph K, in which each part contains $\max(n, kd + 1)$ nodes. Because each part contains at least $kd + 1$ nodes, the only valid (k, d) coloring of K assigns a different color to each part. Now form the graph G' from the graph G by connecting each node of G to d nodes in each part of K (while making sure that no node of K is connected to more than d nodes of G). The resulting graph G' is indeed (k, d)-colorable if and only if G is k-colorable and the number of nodes in G' is $\max(n + k(kd + 1), n + kn)$ or linear in n for constant k and d. □

Theorem 0.2.5 *There exists an $\epsilon > 0$ such that no polynomial time algorithm can n^{ϵ}-approximate the d-defective chromatic number, for constant defect d, unless $P = NP$.*

Proof. Follows immediately from the result of [2], since it is easy to transform a (χ, d)-coloring of a graph G into an $(\chi(d + 1), 0)$ coloring of G, by simply $d + 1$ coloring each color class. This is an $O(\chi)$ coloring of G when d is constant. □

0.3 A generalization of $\Delta + 1$ coloring

This section begins with a generalization of $\Delta + 1$ coloring to defects that has been discovered and re-discovered many times in the literature [21, 7, 20, 4, 27]. The first reference we know of, is due to Lovasz.

Theorem 0.3.1 *[Lovasz] It is always possible to 2-color the vertices of a graph red and blue, so that every blue vertex has at least as many red vertex-neighbors as blue-vertex neighbors, and every red vertex has at least as many blue-vertex neighbors as red vertex-neighbors.*

Proof. Of all colorings of the vertices of G, consider one with the maximum number of bichromatic edges (that is, edges with one red and one blue endpoint). We claim that this is the desired coloring, because if not, there exists some node v with more than half its neighbors self-colored. Switching v's color then increases the number of bichromatic edges, contradicting the maximality of the original coloring. □

Corollary 0.3.2 *Any graph of maximum degree Δ can be $(2, \lfloor \Delta/2 \rfloor)$ colored in $O(E)$ time.*

Proof. Begin with the all blue graph, and greedily construct the Lovasz coloring, by arbitrarily picking any vertex with more self-colored than opposite-colored neighbors, and flipping its color. The procedure terminates in at most E steps, since every time a vertex v is flipped in G, the number of bichromatic edges in G increases by at least 1. □

Corollary 0.3.3 *Any 3-regular graph can be $(2, 1)$-colored in $O(E)$ time.* □

As Lovasz and others also noted, it is straightforward to generalize this to k colors, noting that in any k-coloring of the vertices of G, there is always one color class with at most $\lfloor \frac{\Delta}{k} \rfloor$ members in the neighbor set of v. Thus any vertex adjacent to more than $\lfloor \frac{\Delta}{k} \rfloor$ self-colored neighbors can be flipped to this color, increasing the number of bichromatic edges.

Theorem 0.3.4 *Any graph of maximum degree Δ can be $(k, \lfloor\frac{\Delta}{k}\rfloor)$ colored in $O(E)$ time.*

As before, special case values of Δ and k may be of particular interest: for example, any 6-regular graph can be $(3,2)$ colored.

We remark that Theorem 0.2.4 is equally valid for multigraphs: for the scheduling application mentioned in the introduction, this allows us to weight the conflict for jobs u and v running concurrently by placing multiple edges between them. Of course this will in general increase the value of Δ.

We will use Theorem 0.2.4 to show, in the next section, in analogous way to how Widgerson [25], and others [5, 18] have used $\Delta+1$ coloring, to give polynomial-time approximation algorithms for coloring 3-colorable and k-colorable graphs.

0.4 Approximate Defective Coloring

Widgerson gives the following algorithm to approximately color 3-colorable graphs. Pick a threshold δ. Take the node of highest degree and 2 color its neighborhood with two new colors. Remove its neighborhood. Continue until all nodes have degree at most δ. Then we can δ color the remaining graph by Brooks' Theorem [6]. Each round we eliminate at least δ nodes using 2 colors, so the total number of colors used is $2n/\delta + \delta$. We now show how to modify this allowing for some defect, d.

Theorem 0.4.1 *There exists a linear time algorithm to* $(\lceil (\frac{8n}{d})^{.5} \rceil, d)$*-color any 3-colorable graph.*

Proof: We follow the algorithm of Widgerson until the maximum degree is δ. We then $(\frac{\delta}{d}, d)$ color in linear time by Theorem 0.2.4. Therefore the total number of colors is $2n/\delta + \delta/d$ which is optimized by choosing $\delta = \sqrt{2n/d}$. □

This result is the first tradeoff we obtain between proper approximate coloring and approximate coloring with defect. For instance, if we let $d = n^{1/3}$ we obtain a $(2\sqrt{2}n^{1/3}, n^{1/3})$ coloring, a substantial reduction in colors from the algorithm of Widgerson. However, the bounds Widgerson achieved for approximate coloring have since been improved, most recently by [18].

We next show how to get a similar tradeoff for the new and better approximation algorithms of Karger, Motwani and Sudan [18]. We use the semidefinite program approach of [18] to obtain a vector 3-coloring. We then round to an integer defective coloring. Finally, we generalize to χ-colorable graphs.

0.4.1 Generalizing the KMS algorithm to defects

The approximation algorithms of [18] work as follows. First, the 3-coloring problem is relaxed to the *vector 3-coloring* problem, which is solved in polynomial time using semidefinite programming. The vector 3-coloring assigns unit vectors to the vertices so that two vertices adjacent in the graph have dot product $\leq \frac{-1}{2}$. Next, the vector 3-coloring is rounded to an ordinary coloring by partitioning the space R^n by random hyperplanes, or by an alternate method involving the projection of the vectors onto random centers. As expected from the impossibility results of [22], the rounding step involves an exponential blowup in the number of colors, but the bounds achieved by [18] are still much improved from [25].

The randomized methods in [18] work by transforming the vector coloring into a *semicoloring,* which is an assignment of colors to the n vertices, so that the set of vertices which are not properly colored is of size less than $n/4$. This bears a great deal of superficial resemblance to defective coloring already, because it allows (many) defective (i.e. monochromatic) edges. However, the [18] semicoloring is a *global* condition on defects that will, in general, not place any guarantee on the maximum local defect (note that the angle between the vectors corresponding to two non-adjacent neighbors of a vertex will often be forced to be very small, to satisfy nearby edge constraints. Thus the probability that 2 such edges are both cut by a random hyperplane, or both have endpoints captured by different centers, is far from independent.) Thus we must modify the definition of semicoloring to reflect defects.

Definition 0.4.2 *A* d-defect semicoloring *is an assignment of colors to the* n *vertices, such that the number of nodes that have more than* d *adjacent neighbors of the same color, is less than* $n/4$.

The algorithm which produces a semicoloring with S colors, is used recursively on the set of nodes with more than d adjacent neighbors of the same color, to color the graph with defect d, with less than a factor of 2 blowup in the number of colors. We prove the following theorem.

Theorem 0.4.3 *There exists a polynomial-time algorithm to* $(O((\frac{n}{d})^{.387}), d)$*-color a 3-colorable graph.*

Proof. Given a vector 3-coloring, we select $r + O(1)$ independent random hyperlanes (where a random hyperplane is one with a random normal on the unit sphere in R^n). The fact that the vectors assigned to each node form a vector 3-coloring gives us the probability that a random hyperplane cuts an edge is at least 2/3 [16]. Then, by Markov's inequality, with probability $1/2$, the number of uncut edges is at most $c(\frac{1}{3})^r n\delta$. The number of vertices which are adjacent to at least d uncut edges is at most $2c((\frac{1}{3})^r n\delta)/d$, which is $o(n)$ when $r = \log_3 O(\frac{\delta}{d})$ (and less than $n/4$ by appropriate choice of constants). Repeating the entire process t times ensures that we will find the required coloring with probability at least $1 - 1/2^t$. Thus any graph of maximum degree δ can be d-defect semicolored with $O(2^r) = O((\frac{\delta}{d})^{\log_3 2})$ colors, which is approximately $O((\frac{\delta}{d})^{.631})$. We now combine this with the Widgerson technique. Fix a threshold δ. While the maximum degree of the graph is greater than δ (or until half the vertices have been colored) pick a vertex of maximum degree, and 2-color its neighborhood (its neighborhood is 2-colorable, and 2-coloring is easy), with two new colors. Then we can finish with the above method when the degree falls below δ. This yields a coloring with $O(\frac{n}{\delta} + (\frac{\delta}{d})^{.631})$ colors, which is optimized by setting $\delta = n^{.631}d^{.387}$. □

0.4.2 Generalization to χ Colors

We now indicate how this generalizes to χ-colorable graphs for $\chi > 3$. The guarantee on the size of the angle separating the two endpoints of an edge in a vector χ-coloring is now only $\theta \geq \arccos(-1/\chi - 1)$. The lemma of [16] now gives that the probability that this edge is cut by a random hyperplane is θ/π. Thus the probability that an edge is not cut by r independent random hyperplanes is given by $(1 - \frac{\theta}{\pi})^r = \alpha^r$, and with appropriate choice of constants, just as in the proof of Theorem 0.3.3, since δ is always less than n we immediately obtain an $\tilde{O}$ $((\frac{n}{d})^{log_\alpha 2})$ d-defective semicoloring, which yields a ($\tilde{O}$ $((\frac{n}{d})^{log_\alpha 2}), d)$ coloring. However, we can do somewhat better by noticing that the neighborhood of a vertex in a χ-colorable graph is vector $\chi - 1$ colorable (see [18]), and we can set a degree threshold $\delta_{\chi(i)}$ for each $\chi(i) \leq \chi$, so that we recursively approximately $\chi - 1$ defective color the neighborhood of vertices of degree greater than $\delta_{\chi(i)}$, and then ($\tilde{O}$ $((\frac{\delta_{\chi(i)}}{d})^{log_\alpha 2}), d)$ color the remainder of the graph as above. We note also that it is similarly straightforward to generalize the alternate vector-projection rounding schemes of [18] to defects, using the fact that the probability that t random centers fail to cut an edge is approximately $t^{(-\chi/\chi-2)}$, to construct a d-defective semicoloring.

0.4.3 Discussion of tradeoffs

Finally, we close this section with a discussion of the tradeoffs we obtain. In each of these polynomial time algorithms which approximately 3-color with n^ϵ colors, allowing for a defect d in this manner, we are saving a multiplicative factor of about $1/d^\epsilon$. Current approximation algorithms for 3-coloring, still have ϵ bounded sufficiently far from 0 that allowing defect can still give an interesting tradeoff for even the best algorithms. However, the ideas presented in this section still give tradeoffs where the savings from defect decreases with the savings obtained by employing cleverer proper approximate coloring algorithms. We ask whether it is possible to get better tradeoffs, noting that the best tradeoff we might expect is a factor of $1/d$ savings. If the defective approximation algorithm achieves defect bounded by d with better than a factor of $1/d$ savings off the number of colors, re-coloring each color class with d colors by Brooks' theorem will improve the original approximation algorithm.

0.5 Open Problems

Some open problems and future directions seem obvious. We have already asked if the tradeoffs in the previous section can be tightened.

Another question is to look at alternate definitions of defective coloring. One possibility would be to allow some *total* number of monochromatic edges, rather than the stronger requirement of a maximum threshold of monochromatic edges at each vertex. This is akin to the concept of *semi-coloring* introduced by [18] (and the tradeoff between global defect and number of colors used in this case is fairly straightforward from their analysis.) This global threshold on defects does not make much sense for the scheduling application we mentioned in the introduction, but the scheduling problem suggests several alternate generalizations of defective coloring.

One generalization, that we already discussed briefly at the end of Section 0.2 would be to allow a different number of defects at different nodes: some jobs may be more tolerant of interference than others, or all conflicts could not be equally expensive. As mentioned in Section 0.2, this could be modeled by allowing multiple edges, or equivalently weights on the conflict edges.

Finally, if different colors correspond to different time periods in the schedule, it is possible that some jobs may not be able to schedule in all time- slots, rather each job may have a different subset of slots in which it is allowed to be scheduled. This is the defective version of the "list-coloring" problem, and would allow the modeling of more complicated constraints.

Bibliography

[1] D. Archdeacon. A note on defective coloring of graphs in surfaces. *Journal of Graph Theory*, 11:517–519, 1987.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and hardness of approximation problems. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 14–23, 1992.

[3] M. Bellare and M. Sudan. Improved non-approximability results. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, 1994.

[4] C. Bernardi. On a theorem about vertex colorings of graphs. *Discrete Mathematics*, 64:95–96, 1987.

[5] A. Blum. New approximation algorithms for graph coloring. *Journal of the ACM*, 41:470–516, 1994.

[6] B. Bollobas. *Graph Theory: an introductory course.* Springer–Verlag, New York, 1979.

[7] O. Borodin and A. Kostochka. On an upper bound of a graph's chromatic number depending on the graph's degree and density. *Journal of Combinatorial Theory, Series B*, 23:247–250, 1977.

[8] N. Chiba, T. Nishizeki, and N. Saito. A linear-time algorithm for 5-coloring a planar graph. *Journal of Algorithms*, 2:317–327, 1981.

[9] L. Cowen, R. Cowen, and D. Woodall. Defective colorings of graphs in surfaces: partitions into subgraphs of bounded valence. *Journal of Graph Theory*, 10:187–195, 1986.

[10] L. Cowen, W. Goddard, and E. Jesurum. Defecitve coloring revisited. Unpublished manuscript., 1994.

[11] R. Cowen. Some connections between set theory and computer science. In G. Gottlob, A. Leitsch, and D. Mundici, editors, *Proceedings of the third Kurt Godel colloquium on Computational Logic and proof theory*, Lecture Notes in Computer Science, pages 14–22. Springer-Verlag, 1993.

[12] U. Feige, S. Goldwasser, L. Lovasz, S. Safra, and M. Szegedy. Approximating clique is almost NP-complete. In *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science*, pages 2–12, 1991.

[13] G. Fredrickson. On linear-time alogrithms for five-coloring planar graphs. *Information Processing Letters*, 19:219–224, 1984.

[14] M. Frick. A survey of (m,k)-colorings. In J. Gimbel, J. W. Kennedy, and L. V. Quintas, editors, *Quo Vadis, Graph Theory?*, volume 55 of *Annals of Discrete Mathematics*, pages 45–58. Elsiver Science Publishers, 1993.

[15] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness.* Freeman, San Francisco, CA, 1981.

[16] M. Goemans and D. Williamson. .878-approximation algorithms for MAX CUT and MAX 2SAT. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, 1994.

[17] M. Halldorsson. A still better performance guarantee for approximate graph coloring. *Information Processing Letters*, 45:19–23, 1993.

[18] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, 1994.

[19] S. Khanna. N. Linial. and S. Safra. On the hardness of approximating the chromatic number. In *Proceedings of the 2nd Israeli Symposium on Theory and Computing Systems*, pages 250–260, 1992.

[20] J. Lawrence. Covering the vertex set of a graph with subgraphs of smaller degree. *Discrete Mathematics*, 21:271–273, 1978.

[21] L. Lovasz. On decompositions of graphs. *Studia Sci. Math. Hungar.*, 1:237–238, 1966.

[22] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing*, pages 286–293, 1993.

[23] D. Matula, Y. Shiloah, and R. Tarjan. Two linear-time algorithms for five-coloring a planar graph. Technical Report STAN-CS-80-830, Dept. of CS, Stanford U.

[24] L. Stockmeyer. Planar 3-colorability is NP-complete. *SIGACT news*, 5:19–25, 1973.

[25] A. Widgerson. Improving the performance for approximate graph coloring. *Journal of the ACM*, 30:729–735, 1983.

[26] M. Williams. A linear algorithm for coloring planar graphs with five colors. *Comput. J.*, 28:78–81, 1985.

[27] D. Woodall. Improper colourings of graphs. In R. Nelson and R. J. Wilson, editors, *Graph Colourings*. Longman Scientific and Technical, 1990.