On Constructing Hypergraphs without Property B

Geoffrey Exoo[1]
Department of Mathematics and Computer Science
Indiana State University
Terre Haute, IN 47809

Abstract. A hypergraph has property B (or chromatic number two) if there is a set which intersects each of its edges, but contains none of its edges. The number of edges in a smallest n-graph which does not have property B is denoted $m(n)$. This function has proved difficult to evaluate for $n > 3$. As a consequence, several refinements and variations of the function m have been studied. In this paper we describe an effort to construct, via computer, hypergraphs that improve current estimates of such functions.

Introduction

An *n-graph* (or *n-uniform hypergraph*) H is a pair (V, E), where $V = V(H)$ is a finite set (the vertices) and $E = E(H)$ is a collection of n element subsets of V (the edges). The *chromatic number* of an n-graph H is the minimum number of colors which can be assigned to $V(H)$ so that no edge is monochromatic. An n-graph whose chromatic number is two is said to have *property B*. Erdós and Hajnal [11] defined $m(n)$ as the number of edges in a smallest n-graph which does not have property B (i.e., has chromatic number three or more), and $m_k(n)$ as the same minumum, restricted to hypergraphs on k vertices. He also refined the definition of Property B to that of property $B(s)$ so that for a hypergraph H to have property $B(s)$ there must be a set $S \in V(H)$ which contains at least one but fewer that s vertices from each edge. Then $m(n, s)$ is the minimum number of edges in an n-graph which does not have property $B(s)$, and $m(n, n) = m(n)$.

The study of the behavior of these functions was initiated by Erdós in a series of three papers [8],[9],[10]. In [8], he noted that $m(2) = 3$ and $m(3) = 7$. The current lower bound for $m(4)$ seems to be 19, though a proof has not been published. The upper bound is 23 [12], [14]. The current upper bound for $m(5)$ is 51 [1]. For $m(n)$ in general we know

$$2^n n^{\frac{1}{3}+o(1)} \leq m(n) \leq (1+\epsilon)e \log(2) n^2 2^{n-2}$$

with the lower bound due to Beck [6] and the upper bound to Erdós [9]. Spencer [13] gives a short proof of Beck's result. In [10] Erdós found that

$$m_{2n-1}(n) = m_{2n}(n) = \binom{2n-1}{n}$$

[1]Research supported by Office of Naval Research Grant No. N0014-88-K0211

and posed the problem of computing $m_{2n+1}(n)$, and in particular $m_9(4)$. Abbott and Liu [3] showed that $24 \leq m_9(4) \leq 26$. Then De Vries [7] proved that

$$m_{2n+1}(n) = m_{2n+2}(n) = \frac{1}{n}\binom{2n+1}{n-1}$$

if and only if there is a Steiner system $S(n-1, n, 2n+1)$. Since an $S(4,5,11)$ is known to exist, this gives $m_{11}(5) = m_{12}(5) = 66$.

Regarding the functions $m(n, s)$, Abbott [9] determined that $m(n, 2)$ is 3 if n is even, and 4 if n is odd. Abbott and Liu [2] showed that $m(n, 3) = 7$ if n is a multiple of 3 or 4 and that $7 \leq m(n, 3) \leq 10$ for all n. They also showed that $m(n, 4) \leq 29$ except for (perhaps) $n = 5, 6, 7, 10, 19, 21, 23, 29, 31, 37, 38, 46$, or 47 [4],[5].

In this paper we shall describe two algorithms for constructing 3-chromatic hypergraphs and show some of the constructions they have produced.

Algorithms

The first algorithm can be viewed as an extension of the greedy algorithm as used by Erdős in [9] to prove the upper bound for $m(n)$ cited above. Suppose we wish to construct an n-graph, H, on k vertices that does not have property B. Let C be a list of all possible 2-colorings of the vertex set V. For each coloring in C we will keep a count of the number of times it is *marked*. This value is initially set to zero for each coloring. We wish to construct a set E of edges for our hypergraph. Candidate edges for E will be selected from a set F.

When the algorithm begins, E is the empty set, while F contains all n-subsets of V. Edges for E are selected, one at a time, from F. The selection method is greedy, in the sense that we always select an edge which is monochromatic under the largest number of colorings in F which have zero marks. In effect the selected edge eliminates the largest number of colorings as good colorings of the hypergraph. In case of ties, the decision is made randomly, or by a criterion based on intersection cardinalities. (One apparently effective method is to choose the edge which has an intersection of cardinality one with the largest number of edges in E.) When there are no longer any unmarked colorings in C, this phase of the algorithm terminates. Note termination is guaranteed when $k \geq 2n+1$.

At this point it may be possible to eliminate some edges from E. Some edges may be redundant in that they eliminate colorings which are also eliminated by other edges. Such edges can be identified as those that are monochromatic only under colorings that have at least two marks. This elimination is phase two of our algorithm.

The third and final phase involves exchanging edges between E and F. We look for pairs of edges, $e \in E$ and $f \in F$, such that the hypergraph with edge set $E \cup \{f\} - \{e\}$ does not have property B. After exchanging such pairs of edges, we check to see if any new redundant edges have been created. In practice it turns out to be more effective to perform a (small) random number of exchanges before looking for redundant edges. The algorithm terminates when such exchanges fail to produce an improvement after a specified number of iterations. We use an integer variable, t, which counts the number of iterations since the last improvement, and an integer constant, max, which is used to decide when to terminate the procedure.

The algorithm can be outlined as follows:

0. Let all members of C be unmarked, let $E = \emptyset$, let $F = \binom{V}{n}$, and let $t = 0$.
1. While there are unmarked elements in C:
   a. among all $x \in F$ choose one, x_0, that is monochromatic under the largest number of unmarked colorings in C; in case of ties choose x_0 so that $|\{y \in E : |x \cap y| = 1\}|$ is as large as possible; if this is not decisive, choose randomly;
   b. move x_0 from F to E;
   c. mark all colorings in C that make x_0 monochromatic.
2. If there are any redundant edges, move them from E back to F, set $t = 0$; else increment t and if $t = max$ terminate.
3. Exchange edges between E and F without introducing property B, then go to step 2; if no such exchanges can be made, then terminate.

Experience has shown that if the tie breaking criterion in step 1a is eliminated, performance suffers. The analysis of this phenomenon seems difficult in general, but we offer some data for the case of $m_{13}(6)$. If we eliminate steps 2, 3, and the edge intersection criterion (call this the *simple algorithm*), the hypergraphs produced will have an average of 325 edges. If step 2 is done exactly once, the average falls to 323. If we include steps 2 and 3, the average is approximately 320 edges. If we run the algorithm as given above, using the edge intersection criterion, the average falls to 316. These averages were computed after running 10000 trials for each version. When running the simple algorithm, we correlated the number of edges with the fraction of edge pairs that intersected with cardinality one. The sample correlation coefficient was approximately -0.57.

The motivation for investigating the edge intersection criterion is the prevalence of edge pairs that satisfy the condition in known good constructions (e.g., the Fano configuration) and on the following simple computation. In an n-graph on k vertices, an edge eliminates 2^{k-n-1} colorings as good vertex 2-colorings. Two edges that intersect with cardinality $i > 0$ eliminate

$$2^{k-n+2} - 2^{k-2n+i+1}$$

colorings. The figure for two disjoint edges is the same as for the case $i = 1$. So these two cases give the largest possible values. Looking at triples of edges, we see that the maximum is achieved by three edges that pairwise intersect at one vertex and for which the intersection of all three is empty. The analysis becomes messy for four edges, but the pattern holds. The statistical evidence cited above (concerning the correlation coefficent) provided further motivation for investigating a tie breaking criterion based on edge intersection cardinalities.

In our second algorithm we take a somewhat different approach. Again we wish to construct an n-graph on k vertices that does not have property B. Let $r = \binom{k}{n}$, and $e_1, e_2, \cdots, e_r$ be a list of the n-subsets of a k-set V. Let σ be a permutation of $\{1, \cdots, r\}$. Define $f(\sigma)$ to be the minimum i such the n-graph with edge set $\{e_{\sigma(1)}, \cdots, e_{\sigma(i)}\}$ does not have property B. If $k \geq 2n + 1$ then i exists. This observation can be made into an algorithm by considering another permutation, σ', which is by some measure close to σ. Our method is to use permutations σ' that differ from σ by a transposition (ij) where $\sigma(i) \leq f(\sigma)$ and $\sigma(j) > f(\sigma)$. The values of $f(\sigma)$ and $f(\sigma')$ are compared, and the permutation which gives the smaller value is retained (ties are decided randomly). This process is repeated indefinitely.

In terms of computer time, the computation of $f(\sigma)$ is very expensive. So we considered ways to identify good candidates for σ' without actually doing the full computation. Based on our experience with algorithm 1, it seemed natural to look at the intersections among edges. At any stage in the algorithm the edges e_i for which $\sigma(i) \leq f(\sigma)$ constitute the edge set of a 3-chromatic hypergraph, $H = H(\sigma)$. Define, for any $x \in E(H), g(x) = (r_0, \cdots, r_{n-1})$, where r_t is the number of $y \in E(H)$, $y \neq x$ such that $|x \cap y| = t$. We shall loosely speak of $g(x)$ as x's *intersection vector*. One can also define the distance between such vectors in a variety of ways. We use the metric of the taxicab geometry and the notation $||g(e) - g(f)||$. These ideas suggest the procedure given below (which uses t and max to make termination decisions as in algorithm 1). The procedure uses an *ideal intersection vector*, $\widehat{r}$, which should be viewed as a parameter. The outline of the procedure is:

0. Generate a random permutation σ of $\binom{V}{n}$, and set $t = 0$.
1. Choose (randomly) i such that $\sigma(i) \leq f(\sigma)$ and j such that $\sigma(j) > f(\sigma)$, and then let σ' be the product of σ and the transposition (ij). Increment t.
2. Compute $d = ||\widehat{r} - g(e_{\sigma(i)})||$ and $d' = ||\widehat{r} - g(e_{\sigma'(i)})||$. If $d < d'$ go back to step 1.
3. If $f(\sigma') < f(\sigma)$ then replace σ by σ', set $t = 0$, and go to step 1.
4. If $f(\sigma') = f(\sigma)$ then replace σ by σ' with probability $\frac{1}{2}$.
5. If $t < max$, then go to step 1, else terminate.

This procedure is used inside algorithm 2, which attempts to optimize with respect to the ideal intersection vector. The idea here is begin with a small number of candidates for the ideal intersection vector, run the procedure outlined above with each of them, and compare results. The best few are kept, the worst ones are replaced by random perturbations of the best ones, and the process is repeated. Choices for the initial set of ideal intersection vectors can be made randomly or by examining known good constructions, or by idle speculation. Such an algorithm can make effective use of a parallel machine.

Constructions

Our initial objective was to improve the upper bound for $m(4)$. While this has not been done, the behavior of algorithm 2 on that problem may be of interest. When the algorithm is applied to the $m(4)$ problem with $k = 11$, the Seymour-Toft construction is duplicated on roughly 40% of the runs. On the other 60% of the runs the algorithm gets "stuck" at 25. When we try $k > 11$, the Seymour-Toft construction is still obtained (though less often). In these cases, the extra vertices are not used. This is true even when we do not include step 2 in the algorithm. Since such a simple algorithm does not seem to introduce any biases that might favor one particular construction, it is tempting to see this as evidence that 23 is the correct value.

As noted above, the value of $m_{11}(5)$ was determined by De Vries [7] to be 66. The extremal hypergraph is the unique Steiner system $S(4,5,11)$. Surprisingly, algorithm 1 will reproduce that Steiner system on approximately 25% of the runs. For the next three instances of $m_{2n+1}(n)$ problem, there is no Steiner system [7]. Theorem 1 gives the best upper bounds produced by algorithm 1. It is worth noting that, according to [7], the first case where the existence question for the relevant Steiner system has not been settled is $m_{23}(11)$.

Theorem 1.

a) $m_{13}(6) \leq 302$
b) $m_{15}(7) \leq 1041$
c) $m_{17}(9) \leq 3799$

Proof: The construction shown in Figure 1 proves (a). The constructions for (b) and (c) would require many pages to present, and so are omitted.

The next theorem deals with $m_{10}(4)$. Recall that upper bounds for $m_9(4)$ and $m_{11}(4)$ are 26 and 23 repsectively. Both of our algorithms have produced constructions for all three of these upper bounds.

Theorem 2. $m_{10}(4) \leq 25$.

Proof: The theorem is proved by the construction in Figure 2.

The last theorem details some improvement in a theorem of Abbott and Liu. Further improvement may be possible, but, using our methods, would require an enormous amount of computer time. All three of the hypergraphs used in theorem 3 were produced by algorithm 2.

Theorem 3. *$m(n,4) \leq 29$, except possibly for $n = 19, 23, 29, 31, 37, 28, 46$, and 47.*

Proof: The statement of the theorem is just that of Theorem 2 in [5], except that the cases n = 5, 6 ,7, 10, and 21 have been settled. Using the inequality $m(rn,s) \leq m(n,s)$ from [2], it will suffice to settle the cases n = 5, 6, and 7. The hypergraphs for these cases are shown in Figures 3, 4, and 5. Note that we actually show $m(5,4) \leq 27$.

12345*a*	12346*c*	12347*b*	123489	1234*bd*	123567	12358*c*	12359*b*
12368*b*	12369*a*	12378*a*	12378*d*	12379*c*	1237*ad*	1239*cd*	123*abc*
124569	12457*c*	12458*b*	1245*cd*	124678	1246*ab*	1246*cd*	12479*a*
12479*d*	1247*ad*	1248*ac*	1249*bc*	124*abd*	12567*d*	12568*a*	12568*d*
1256*bc*	125789	1257*ab*	1257*bd*	1259*ac*	1259*ad*	12679*b*	1267*ac*
1267*cd*	12689*c*	12689*d*	126*abd*	1278*bc*	1289*ab*	1289*bd*	128*acd*
134569	13456*d*	13457*c*	13458*d*	13459*c*	13467*b*	13467*d*	13468*a*
1346*ad*	1346*cd*	134789	13479*a*	1347*ad*	1347*cd*	13489*c*	13489*d*
1348*bc*	1349*ad*	134*abc*	135689	1356*ac*	1356*bd*	135789	1357*ab*
13589*c*	1358*ad*	1358*cd*	1359*ab*	135*acd*	13678*b*	13678*c*	1368*ad*
1369*ac*	1369*bc*	136*acd*	1379*bd*	1379*cd*	1389*ab*	138*abc*	138*bcd*
14567*a*	14569*c*	14578*d*	14579*b*	1457*ac*	1458*ab*	1458*ad*	1459*bd*
14679*b*	1467*bc*	14689*b*	1468*ac*	1469*bd*	146*abd*	14789*c*	1478*bd*
1479*ad*	1489*cd*	148*acd*	149*abc*	14*abcd*	15678*b*	15679*c*	15679*d*
1567*ad*	1567*cd*	1568*bc*	1568*cd*	1569*ad*	156*abc*	15789*c*	1578*ac*
1579*cd*	1589*ad*	1589*bc*	158*bcd*	159*bcd*	16789*a*	1678*bd*	1678*cd*
1679*ad*	1689*ab*	1689*bd*	178*abd*	179*abc*	17*abcd*	234567	23456*a*
23456*b*	234578	23457*b*	23458*a*	23459*d*	2345*ad*	234679	23468*c*
23468*d*	2346*ac*	2346*cd*	23479*c*	2347*bd*	23489*b*	2348*cd*	2349*ab*
23567*c*	23568*a*	23569*d*	2356*cd*	23579*a*	2357*bc*	23589*b*	2358*ab*
2358*bd*	2359*bc*	2359*cd*	235*abc*	235*abd*	23678*b*	2367*ab*	2367*bd*
23689*a*	23689*c*	2369*ad*	236*bcd*	23789*a*	23789*b*	2378*ad*	2378*cd*
2379*ac*	237*abc*	237*bcd*	2389*ac*	2389*ad*	23*abcd*	245678	24569*b*
2456*ac*	256*bc*	2456*bd*	24578*c*	2457*ab*	24589*a*	24589*c*	24589*d*
2458*cd*	24678*d*	2467*ac*	2468*ad*	2468*bc*	2469*ac*	2469*ad*	2478*ab*
2479*cd*	247*bcd*	2489*ad*	218*acd*	249*abc*	249*bcd*	25678*a*	25679*c*
2568*ad*	2569*ab*	2569*cd*	2578*cd*	2579*ac*	2579*ad*	257*acd*	2589*bd*
258*abd*	26789*a*	26789*c*	2679*bd*	267*acd*	267*bcd*	2689*ac*	268*abc*
269*bcd*	2789*cd*	279*abd*	29*abcd*	34568*a*	34568*b*	3456*bc*	3456*bd*
34578*b*	34578*d*	34579*a*	3459*bc*	3459*cd*	346789	34679*d*	3467*ab*
3468*ad*	3469*ac*	3469*bc*	34789*b*	3478*ac*	347*acd*	348*bcd*	349*abd*
35678*b*	35679*a*	3567*ad*	35689*d*	3569*ab*	356*acd*	35789*d*	3579*ac*
357*bcd*	3589*ac*	358*abd*	36789*d*	3678*ac*	3678*cd*	367*abc*	368*abd*

369 *bcd*	3789 *bc*	379 *abd*	389 *acd*	45678 *c*	45679 *c*	4567 *ad*	45689 *c*
4568 *cd*	4578 *ac*	4579 *ab*	4579 *bd*	4589 *ad*	458 *abc*	45 *abcd*	46789 *a*
467 *abc*	467 *bcd*	4689 *ab*	4689 *bd*	468 *abc*	469 *acd*	4789 *bc*	4789 *cd*
478 *abd*	56789 *b*	5678 *bd*	5679 *bc*	5679 *cd*	5689 *bd*	569 *abd*	578 *abc*
578 *acd*	589 *bcd*	6789 *ad*	689 *abc*	68 *abcd*	79 *abcd*		

Figure 1. The edge list of a hypergraph that shows $m_{13}(6) \leq 302$. The vertices are represented by hexadecimal digits.

1	2	5	10
1	2	6	8
1	2	9	10
1	3	4	7
1	3	4	9
1	3	7	9
1	4	6	8
1	4	6	8
1	5	6	8
1	5	9	10
1	6	7	8
2	3	5	8
2	3	6	10
2	3	8	9
2	4	5	7
2	4	6	9
2	6	7	9
3	4	6	10
3	5	6	10
3	5	8	9
3	6	7	10
4	5	6	9
4	7	8	10
4	8	9	10
5	6	7	9
7	8	9	10

Figure 2. The edge list of a hypergraph that shows $m_{10}(4) \leq 25$.

1	3	8	9	10
2	4	5	6	10
2	3	8	10	11
1	5	6	7	10

3	6	7	9	11
2	3	4	6	8
2	5	7	10	11
2	4	7	8	11
1	2	6	9	11
4	5	6	7	8
1	2	3	5	10
3	5	7	8	9
4	6	9	10	11
1	3	4	5	11
1	7	8	10	11
2	3	6	9	11
4	5	9	10	11
3	5	8	9	10
3	4	7	9	10
1	2	4	7	9
5	6	8	9	11
4	5	7	8	11
2	3	5	6	7
1	3	6	7	11
1	2	5	8	9
1	3	6	7	8
1	4	6	8	10

Figure 3. The edge list of a hypergraph showing $m(5,4) \leq 27$.

1	2	3	5	7	10
1	2	3	6	7	9
1	2	3	6	8	10
1	2	3	7	8	11
1	2	4	5	6	11
1	2	4	8	10	11
1	2	5	7	9	11
1	2	7	8	9	11
1	3	4	5	8	9
1	3	4	5	10	11
1	3	4	6	8	9
1	3	4	7	9	11
1	3	5	6	9	10
1	3	5	7	8	11
1	4	5	6	7	8
1	4	7	8	9	10
1	5	6	7	10	11

2	3	4	5	9	10
2	3	4	6	10	11
2	3	5	8	9	11
2	4	5	7	8	10
2	4	6	7	9	11
2	4	6	8	9	10
2	5	6	7	9	10
2	5	6	8	9	11
3	4	5	6	7	8
3	4	7	9	10	11
3	6	7	8	10	11
5	6	8	9	10	11

Figure 4. The edge list of a hypergraph showing $m(6,4) \leq 29$.

1	2	3	4	5	7	8
1	2	3	4	9	11	12
1	2	3	5	6	10	13
1	2	3	7	11	12	13
1	2	4	7	10	12	13
1	2	5	6	7	9	12
1	2	5	7	8	10	13
1	2	6	8	9	10	11
1	3	4	6	7	8	10
1	3	4	9	10	11	13
1	3	5	6	8	11	13
1	3	5	8	10	11	12
1	4	5	7	8	9	12
1	4	5	8	11	12	13
1	4	6	9	10	12	13
1	5	6	7	9	11	13
2	3	4	5	8	9	13
2	3	4	6	11	12	13
2	3	5	6	7	11	12
2	3	6	7	8	9	13
2	4	5	6	10	12	13
2	4	6	8	9	11	13
2	4	7	8	10	11	12
2	5	7	9	10	11	13
3	4	5	6	7	10	11

3	4	5	7	8	9	11
3	5	6	8	9	10	12
3	7	8	9	10	12	13
4	6	7	8	11	12	13

Figure 5. The edge list of a hypergraph showing $m(74) \leq 19$.

References

1. H. L. Abbott and D. Hanson, *On a Combinatorial Problem of Erdös*, Canad. Math. Bull. **12** (1969), 823–829.
2. H. L. Abbott and A. C. Liu, *On Property* $B(s)$, Ars Combinatoria **7** (1979), 255–260.
3. H. L. Abbott and A. C. Liu, *On Property B of Families of Sets*, Canad. Math. Bull. **23** (1980), 429–435.
4. H. L. Abbott and A. C. Liu, *On Property* $B(s)$, *II*, Discrete Math. **37** (1981), 135–141.
5. H. L. Abbott and A. C. Liu, *On Property* $B(4)$ *of Families of Sets*, Ars Combinatoria **26** (1988), 59–68.
6. J. Beck, *On 3-chromatic hypergraphs*, Discrete Math. **24** (1978), 127–137.
7. H. L. De Vries, *On Property B and on Steiner Systems*, Mathematische Zeitschrift **153** (1977), 155–159.
8. P. Erdös, *On a combinatorial problem*, Nordisk Mat. Tidskr. **11** (1963), 5–10.
9. P. Erdös, *On a combinatorial problem II*, Acta Math. Acad. Sci. Hungar. **15** (1964), 445–447.
10. P. Erdös, *On a combinatorial problem III*, Canad. Math. Bull. **12** (1969), 413–416.
11. P. Erdös and A. Hajnal, *On a Property of Families of Sets*, Acta Math. Acad. Hung. Sci. **12** (1981), 87–123.
12. P. D. Seymour, *A Note on a Combinatorial Problem of Erdős and Hajnal*, Bull. London Math. Soc. **2:8** (1974), 681–682.
13. J. Spencer, *Coloring n-sets Red and Blue*, J. Combin. Theory, Series A **30** (1981), 112–113.
14. B. Toft, *On Colour-Critical Hypergraphs*, in "Infinite and Finite Sets", Ed. A. Hajnal et al, North Holland Publishing Co., 1975, pp. 1445–1457.