Linear Programming in some Ramsey Problems

BRENDAN D. MCKAY

Computer Science Department
Australian National University
Canberra, ACT 2601, Australia

and

STANISŁAW P. RADZISZOWSKI *

Department of Computer Science
Rochester Institute of Technology
Rochester, New York 14623

Abstract. We derive new upper bounds for the classical two-color Ramsey numbers $R(4,5) \le 27$, $R(5,5) \le 52$ and $R(4,6) \le 43$; the previous best upper bounds known for these numbers were 28, 53 and 44, respectively. The new bounds are obtained by solving large integer linear programs and with the help of other computer algorithms.

1. Introduction

The two-color Ramsey number $R(k,l)$ is the smallest integer n such that for any graph F on n vertices, either F contains a copy of K_k or $\bar{F}$ contains a copy of K_l, where $\bar{F}$ denotes the complement of the graph F. A graph F is called (k,l)*–good* if F does not contain a K_k and $\bar{F}$ does not contain a K_l. Any (k,l)-good graph on n vertices will be called a (k,l,n)-good graph.

The best known lower bound of 25 for $R(4,5)$ was established in 1965 by Kalbfleisch [Ka1], who constructed a cyclic (4,5,24)-good graph. Until January 1991, the best upper bound of 28 was due since 1971 to Walker [Wa2], but this paper shows $R(4,5) \le 27$. In [MR2] we proved that $R(5,5) \le 53$. The best lower bound for this number, $R(5,5) \ge 43$, was obtained by Exoo [Ex1] in 1989, who constructed a (5,5)-good graph on 42 vertices using a simulated annealing technique. Recently Exoo found a new (4,6)-good graph on 34 vertices [Ex2], thus showing $R(4,6) \ge 35$; the previous best upper bound of 44 for this number was due to Walker [Wa2].

In order to find new extremal Ramsey graphs or to show that some family of them is empty by a full enumeration, one may have to search through a prohibitively large space of possibilities. We prune this search space in two stages. Firstly, various combinatorial lemmas relating the properties of the graphs we are searching for to other parameters can reduce the size of the search space. Secondly, large instances of integer linear programming problems are constructed, which when solved provide substantial information about such graphs (including proofs of nonexistence). Finally, a computer search is performed within the remaining part of the search space.

2. Attack on R(4,5)

In this paper we focus our attention on standard two-color Ramsey numbers $R(k,l)$, and especially on $R(4,5)$. The neighborhood $N(x)$ of a vertex x in an (k,l,n)-good graph $F=(V,E)$ induces an $(k-1,l,deg_F(x))$-good graph, and $V-N(x)-\{x\}$ induces an $(k,l-1,n-deg_F(x)-1)$-good graph; for a fixed F these two induced subgraphs will be denoted by G_x and H_x, respectively. The knowledge of

* Supported in part by a grant from the National Science Foundation CCR-8920692

$(k-1,l)$-good and $(k,l-1)$-good graphs is critical in the study of $R(k,l)$. For the case of $R(4,5)$ we need (3,5)-good and (4,4)-good graphs, and all of them are known [MR2,MZ,RK]. An exhaustive enumeration of all (4,5)-good graphs is out of a question, since there are too many such graphs.

Let $e(k,l,n)$ and $E(k,l,n)$ denote the minimum and maximum number of edges in any (k,l,n)-good graph, and let $t(k,l,n)$ be the minimum number of triangles in any such graph. As stated earlier, we know the values of $e()$, $E()$ and $t()$ for all n in the cases of $(k,l)=(4,4)$ and $(k,l)=(3,5)$. Also let $e(F)$ and $t(F)$ denote the number of edges and triangles, respectively, in any graph F. Walker [Wa1, see also MR2] proved the following Theorem 1; Theorems 2 and 3 below appear in [MR2] as lemmas 3 and 2, respectively.

Theorem 1: *If n_i is the number of vertices of degree i in any (k,l,n)-good graph F, then*

$$0 \leq \sum_{i=0}^{n-1} \Big[2E(k-1,l,i) + 2E(l-1,k,n-i-1) + 3i(n-i-1) - (n-1)(n-2) \Big] n_i . \qquad (1)$$

Theorem 2: *If n_i is the number of vertices of degree i in any (4,5,n)-good graph F on at least 24 vertices, then*

$$0 \leq \sum_{i=6}^{13} \Big[(n+9-3i)E(3,5,i) + 6i - 3t(4,4,n-i-1) \Big] n_i .$$

Theorem 3: *If a_i is the number of edges contained in exactly i triangles of some (4,5,n)-good graph F, then*

$$\sum_{x \in V(F)} t(H_x) = 4a_4 - 2a_2 - 2a_1 + \sum_{x \in V(F)} \Big(n/3 + 3 - deg_F(x) \Big) e(G_x).$$

Assume that F is a $(4,5,n)$-good graph with e edges and n_i vertices of degree i. In the following we describe an increasing sequence of systems of linear constraints LPi, for $0 \leq i \leq 4$, all of them satisfied for graph F. The variables range over nonnegative integers, and the number of edges e is optimized.

LP0. Variables n_i for $max(n-R(4,4),0) \leq i < min(n, R(3,5))$, constant n, constraints:

$$n = \sum_i n_i \qquad \text{and} \qquad 2e = \sum_i i n_i .$$

LP1. Extend LP0 by a constraint obtained from Theorem 1 by instantiating $k=4$ and $l=5$.

LP2. Extend LP1 by the inequality of Theorem 2 (LP2 is valid for (4,5)-good graphs on at least 24 vertices).

LP3. We extend LP1 as follows (the proofs of lemmas 2 and 3 in [MR2] easily imply that LP3 is also an extension of LP2). First introduce new variables g_{im} denoting the number of vertices $x \in V(F)$ such that $deg_F(x)=i$ and $e(G_x)=m$, and h_{im} denoting the number of vertices $x \in V(F)$ such that $deg_F(x)=n-i-1$ and $e(H_x)=m$. The ranges of indices i and m can be determined from n and the enumerations of (3,5)- and (4,4)-good graphs. The total number of variables is not very far from 100. We attach the obvious constraints for each i

$$n_i = \sum_m g_{im} = \sum_m h_{n-i-1,m} ,$$

and a non-obvious one given by an expression below equal to the right hand side of (1) for $k=4$ and $l=5$, which can be easily derived by analyzing the proof of Theorem 1.

$$2\sum_i\sum_m (E(3,5,i) - m) g_{im} + 2\sum_i\sum_m (m - e(4,4,i)) h_{im} = RHS(1). \qquad (2)$$

Theorem 3 implies further constraints. Let t_{im} and T_{im} be the known minimum and maximum, respectively, of the number of triangles in any $(4,4,i)$-good graph with m edges. Similarly, let s_{im} and S_{im} be the known bounds on the expression $2n_4(G)-n_2(G)-n_1(G)$ in any $(3,5,i)$-good graph G with m edges, where $n_j(G)$ is the number of vertices of degree j in G. Introduce two auxiliary variables T and S with the postulated meanings (S is the only variable which can be negative):

$$T = \sum_{x \in V(F)} t(H_x) \qquad \text{and} \qquad S = 4a_4 - 2a_2 - 2a_1,$$

which are reflected in the following two new constraints of LP3

$$\sum_i \sum_m h_{im} t_{im} \le T \le \sum_i \sum_m h_{im} T_{im},$$
$$\sum_i \sum_m g_{im} s_{im} \le S \le \sum_i \sum_m g_{im} S_{im}.$$

Finally, Theorem 3 itself becomes the last constraint of LP3 as follows:

$$3T = 3S + \sum_i \sum_m (n + 9 - 3i) m g_{im}.$$

n	lower bounds				upper bounds			
	LP0	LP1	LP2	LP3	LP3	LP2	LP1	LP0
24	72	101	101	109	138	139	154	156
25	88	116	116	123	145	148	160	162
26	104	130	130	138	152	154	169	169
27	122	153	153	154	158	160	171	175

Table I. Bounds for $e(F)$ in $(4,5,n)$-good graphs F from LPi, $0 \le i \le 3$.

Table I shows the results of optimizing the number of edges e for LPi, $0 \le i \le 3$. All of them, except LP0, have no feasible integer solutions for $n = 28$, thus showing $R(4,5) \le 28$. The minimization of $\Delta = g_{12,24} + h_{14,41} - n_{12}$ in LP3 for $n = 27$ yields $\Delta \ge 4$, thus in any (4,5,27)-good graph F there are at least 4 vertices $x \in V(F)$ such that

(a) G_x is an (3,5,12)-good graph with 24 edges, and
(b) H_x is an (4,4,14)-good graph with 41 edges.

There are only two possible graphs for G_x [Ka2], and 40 possibilities for H_x [MR2], totaling 80 possible pairs of graphs (G_x, H_x). The reconstruction of the possible graphs F from such pairs is time consuming, but already feasible.

Glue Algorithms. Given a $(3,5,n)$-good graph G and a $(4,4,m)$-good graph H, a *glue algorithm* is one that can find all $(4,5,n+m+1)$-good graphs F, such that for some vertex $x \in V(F)$ the graphs G_x and H_x are isomorphic to G and H, respectively.

Similar approaches for reconstructing $(3,l)$-good graphs have been described in [MZ,RK], for example. It is rather clear what has to be done but quite difficult to make it sufficiently efficient, and the *glue* algorithm needed here poses probably the highest complexity requirements amongst them. We have implemented two independent *glue* algorithms. One version roughly followed previous approaches; the other is sketched below.

Sketches of a glue algorithm. Let v be a vertex of the graph F whose neighborhood induces the graph G. The task of *glue* consists of deciding for each pair of vertices x, y, such that $x \in V(G)$ and $y \in V(H)$, whether $\{x, y\}$ is an edge of F. There are nm such pairs, which is much too many for an approach where each of them could be treated naturally as a 0-1 variable. The algorithms mentioned in the previous paragraph, as well as one of our implementations, manipulate certain sets of such variables as units, namely sets of edges with common vertex $x \in V(G)$ and the other endpoint in $V(H)$, let's call them *cones* with apex x, which can possibly appear in F. The second implementation considered as elementary units *regular sets of cones*; the following formalizes this concept.

Let LH be the lattice of all the subsets of $V(H)$, where the partial order is defined by inclusion. If $s \le S$ (i.e. $s \subseteq S \subseteq V(H)$) then $LH(s,S)$ denotes the sublattice of LH formed by all u such that $s \le u \le S$. The *dimension* of $L = LH(s,S)$ is denoted by $dim(L)$ and is equal to $|S| - |s|$. Note that L has $2^{dim(L)}$ elements, and each of them will be treated as a potential candidate for the base of a cone. For any $x \in V(G)$, $Reg(x,s,S)$ is called a *regular set of cones*, and it contains all cones with apex x and base in $LH(s,S)$. For every $u \subseteq V(H)$ let $\alpha(u)$ and $\beta(u)$ denote the clique and independence number of the subgraph of H induced by u. Assume that C_i and R_i, for $1 \le i \le 2$, are a cone and a regular set of cones, such that $C_i = (x_i, u_i)$, $s_i \le u_i \le S_i$ and $R_i = Reg(x_i, s_i, S_i)$. Note that if the cones C_1 and C_2 appear in F, then if $\{x_1, x_2\}$ is an edge of G then $\alpha(s_1 \cap s_2) \le 1$, and if it is not an edge then $\beta(\overline{S_1 \cup S_2}) \le 2$. These conditions permit us to eliminate many possible pairs of cones by doing just one test on the regular sets R_1 and R_2, since $\alpha(u)$ and $\beta(u)$ are computed and stored just once for all $u \in LH$. A recursive backtracking algorithm was implemented which assigns regular sets of cones to vertices in G. If we consider only regular sets $R = Reg(x, s, S)$ with $s = S$, i.e. of dimension 0, this approach is equivalent to simple cone manipulation. If the dimension of R is large (in which case pruning is unlikely) we can easily split it into two regular sets of dimension one less. If the dimension of the regular sets is kept at a moderate level, this algorithm outperforms previous approaches.

Table II contains the known exact values and best bounds for $e(4,5,n)$ and $E(4,5,n)$ we were able to derive. The exact values were found by explicit enumeration. For the inexact values, the upper bounds for $E()$ and lower bounds for $e()$ were obtained by optimizing the corresponding instances of the system LP3, except that the bound $E(4,5,26) \le 151$ was obtained by the *glue* algorithm. The upper bounds for $e()$ and lower bounds for $E()$ were established by constructing graphs; most of them can probably be improved, except 116 and 132 for $n = 24$, which we believe to be the true values. We currently know of over 300000 nonisomorphic (4,5,24)-good graphs, which were obtained by starting with a smaller set made by the *glue* algorithm and by Exoo [Ex2] using simulated annealing, then expanding that set by repeatedly taking induced subgraphs on 23 vertices and extending them back to 24 vertices in every possible way. None of these graphs extends to a (4,5,25)-good graph.

n	$e(4,5,n)$	$E(4,5,n)$	n	$e(4,5,n)$	$E(4,5,n)$	n	$e(4,5,n)$	$E(4,5,n)$
6	2	12	13	17	53	20	66-71	98-105
7	3	16	14	22	60	21	75-83	103-114
8	4	21	15	27	66	22	86-93	113-122
9	6	27	16	32	72	23	98-104	121-130
10	8	33	17	41	79-81	24	109-116	132-138
11	10	40	18	48-50	84-90	25	123-	-145
12	12	48	19	56-59	91-97	26	138-	-151

Table II. Values and bounds for $e(4,5,n)$ and $E(4,5,n)$.

The two *glue* algorithms agreed on many positive constructions, and both claimed that there is no (4,5,27)-good graph F for any of the 80 cases as in (a) and (b) above. Hence we have a new upper bound $R(4,5) \le 27$. Our current *glue* implementations are not adequate for deciding in a reasonable time whether $R(4,5) \le 26$, since there are many more than 80 cases and many are much harder. The number of cases could be decreased (perhaps to 0) by the discovery of a strong system LP4 or by finer analysis of the system LP3. A direct application of Theorem 1, as in [MR2], to the results from LP3 proves that $R(5,5) \le 52$, i.e. 3 less than the 1971 bound of Walker [Wa2], who essentially used calculations equivalent to LP1. We reported the bound $R(5,5) \le 53$ implied by LP2 in [MR2] shortly before constructing the system LP3. Similarly, the known values of $E(3,6,n)$ [MZ,RK], the lower bounds for $e(4,5,n)$ in Table II and Theorem 1 improve the Walker's upper bound $R(4,6) \le 44$ [Wa2] by one. Thus we have:

Theorem 4:

$$R(4,5) \le 27, \quad R(5,5) \le 52, \quad \textit{and} \quad R(4,6) \le 43.$$

3. General Algorithms and Other Combinatorial Configurations

In this section we briefly describe the most important algorithms used in this work. We would like to stress their general applicability in the searches for other Ramsey graphs, and also for other difficult combinatorial configurations like t-designs or covering set systems. A detailed knowledge of (3,6)-good and (4,5)-good (thus also (5,4)-good) graphs forms a firm base for deeper study of (5,5)-good and (4,6)-good graphs. All (3,6)-good graphs are known [MZ,RK], so we believe it is feasible to apply our techniques to obtain further improvements of the upper bounds for the numbers $R(5,5)$ and $R(4,6)$. Similarly, it also seems quite possible to improve all of the old, but not yet challenged, upper bounds derived by Giraud [Gi] in 1969. The major components of the software used in this work consisted of:

Nauty. *Nauty* is a very efficient set of procedures written by the first author [Mc] for determining the automorphism group of a graph, and optionally for canonically labeling it. Two graphs are isomorphic iff they have identical canonically labeled isomorphs; thus *nauty* can be used as a tool to detect isomorphs amongst large families of graphs and, indirectly via graphs, in families of any reasonable finite objects.

Constructing large instances of LP. Combinatorial theorems can often lead to systems of linear constraints of a size that depends on how fine a classification of subconfigurations we are willing to do. We can handle hundreds of variables, which makes a dramatic difference when compared to the few variables involved in classical proofs. Natural programs build such systems as input to the LP software.

Solving ILP. There are many good real and integer linear programming software packages which are publicly available, like LINDO [Sch] or MAPLE [CGGM], which were used in this work. We are also experimenting with our own algorithms for optimizing systems of rational equations and/or inequalities. They use existing real LP software for finding approximate optima, then "round" them to rationals, and finally try to verify them rigorously as rational optima with a separate algorithm. Such a tool is very useful in the search for integer optima.

Direct searches. If results of other techniques still admit an existence of a configuration searched for, then we develop a specialized search algorithm for the remaining possibilities. Typically, the results of LP provide a number of strong restrictions, which make such a search feasible. Some examples of algorithms of this kind are presented in [MZ,MR1], as well as the *glue* algorithm described in Section 2.

References

[CGGM] B.W. Char, K.O. Geddes, G.H. Gonnet, M.B. Monagan and S.M. Watt, MAPLE Reference Manual, *WATCOM Publications Limited,* Waterloo, 1990.

[Ex1] G. Exoo, A Lower Bound for $R(5,5)$, *Journal of Graph Theory,* 13 (1989) 97-98.

[Ex2] G. Exoo, *personal communication* (1991).

[Gi] G. Giraud, Nouvelles majorations des nombres de Ramsey binaires-bicolores, *C.R. Acad. Sc. Paris, A,* 268 (1969) 5-7.

[Ka1] J.G. Kalbfleisch, Construction of Special Edge-Chromatic Graphs, *Canadian Math. Bull.,* Vol. 8 (1965) 575-584.

[Ka2] J.G. Kalbfleisch, Chromatic Graphs and Ramsey's Theorem, *Ph.D. thesis,* University of Waterloo, January 1966.

[Mc] B.D. McKay, Nauty User's Guide (Version 1.5), *Technical Report TR-CS-90-02,* Computer Science Department, Australian National University (1990).

[MR1] B.D. McKay and S.P. Radziszowski, The First Classical Ramsey Number for Hypergraphs is Computed, *Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms,* SODA'91, San Francisco, (1991) 304-308.

[MR2] B.D. McKay and S.P. Radziszowski, New Upper Bound for the Ramsey Number $R(5,5)$, *to appear in Proceedings of Sixteenth Australasian Conference on Combinatorial Mathematics and Combinatorial Computing.*

[MZ] B.D. McKay and K. Zhang, The Value of the Ramsey Number $R(3,8)$, *to appear*.

[RK] S.P. Radziszowski and D.L. Kreher, On $R(3,k)$ Ramsey Graphs: Theoretical and Computational Results, *Journal of Combinatorial Mathematics and Combinatorial Computing*, 4 (1988) 37-52.

[Sch] L.E. Schrage, User's Manual for LINDO, *The Scientific Press*, 1981.

[Wa1] K. Walker, Dichromatic Graphs and Ramsey Numbers, *Journal of Combinatorial Theory*, 5 (1968) 238-243.

[Wa2] K. Walker, An Upper Bound for the Ramsey Number $M(5,4)$, *Journal of Combinatorial Theory*, 11 (1971) 1-10.