Lecturers: Hung Q. Ngo and Atri Rudra
Last update: August 23, 2011

(Strongly) Explicit Constructions

1 A greedy algorithm

Let S be the collection of all binary row vectors of length N, each with weight w (i.e. each vector has w non-zero entries), where $w \leq Nd$. We construct a d-disjunct matrix $\mathbf{M}$ by picking members of S to be rows of $\mathbf{M}$. For $\mathbf{M}$ to be d-disjunct, for each $j \in [N]$ and each d-subset $A \in \binom{[N]}{d}$, $j \notin A$, we want to pick d a row $\mathbf{s} \in S$ such that $s_j = 1$ and $\mathbf{s}_A = \mathbf{0}$, in which case we say that s "covers" (j, A). The main objective is to pick a small subset of S which covers all $(d+1)\binom{N}{d+1}$ possible pairs (j, A). This is an instance of the SET COVER each "set" is a member of S and the universe consists of pairs (j, A) as described.

A natural algorithm for the SET COVER problem is to pick a set which covers as many uncovered elements as possible, then remove all covered elements and repeat until all elements are covered. This is the well-known greedy algorithm for SET COVER. A classic result by Lovász [4] (and independently by Chvátal [1]) implies that the greedy algorithm finds a set cover for all the (j, A) of size at most

$$\begin{aligned}
t &\leq \frac{\binom{N}{w}}{\binom{N-d-1}{w-1}} \left(1 + \ln\left(w\binom{N-w}{d}\right)\right) \\
&= \frac{N-d}{w} \cdot \frac{N}{N-d} \cdot \frac{N-1}{n-d-1} \cdots \frac{N-w+1}{N-d-w+1} \left(1 + \ln\left(w\binom{N-w}{d}\right)\right) \\
&< \frac{N-d}{w} \left(\frac{N-w+1}{N-d-w+1}\right)^w \left(1 + \ln w + d \ln\left(\frac{e(N-w)}{d}\right)\right) \\
&= \frac{N-d}{w} \left(1 + \frac{d}{N-d-w+1}\right)^w \left(1 + d + \ln w + d \ln\left(\frac{N-w}{d}\right)\right) \\
&< \frac{N-d}{w} e^{\frac{dw}{N-d-w+1}} \left(1 + d + \ln w + d \ln\left(\frac{N-w}{d}\right)\right).
\end{aligned}$$

This fact can also be seen from the *dual-fitting analysis* of the greedy algorithm for SET COVER [7]. This set cover is exactly the set of rows of the d-disjunct matrix we are looking for. The final expression might seem a little unwieldy. Note, however, that for most meaningful ranges of w and d, the factor $\left(1 + d + \ln w + d \ln\left(\frac{N-w}{d}\right)\right)$ can safely be thought of as $O(dln(N/d))$. Furthermore, if $dw = O(N)$ then $e^{\frac{dw}{N-d-w+1}} = O(1)$ and so the number of rows t of the matrix is not exponential. Also, when $dw = \Theta(N)$ the overall cost is $t = O(d^2 log(N/d))$, matching the best known bound for disjunct matrices. This optimality only applies when we are free to choose w in terms of N and d; in particular, when we have this freedom we can pick $w = \Theta(N/d)$.

Exercise 1. Suppose instead of applying the greedy algorithm, we simply pick independently each round a random member $\mathbf{s}$ of S to use as a row of $\mathbf{M}$. In expectation, how many rounds must be performed so that $\mathbf{M}$ is d-disjunct. You should set $w = N/d$, and assume $\binom{d+2}{2} < N$ as usual.

Hwang and Sós [2] gave a different greedy algorithm achieving asymptotically the same number of tests. These algorithms have running time $\Omega(Nd)$, and thus are not practical unless d is a small constant.

2 Concatenating the Reed-Solomon code with the identity code

Code concatenation seems like a neat little trick to construct disjunct matrices. However, how do we choose the inner and outer codes? What are the necessary and/or sufficient conditions required on the properties of the codes so that the concatenation is d-disjunct? We will derive a simple sufficient condition due to Kautz and Singleton [3] (this is the same Richard Collom Singleton of the Singleton bound fame). Kautz and Singleton studied and constructed the so called *superimposed codes* which turn out to be equivalent to disjunct matrices. Their influential 1964 paper was also the first to give a strongly explicit construction of disjunct matrices with $t = O(d^2 \log^2 N)$, which we present in this section.

We first need a simple lemma which relates the weights of the codewords and their pairwise intersections to disjunctiveness. Two columns of a binary matrix "intersects" at a row if both columns contain a 1 on that row.

Proposition 2.1. *Let $\mathbf{M}$ be a binary matrix such that each column has weight at least w and every two different columns intersect at at most λ rows. Then, $\mathbf{M}$ is a d-disjunct matrix for any $d \leq (w-1)/\lambda$.*

Proof. Consider $(d+1)$ arbitrary columns $C_0, \ldots, C_d$ of $\mathbf{M}$. When $w \geq 1 + d\lambda$, there must be at least one row in which C_0 has a 1 and the other d columns have 0s. □

Lemma 2.2. *Suppose C_{out} is an $(n, k, \Delta)_q$-code, and the matrix corresponding to C_{in} is d-disjunct, then $\mathbf{M} = C_{\text{out}} \circ C_{\text{in}}$ is d-disjunct if $n > d(n - \Delta)$.*

Proof. Consider $(d+1)$ arbitrary columns $C_0, \cdots, C_d$ of $\mathbf{M}$. Every two codewords of the outer code share symbols in at most $n - \Delta$ positions. Hence, there is a position $p \in [n]$ such that the codeword C_0 (we overload notation a little here) has a symbol different from all the symbols of $C_1, \ldots, C_d$. Due to the fact that the inner matrix is d-disjunct, there is a row in $\mathbf{M}$ belonging to this position which C_0 has a 1 and the other d columns all have 0s. □

Open Problem 2.3. The above sufficient condition might be a little too strong to derive good bounds. Find a more "relaxed" condition.

Corollary 2.4. *Let $k \leq n \leq q$ be positive integers with q a prime power. Let C_{out} be the $[n, k]_q$-RS code, and C_{in} be the ID_q code. Then, $\mathbf{M} = C_{\text{out}} \circ C_{\text{in}}$ is a strongly explicit d-disjunct matrix for any $d \leq (n-1)/(k-1)$.*

Corollary 2.5. *Given $1 \leq d < N$, there is a strongly explicit d-disjunct matrix with N columns and $t = O(d^2 \log^2 N / \log^2(d \log N))$ rows.*

Proof. We want to pick parameters $k \leq n \leq q$ such that $d \leq (n-1)/(k-1)$ and $N \leq q^k$, and then apply the previous corollary. To make the calculation simpler, we replace the constraint $d \leq (n-1)/(k-1)$ by $d \leq n/k$. This replacement is OK because $n/k \leq (n-1)/(k-1)$.

Let us ignore the integrality issue for the moment. Suppose we pick $n = q$ and $\log N = k \log q$. Then, we need $d \log N / \log q \leq q$. Hence, we should pick q to be the smallest number such that $q \log q \geq d \log N$. Let's pick $n = q \approx \frac{2d \log N}{\log(d \log N)}$, and then set $k \approx \log N / \log q$. The overall number of tests is $t = qn = \Theta\left(d^2 \log^2 N / \log^2(d \log N)\right)$. With the integrality issue taken into account, it is also not hard to see that the same bound holds. □

Nguyen and Zeisel [5] used Lemma 2.2 and a result by Zinoviev [8] to prove an interesting upper bound on $t(d, N)$. The main idea is to recursively apply Lemma 2.2 many times with suitably chosen parameters.

3 The Explicit Construction by Porat-Rothschild

Porat and Rothschild [6] used Lemma 2.2 where C_{out} is a random code meeting the Gilbert-Varshamov bound and C_{in} is the identity code. They also showed how to derandomize the C_{out} construction, effectively provided a poly-time construction of a good disjunct matrix. We discuss their result in this section.

Recall the asymptotic GV-bound states the following. Let $q \geq 2$ be any prime power, $0 < \delta \leq 1 - 1/q$, and $n \geq 2$ be any integer. Then, for any integer $k \leq (1 - H_q(\delta))n$ there exists an $[n, k, \delta n]_q$-code. We proved the result using the union bound. Here, we prove it again using the argument from expectation which is essentially the same as the union bound proof. However, the argument from expectation allows us to derandomize the probabilistic proof using the conditional expectation method.

Pick a random $n \times k$ generator matrix $\mathbf{G}$. For each non-zero vector $\mathbf{y} \in \mathbb{F}_q^k$ let $\mathbf{1}_{\text{wt}(\mathbf{Gy})<\delta n}$ denote the indicator variable for the event that $\text{wt}(\mathbf{Gy}) < \delta n$. Note that

$$\text{Prob}[\mathbf{1}_{\text{wt}(\mathbf{Gy})<\delta n} = 1] \leq \frac{\text{Vol}_q(n, \delta n)}{q^n} \leq q^{n(H_q(\delta)-1)}.$$

Define

$$goal(\mathbf{G}) = \sum_{\mathbf{0}\neq\mathbf{y}\in\mathbb{F}_q^k} \mathbf{1}_{\text{wt}(\mathbf{Gy})<\delta n}.$$

Then, by linearity of expectation

$$\text{E}[goal(\mathbf{G})] = \sum_{\mathbf{0}\neq\mathbf{y}\in\mathbb{F}_q^k} \text{E}[\mathbf{1}_{\text{wt}(\mathbf{Gy})<\delta n}] = \sum_{\mathbf{0}\neq\mathbf{y}\in\mathbb{F}_q^k} \text{Prob}[\text{wt}(\mathbf{Gy}) < \delta n] < q^k q^{n(H_q(\delta)-1)} \leq 1.$$

To find a particular $\mathbf{G}$ for which $goal(\mathbf{G}) < 1$, we set the entries g_{ij} of $\mathbf{G}$ one by one, row by row from top to bottom. At each step, we choose a $g_{ij} \in \mathbb{F}_q$ which minimizes the conditional expectation $\text{E}[goal(\mathbf{G}) \mid \mathbf{G}_{(i,j)}]$ where $\mathbf{G}_{(i,j)}$ denote the matrix $\mathbf{G}$ with all entries up to the (i, j)th entry already chosen. It remains to show how to compute the conditional expectations $\text{E}[goal(\mathbf{G}) \mid \mathbf{G}_{(i,j)}]$ in polynomial time. Fix a non-zero vector $\mathbf{y} \in \mathbb{F}_q^k$. Suppose $\mathbf{G}_{(i,j)-1}$ has guaranteed w non-zero entries for $\mathbf{Gy}$ in the fully filled rows of $\mathbf{G}$. If $w \geq \delta n$ then $\text{Prob}[\text{wt}(\mathbf{Gy}) < \delta n \mid \mathbf{G}_{(i,j)}] = 0$.

Assume $w < \delta n$. Let $F(x; m, p)$ denote the cumulative distribution function of the binomial distribution with parameters (m, p), i.e.

$$F(x; m, p) := \sum_{i=0}^{\lfloor x \rfloor} \binom{m}{i} p^i (1-p)^{m-i}.$$

If $\mathbf{y}_{j+1..n} = \mathbf{0}$ and the partially filled row i of $\mathbf{G}$ has a non-zero dot product with $\mathbf{y}_{1..j}$ then

$$\text{Prob}[\text{wt}(\mathbf{Gy}) < \delta n \mid \mathbf{G}_{(i,j)}] \leq F(\delta n - w - 1; n - i, 1 - 1/q).$$

If $\mathbf{y}_{j+1..n} = \mathbf{0}$ and the partially filled row i of $\mathbf{G}$ has a zero dot product with $\mathbf{y}_{1..j}$ then

$$\text{Prob}[\text{wt}(\mathbf{Gy}) < \delta n \mid \mathbf{G}_{(i,j)}] \leq F(\delta n - w; n - i, 1 - 1/q).$$

Finally, if $\mathbf{y}_{j+1..n} \neq \mathbf{0}$ then

$$\text{Prob}[\text{wt}(\mathbf{Gy}) < \delta n \mid \mathbf{G}_{(i,j)}] \leq F(\delta n - w; n - i - 1, 1 - 1/q).$$

In all cases, we can compute the conditional probabilities and thus the conditional expectation $\text{E}[goal(\mathbf{G}) \mid \mathbf{G}_{(i,j)}]$. The running time is $q^k \text{poly}(n, q)$, which could be reduced a little by designing an auxiliary array which computes the change in expectation rather than re-computing the conditional expectation each time.

To apply Lemma 2.2, we want to pick parameters n, k, q, δ such that

$$\begin{aligned} q^k &\geq N \\ k &\leq (1 - H_q(\delta))n \\ n &> d(n - \delta n) \\ \delta &\leq 1 - \frac{1}{q}. \end{aligned}$$

The end result will be a $O(d^2 \log N)$-row d-disjunct matrix with N columns, which is explicitly (but not strongly explicitly) constructable.

The constraint $n > d(n - \delta n)$ is equivalent to $\delta > 1 - 1/d$. We pick $\delta = 1 - 1/(d + 1)$ which satisfies the constraint. Obviously, we do not want large δ because that only can reduce the size of the code. We ignore the issue of integrality for the sake of clarity. After picking $\delta = d/(d + 1)$, to satisfy $\delta \leq 1 - \frac{1}{q}$ we need $q \geq d + 1$. Then, set $k = \log_q N$ and $n = \frac{k}{1 - H_q(\delta)}$. The overall number of rows of the d-disjunct matrix is

$$t = nq = \frac{q}{(1 - H_q(\delta)) \log q} \log N.$$

To minimize this expression, we choose q to be as small as possible, which is the least power of 2 greater than $2d$. Hence, $q = \Theta(d)$. We need to estimate the value of $1 - H_q(\delta)$. We will use the fact that $\log(1 + x) \approx x$ for small x extensively.

$$\begin{aligned} 1 - H_q(\delta) &= 1 - \delta \log_q(q - 1) + \delta \log_q \delta + (1 - \delta) \log_q(1 - \delta) \\ &= 1 - \log_q(q - 1) + (1 - \delta) \log_q[(q - 1)(1 - \delta)] + \delta \log_q \delta \\ &= \frac{\log\left(\frac{q}{q-1}\right)}{\log q} + \frac{\log[(q - 1)/(d + 1)]}{(d + 1) \log q} - \frac{d}{d + 1} \frac{\log(1 + 1/d)}{\log q} \\ &\approx \frac{1}{(q - 1) \log q} + \frac{\log[\Theta(1)]}{(d + 1) \log q} - \frac{1}{(d + 1) \log q} \\ &= \Theta\left(\frac{1}{d \log q}\right) \end{aligned}$$

Overall, $t = \Theta(d^2 \log N)$. The constant inside is very small, something like 4 or so.

Open Problem 3.1. The Porat-Rothschild's construction is explicit but not strongly explicit. It is an important open problem to come up with a strongly explicit construction of d-disjunct matrices with $\Theta(d^2 \log N)$ rows. (Of course, we only consider cases when $d < \sqrt{2N}$, thanks to Bassalygo's bound.)

References

[1] V. CHVÁTAL, *A greedy heuristic for the set-covering problem*, Math. Oper. Res., 4 (1979), pp. 233–235.

[2] F. K. HWANG AND V. T. SÓS, *Nonadaptive hypergeometric group testing*, Studia Sci. Math. Hungar., 22 (1987), pp. 257–263.

[3] W. H. KAUTZ AND R. C. SINGLETON, *Nonrandom binary superimposed codes*, IEEE Trans. Inf. Theory, 10 (1964), pp. 363–377.

[4] L. LOVÁSZ, *On the ratio of optimal integral and fractional covers*, Discrete Math., 13 (1975), pp. 383–390.

[5] A. Q. NGUYEN AND T. ZEISEL, *Bounds on constant weight binary superimposed codes*, Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform., 17 (1988), pp. 223–230.

[6] E. PORAT AND A. ROTHSCHILD, *Explicit non-adaptive combinatorial group testing schemes*, in Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP), 2008, pp. 748–759.

[7] V. V. VAZIRANI, *Approximation algorithms*, Springer-Verlag, Berlin, 2001.

[8] V. A. ZINOVIEV, *Cascade equal-weight codes and maximal packings*, Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform., 12 (1983), pp. 3–10.