Available at
www.ElsevierMathematics.com
POWERED BY SCIENCE DIRECT®
Discrete Applied Mathematics 137 (2004) 97–107
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Trivial two-stage group testing for complexes using almost disjunct matrices

Anthony J. Macula[a,*,1], Vyacheslav V. Rykov[a], Sergey Yekhanin[b]

[a] *Department of Mathematics, SUNY Geneseo, Geneseo, NY 14454, USA*
[b] *Department of Cybernetics and Computer Science, Moscow State University, Moscow, 119899 Russia*

Received 29 November 2001; received in revised form 1 June 2002; accepted 15 October 2002

Abstract

Let $[t]$ represent a finite population with t elements. Suppose we have an unknown d-family of k-subsets Γ of $[t]$. We refer to Γ as the set of *positive k-complexes*. In the *group testing for complexes problem*, Γ must be identified by performing 0, 1 tests on subsets or *pools* of $[t]$. A pool is said to be positive if it completely contains a complex; otherwise the pool is said to be negative. In classical group testing, each member of Γ is a singleton. In this paper, we exhibit and analyze a probabilistic trivial two-stage algorithm that identifies the positive complexes.

MSC: 05B20; 05D05; 62K99

Keywords: Group testing; Complexes; Disjunct martix; Two-stage algorithm; Hypergraphs

1. Group testing for complexes

The screening of data sets is essential to modern technology. Whenever the objective is to find "positive objects" in a data set, a test indicating whether at least one positive is in a specific part of the data set can greatly facilitate their isolation. Such tests are called *binary group tests* and the general mathematical method behind the identification of the positives using such tests is known as *classical group testing*. See [3]. The use of classical group testing to isolate objects that are *individually* positive has become

* Corresponding author.
E-mail address: macula@geneseo.edu (A.J. Macula).
[1] The author was supported in part by Information Assurance Division of the Air Force Research Laboratory, Rome NY under contract F30602-01-M-V022 supervised by Dr. L. Popyack and by NSF grant 0107179.

0166-218X/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0166-218X(03)00191-4

standard experimental procedure. See [1], [2] and [5]. However, very little work has been done in applying group testing techniques to the identification of objects that are collectively positive. This paper is an extension of the ideas in [6]. See remarks in Section 6.

Throughout the remainder of this paper, all simple lower case variables are assumed to be non-negative integers unless otherwise stated. Given set S, $|S|$ denotes its cardinality. $[t]$ denotes the positive integers $\{1,2,\ldots,t\}$. A subset of $[t]$ with cardinality k is called a *k-set*. $\binom{[t]}{k}$ denotes the k-sets of $[t]$. Let $[t]$ represent a finite population with t elements. Suppose we have an unknown collection of k-sets $\Gamma=\{S_1,\ldots,S_d\}$ of $[t]$. We refer to Γ as the set of *positive k-complexes* and we simply call a subset in Γ a *k-complex*. In the *group testing for complexes* (GTC) problem, Γ (or a portion of Γ) must be identified by performing certain 0,1 tests on subsets or *pools*[2] of $[t]$. A pool is said to be positive if it completely contains a complex; otherwise the pool is said to be negative. In short, if $\Gamma=\{S_1,\ldots,S_d\}$, then a pool $P\subset[t]$ is positive if and only if there is an $S_i\subset P$ for some i with $1\leqslant i\leqslant d$. In classical group testing, each member of Γ is assumed to be a singleton. A GTC *pooling design* on $[t]$, $\{P_i\}_{i\in[n]}$, is simply a collection of pools of the population assayed to identify some or all of the complexes. We use the *incidence matrix representation* of a GTC pooling design. That is, given a binary $n\times t$ matrix M, identify an element u of $[t]$ with the uth column of M. Then the ith pool, P_i, in this design is given by the ith row of M. P_i is the set of all columns of M that have a 1 in the ith row.

2. Random modifications of matrices

Definition 1. Let $0<p<1$ be a real number. Let r_i be a random row vector of length t, each entry of which is 1 with probability p. Given an $n\times t$ 0,1 matrix Ω, we define $\Omega(m,p,t)$ to be the $(m+n)\times t$ matrix that results from adding m random rows r_i with $1\leqslant i\leqslant m$ to Ω. We let ω_j with $1\leqslant j\leqslant n$ be the jth row vector of Ω. We let $u_1(i),\ldots,u_v(i),\ldots,u_t(i)$ where $1\leqslant j\leqslant n$ and $u_1(i),\ldots,u_v(i),\ldots,u_t(i)$ with $1\leqslant i\leqslant n+m$ denote the column vectors of Ω and $\Omega(m,p,t)$, respectively. The meaning of $u_v(i)$ will be clear from the context.

Definition 2. Given an $n\times t$ 0,1 matrix Ω, we define the $mn\times t$ 0, 1 matrix $\Omega^*(m,p,t)$ whose rows are the coordinate-wise intersections $r_i\wedge\omega_j$ of the rows r_i and ω_j in $\Omega(m,p,t)$ with $1\leqslant i\leqslant m$ and $1\leqslant j\leqslant n$. We order the rows $r_i\wedge\omega_j$ lexicographically. We let $u_1(i,j),\ldots,u_v(i,j),\ldots,u_t(i,j)$ denote the column vectors of $\Omega^*(m,p,t)$. See Fig. 1.

In this paper we focus exclusively on binary matrices Ω that are complements (interchange 0s and 1s) of what we call *almost disjunct* matrices.

Definition 3. Let A be $n\times t$ 0,1 matrix and let $\{a_v(i)\}$, where $1\leqslant i\leqslant n$ and $1\leqslant v\leqslant t$, be the column vectors of A. Let E be the event that an r-set of columns $\{a_{v_s}(i)\}_{s=1}^{r}$ has

[2] These subsets are also called "groups". Hence the name "group testing". Since this technique has been widely applied in biotechnical screenings, the term "pool" has become very common.

Ω	$u_1(i)$	$u_2(i)$	$u_3(i)$	$u_4(i)$
ω_1	1	1	1	0
ω_2	1	1	0	1
ω_3	1	0	1	1
ω_4	0	1	1	1

$\Omega(3,0.6,4)$	$u_1(i)$	$u_2(i)$	$u_3(i)$	$u_4(i)$
ω_1	1	1	1	0
ω_2	1	1	0	1
ω_3	1	0	1	1
ω_4	0	1	1	1
r_1	1	1	1	0
r_2	0	1	0	1
r_3	0	1	1	1

$\Omega^*(3,0.6,4)$	$u_1(i,j)$	$u_2(i,j)$	$u_3(i,j)$	$u_4(i,j)$
$r_1 \wedge \omega_1$	1	1	1	0
$r_1 \wedge \omega_2$	1	1	0	0
$r_1 \wedge \omega_3$	1	0	1	0
$r_1 \wedge \omega_4$	0	1	1	0
$r_2 \wedge \omega_1$	0	1	0	0
$r_2 \wedge \omega_2$	0	1	0	1
$r_2 \wedge \omega_3$	0	0	0	1
$r_2 \wedge \omega_4$	0	1	0	1
$r_3 \wedge \omega_1$	0	1	1	0
$r_3 \wedge \omega_2$	0	1	0	1
$r_3 \wedge \omega_3$	0	0	1	1
$r_3 \wedge \omega_4$	0	1	1	1

Fig. 1.

$a_v(i) \leqslant \bigvee_{s=1}^{r} a_{v_s}(i)$ with $a_v(i) \notin \{a_{v_s}(i)\}_{s=1}^{r}$. Let $0 < \alpha \leqslant 1$ be a real number. Given the uniform distribution on the r-sets of columns of A, we say that A is α-*almost r-disjunct* if $\mathrm{Prob}(E) \leqslant 1 - \alpha$.

In other words, if A is α-almost r-disjunct, then for a randomly selected r-set of columns $S = \{a_{v_s}(i)\}_{s=1}^{r}$, the probability that the only columns that are below the sup of S are those in S is at least α. A matrix is r-disjunct [3] (or r-superimposed [4]) if and only if it is 1-almost r-disjunct.

Suppose that Ω is the complement of an α-almost r-disjunct matrix. Suppose $u_v(i) \notin \{u_{v_s}(i)\}_{s=1}^{r}$. Then $1 - \alpha$ is an upper bound on the probability that $u_v(i) \geqslant \bigwedge_{s=1}^{r} u_{v_s}(i)$. In other words, if Ω is the complement of an α-almost r-disjunct matrix, then the set of column vectors above the inf of a randomly selected r-set of column vectors $C = \{u_{v_s}(i)\}_{s=1}^{r}$ is *exactly the set C itself* with probability at least α. Henceforth, we assume that Ω is the complement of an α-almost r-disjunct matrix.

Henceforth, we assume that Ω is an α-almost r-disjunct matrix. Note that for our applications in group testing for k-complexes, we must assume that $k = r$. This is, *we use an α-almost k-disjunct matrix in our group testing for k-complexes algorithm*.

Since the columns of Ω, $\Omega(m,p,t)$ and $\Omega^*(m,p,t)$ are in an obvious correspondence, then given a d-family of k-sets $S_1,\ldots,S_\ell,\ldots,S_d$ of columns of Ω, we have the

corresponding d-families of sets of columns in $\Omega(m, p, t)$ and $\Omega^*(m, p, t)$, respectively and vice versa. We use the single notation $S_1, \ldots, S_\ell, \ldots, S_d$ to denote a family of k-sets in Ω, $\Omega(m, p, t)$ and/or $\Omega^*(m, p, t)$. The meaning of S_ℓ will be clear from the context.

Also, we shall assume that a family of k-sets $\Gamma = \{S_1, \ldots, S_\ell, \ldots, S_d\}$ has been generated by selecting each member S_ℓ from the uniform distribution for the k-sets without replacement. Thus for $1 \leqslant \ell \leqslant d$, $\ell \neq \ell_0$, the $d-1$ random variables $X_{\ell,\ell_0} = |S_\ell \setminus S_{\ell_0}|$ are independent and identically distributed with distribution function

$$f(y) = \binom{t-k}{y}\binom{k}{k-y}\binom{t}{k}^{-1} \quad \text{where } 0 \leqslant y \leqslant k. \tag{1}$$

Definition 4. Let $S_1, \ldots, S_\ell, \ldots, S_d$ be k-sets of columns of Ω. We say that the *random part of* $\Omega(m, p, t)$ *separates* S_{ℓ_0} from $S_1, \ldots, S_\ell, \ldots, S_d$ with $\ell_0 \neq \ell$ if there is a row r_i with $1 \leqslant i \leqslant m$ of $\Omega(m, p, t)$ such that every column of S_{ℓ_0} in $\Omega(m, p, t)$ has a 1 in row r_i and for each S_ℓ with $\ell_0 \neq \ell$ there is a column of S_ℓ in $\Omega(m, p, t)$ with a 0 in row r_i. In other words, the row vector r_i *covers* S_{ℓ_0} in $\Omega(m, p, t)$ and does not cover S_ℓ in $\Omega(m, p, t)$ with $\ell_1 \neq \ell$.

Note that for our applications, we assume that S_{ℓ_0} is one of the sets $S_1, \ldots, S_\ell, \ldots, S_d$.

3. Pooling and decoding

We identify a population of cardinality $[t]$ with the columns $u_1(i, j), \ldots, u_v(i, j), \ldots, u_t(i, j)$ of $\Omega^*(m, p, t)$. Suppose there are d positive k-complexes $\Gamma = \{S_1, \ldots, S_\ell, \ldots, S_d\}$ (selected as indicated in the end of Section 2) with $S_\ell = \{u_{v_s}(i, j)\}_{s=1}^k$. We use the matrix $\Omega^*(m, p, t)$ to construct our pools. A row $r_i \wedge \omega_j$ of $\Omega^*(m, p, t)$ determines a pool of the population in the obvious way. That is, an element $u_v(i, j)$ is in the pool determined by $r_i \wedge \omega_j$ if and only if there is a 1 in the entry where $r_i \wedge \omega_j$ and the column $u_v(i, j)$ intersect. This happens exactly when both r_i and ω_j cover $u_v(i)$ in $\Omega(m, p, t)$. By testing each pool $r_i \wedge \omega_j$, we define a binary mn-vector $o(i, j)$ called the *output vector* by setting the (i, j)th entry in the lexicographical order equal to 1 if the test result of pool $r_i \wedge \omega_j$ is positive and 0 if negative. So pool $r_i \wedge \omega_j$ is positive if and only if there is some $S_\ell = \{u_{v_s}(i)\}_{s=1}^k$ where both r_i and ω_j cover S_ℓ in $\Omega(m, p, t)$.

The algorithm is quite simple. Since there is a trivial confirmatory phase, our algorithm is a *trivial two-stage algorithm*. Suppose we have an output vector $o(i, j)$ generated by an application of the pooling strategy outlined above. Let $o_i(j)$ denote the subvector of $o(i, j)$ where i is fixed.

Algorithm 1. For each i with $1 \leqslant i \leqslant m$, consider the set of columns of $\Omega^*(m, p, t)$, $C_i = \{u_v(i, j)\colon u_v(j) \geqslant o_i(j)$ where i is fixed and $u_v(j)$ a column in $\Omega\}$. Then for each C_i with $|C_i| = k$, test the set C_i as a small pool. If pool C_i is positive, then since it is a k-set, it must be a k-complex.

Example 1. In Fig. 2, we use a small example to demonstrate the pooling and decoding scheme. Ω is the complement of 4×4 identity matrix which is 2-disjunct (indeed,

Population = $\{u_1(i,j), u_2(i,j), u_3(i,j), u_4(i,j)\}$, $\Gamma = \{S_1, S_2\}$ where
$S_1 = \{u_2(i,j), u_3(i,j)\}$ $S_2 = \{u_3(i,j), u_4(i,j)\}$

Ω	$u_1(i)$	$u_1(i)$	$u_1(i)$	$u_1(i)$
ω_1	1	1	1	0
ω_2	1	1	0	1
ω_3	1	0	1	1
ω_4	0	1	1	1

$\Omega(3, 0.6, 4)$	$u_1(i)$	$u_1(i)$	$u_1(i)$	$u_1(i)$
ω_1	1	1	1	0
ω_2	1	1	0	1
ω_3	1	0	1	1
ω_4	0	1	1	1
r_1	1	1	1	0
r_2	1	1	0	1
r_3	0	1	1	1

$\Omega^*(3, 0.6, 4)$	$u_1(i,j)$	$u_2(i,j)$	$u_3(i,j)$	$u_4(i,j)$	pool outcome	$o(i,j)$
$r_1 \wedge \omega_1$	1	1	1	0	+	1
$r_1 \wedge \omega_2$	1	1	0	0	−	0
$r_1 \wedge \omega_3$	1	0	1	0	−	0
$r_1 \wedge \omega_4$	0	1	1	0	+	1
$r_2 \wedge \omega_1$	1	1	0	0	−	0
$r_2 \wedge \omega_2$	1	1	0	1	−	0
$r_2 \wedge \omega_3$	1	0	0	1	−	0
$r_2 \wedge \omega_4$	0	1	0	1	−	0
$r_3 \wedge \omega_1$	0	1	1	0	+	1
$r_3 \wedge \omega_2$	0	1	0	1	−	0
$r_3 \wedge \omega_3$	0	0	1	1	+	1
$r_3 \wedge \omega_4$	0	1	1	1	+	1

Fig. 2.

4-disjunct). The population has four objects and there are two positive 2-complexes. Since only C_1 has $|C_1| = 2$, then only C_1 is tested and would be discovered to be a 2-complex. Here 13 tests (12 first stage, one confirmatory stage) are used to identify one of two positive 2-complexes.

$$C_1 = \left\{ u_v(i,j) \colon u_v(j) \geqslant o_1(j) = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\} = \{u_2(i,j), u_3(i,j)\},$$

$$C_2 = \left\{ u_v(i,j) \colon u_v(j) \geqslant o_2(j) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\} = \{u_1(i,j), u_2(i,j), u_3(i,j), u_4(i,j)\},$$

$$C_3 = \left\{ u_v(i,j)\colon\ u_v(j) \geqslant o_3(j) = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \right\} = \{u_3(i,j)\}.$$

4. Analysis of the algorithm

To analyze Algorithm 1, we need to understand the conditions that make C_i a k-complex. Suppose $S_\ell = \{u_{v_s}(i,j)\}_{s=1}^{k}$ is a k-complex. Consider the corresponding subset of columns in Ω. Let $\wedge S_\ell = \bigwedge_{s=1}^{k} u_{v_s}(j)$ in Ω with $1 \leqslant j \leqslant n$. Clearly $S_\ell \subset C_i$ with $1 \leqslant i \leqslant m$ if and only if $\wedge S_\ell \geqslant o_i(j)$. Now *if* $\wedge S_\ell = o_i(j)$, then $S_\ell = C_i$ with probability α because Ω is the complement of an α-almost k-disjunct matrix.

Proposition 1. *Let $1 \leqslant i \leqslant m$. If there is a row r_i of $\Omega(m,p,t)$ that separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$, then $\wedge S_{\ell_0} = o_i(j)$ in Ω.*

Proof. Fix i and suppose r_i separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$. Suppose $1 \leqslant j \leqslant n$. Pool $r_i \wedge \omega_j$ is positive and thus $o_i(j)=1$ if and only if both rows r_i and ω_j cover *some* S_ℓ in $\Omega(m,p,t)$. Since r_i only covers S_{ℓ_0}, it follows that $o_i(j)=1$ if and only if ω_j covers S_{ℓ_0} in $\Omega(m,p,t)$. On the other hand, $\wedge S_{\ell_0}(j)=1$ if and only if ω_j covers S_{ℓ_0} in Ω. Thus $\wedge S_{\ell_0}(i) = o_i(j)$ in Ω. □

Corollary 1. *Let C_i be defined as in Algorithm* 1. *If row r_i of $\Omega(m,p,t)$ separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$, then $S_\ell = C_i$ with probability α.*

Proof. From the discussion in first paragraph of this section, if $\wedge S_\ell = o_i(j)$, then $S_\ell = C_i$ with probability α. Hence, the result follows immediately from Proposition 1. □

Lemma 1. *Let $\Gamma = \{S_1,\ldots,S_\ell,\ldots,S_d\}$ be a family of k-complexes of columns of $\Omega^*(m,p,t)$. Let $S_{\ell_0} \in \Gamma$. We define $\Phi(\ell_0,d,p)$ to be the probability that the random part of $\Omega(m,p,t)$ separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$. Then*

$$\Phi(\ell_0,d,p) \geqslant 1 - \left(1 - p^k \left(\sum_{y=1}^{k} \binom{t-k}{y}\binom{k}{k-y}\binom{t}{k}^{-1}(1-p^y)\right)^{d-1}\right)^m.$$

Proof. Suppose $1 \leqslant i \leqslant m$. Without loss of generality, assume $\ell_0 = 1$. For $2 \leqslant \ell \leqslant d$, let $T_\ell = S_\ell \setminus S_1$. Fix i and let E_ℓ be the event that r_i does not cover T_ℓ in $\Omega(m,p,t)$. We show that for $3 \leqslant \ell \leqslant d$, $\mathrm{Prob}(E_\ell | \bigcap_{v=2}^{\ell-1} E_v) \geqslant \mathrm{Prob}(E_\ell)$. First, if T_ℓ is entirely 1s in row r_i, then in row r_i, the probability that there is a 0 in each set T_v with $2 \leqslant v \leqslant \ell-1$ is not increased. Hence, $\mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v | \neg E_\ell) \leqslant \mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v)$.

Then, since

$$\mathrm{Prob}\left(\neg E_\ell \,\middle|\, \bigcap_{v=2}^{\ell-1} E_v\right) = \frac{\mathrm{Prob}\left(\neg E_\ell \cap \bigcap_{v=2}^{\ell-1} E_v\right)}{\mathrm{Prob}\left(\bigcap_{v=2}^{\ell-1} E_v\right)} = \frac{\mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v | \neg E_\ell)\mathrm{Prob}(\neg E_\ell)}{\mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v)}$$

$$= \frac{\mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v | \neg E_\ell)\, p^{|T_\ell|}}{\mathrm{Prob}(\bigcap_{v=2}^{\ell-1} E_v)},$$

we have that $\mathrm{Prob}(\neg E_\ell | \bigcap_{v=2}^{\ell-1} E_v) \leqslant p^{|T_\ell|}$. Thus $\mathrm{Prob}(E_\ell | \bigcap_{v=2}^{\ell-1} E_v) \geqslant 1 - p^{|T_\ell|} = \mathrm{Prob}(E_\ell)$. From here it follows that the probability that r_i covers S_1 and does not cover S_ℓ for $2 \leqslant \ell \leqslant d$ is at least $p^k \prod_{\ell=2}^{d}(1-p^{|T_\ell|})$ because $S_1 \cap T_\ell \neq \emptyset$. Letting y represent the number of columns in T_ℓ, it follows from the independence of the family $\{X_{\ell,1}\}_{\ell=2}^{d}$ (see (1)) that

$$\Phi(\ell_0, d, p) \geqslant 1 - \left(1 - p^k \left(\sum_{y=1}^{k} \binom{t-k}{y}\binom{k}{k-y} \times \binom{t}{k}^{-1}(1-p^y)\right)^{d-1}\right)^m. \qquad \square$$

Theorem 1. *Suppose $\Gamma = \{S_1, \ldots, S_2 \ldots, S_d\}$ is the family of k-complexes in the set of columns of $\Omega^*(m, p, t)$. If Ω is the complement of α-almost k-disjunct matrix, then by testing mn first stage pools and performing at most m confirmatory tests, the expected number of positive complexes identified by our algorithm is at least $\alpha \cdot d \cdot \Phi(\ell_0, d, p)$.*

Proof. This follows from Corollary 1, Lemma 1 and the additivity of expectation. $\square$

5. A class of α-almost k-disjunct matrices

A maximal distance separable (MDS) code is a q-ary code with $t = q^r$ codewords of length N such that the Hamming distance d between any two codewords is $d = N - r + 1$. For any prime power q and r with $2 \leqslant r \leqslant q+1$ there are q-ary linear MDS Reed–Solomon (RS) codes with parameters $t = q^r$, $N = q+1$ and $d = q - r + 2$. In [4], the ideas in [5] coupled with generalized and shortened RS codes are used to construct $n \times t$ s-disjunct matrices with $n = q[s(\lceil m/\log_2 q \rceil - 1) + 1]$, $2^m \leqslant t \leqslant 2^{m+1}$ and $s(\lceil m/\log_2 q \rceil - 1) \leqslant q$. A table of codes optimizing these parameters appears in [4].

However, from a practical standpoint these codes are much stronger. That is, these s-disjunct matrices are also α-almost k-disjunct matrices where $\alpha \approx 1$ and $k > s$. In general, let B be a linear MDS q-ary code with $t = q^r$ codewords of length N. We identify B with the $N \times t$ matrix whose columns $\{b_v(i)\}_{v=0}^{t-1}$ where $1 \leqslant i \leqslant N$ are the codewords of B. Then B is concatenated into a binary $qN \times t$ matrix B' by replacing

each q-ary symbol with the weight 1 binary column q-vector e_j^q with $0 \leqslant j \leqslant q-1$. For example,

$$e_1^3 = \begin{matrix} 0 \\ 1 \\ 0 \end{matrix}.$$

We call B' the *trivial concatenation* of B. We let the trivial concatenation of column $b_v(i)$ be denoted by $b'_v(i,j)$ where $1 \leqslant i \leqslant N$, $0 \leqslant j \leqslant q-1$ and (i,j)s are in the lexicographic order. Then $b'_v(i,j) = 1$ if and only if $j = b_v(i)$. The columns of B' are $\{b'_v(i,j)\}_{v=0}^{t-1}$. Let $S = \{b_{v_\ell}(i)\}_{\ell=1}^{k}$ be a k-set of column in B and let $S' = \{b'_{v_\ell}(i,j)\}_{\ell=1}^{k}$ be the corresponding k-set of concatenation columns in B'. Suppose $b_v(i) \notin S$, then $b'_v(i,j) \notin S'$ and $b'_v(i,j) \leqslant \vee S'$ if and only if, for each i with $1 \leqslant i \leqslant N$ there is an ℓ with $1 \leqslant \ell \leqslant k$ such that $b_v(i) = b_{v_\ell}(i)$ in B. Suppose $d = q - r + 2$. Let $C_v(q,d,N,k)$ be the number of k-sets of columns of B *without* the later property. That is $C_v(q,d,N,k)$ is the number of k-sets $S = \{b_{v_\ell}(i)\}_{\ell=1}^{k}$ that have a $b_v(i) \notin S$ for which there is an i for $1 \leqslant i \leqslant N$, with $b_v(i) \neq b_{v_\ell}(i)$ for each ℓ with $1 \leqslant \ell \leqslant k$. Let $b_0(i)$ be the constant 0 vector in B. Since B is a linear code, it follows that $C_0(q,d,N,k) = C_v(q,d,N,k)$. Then the number of k-sets S' of columns in B' for which there is $b'_v(i,j) \notin S'$ with $b'_v(i,j) \leqslant \vee S'$ is at most

$$q^r \left(\binom{q^r - 1}{k} - C_0(q,d,N,k) \right). \tag{2}$$

We now compute $C_0(q,d,N,k)$. Let $A_\omega(N)$ be the number of codewords of weight ω in an MDS q-ary code of length N and distant d. From [7], we have that

$$A_\omega(q,d,N) = \binom{N}{\omega} (q-1) \sum_{j=0}^{\omega-d} (-1)^j \binom{\omega - 1}{j} q^{\omega-d-j}.$$

Let $D(q,d,N,k,h)$, where $1 \leqslant h \leqslant N$, be the number of k-sets $S = \{b_{v_\ell}(i)\}_{\ell=1}^{k}$ for which there are h indices i with $0 \neq b_{v_\ell}(i)$ for all ℓ with $1 \leqslant \ell \leqslant k$. We have that

$$D(q,d,N,k,h) = \begin{cases} \binom{q^{r-h}(q-1)^h}{k} & \text{if } h \leqslant r, \\ \binom{A_h(q,h-r+1,h)}{k} & \text{if } h > r. \end{cases}$$

This is because:

1. Any h positions in an MDS code can be regarded as information positions when $h \leqslant r$. Thus in these h positions, every q-ary vector of length h is repeated q^{r-h} times.

And,

2. The code that is achieved by restricting the MDS code B to h positions when $h > r$ is also an MDS code of length h and volume q^r.

Now by inclusion–exclusion we have that

$$C_0(q,d,N,k) = \sum_{h=1}^{N} (-1)^{h+1} \binom{N}{h} D(q,d,N,k,h).$$

Proposition 2. *Let B' be the trivial concatenation of a linear MDS q-ary code B with $t = q^r$ codewords of length N. Let*

$$\alpha = 1 - q^r \left(\binom{q^r - 1}{k} - C_0(q,d,N,k) \right) \binom{q^r}{k}^{-1}.$$

Then B' is α-almost k-disjunct.

Proof. This follows (2). □

Example 2. For prime power q and integer r with $2 \leqslant r \leqslant q+1$, we have the linear MDS RS code $\mathrm{RS}(q,r)$ with parameters $t = q^r$, $N = q+1$ and $d = q - r + 2$. Take $B = \mathrm{RS}(7,3)$. So B is MDS with $t = 343$, $N = 8$, and $d = 5$. From Proposition 2 it follows that B' is a 56×343 0.918-almost 4-disjunct matrix. Let Ω be the complement of B' and consider $\Omega(25, 0.67, 343)$. If there are five positive 4-complexes in [343], then Theorem 1 tells us that 1400 first stage tests and at most 25 confirmatory tests will identify and expected number of at least 4.01 of the five positive 4-complexes.

6. Remarks

The methods and constructions in this paper have some things in common with those in [6], but there are also significant differences. The main similarity is that in both cases, pairwise intersections of rows of binary matrices are used to construct the pools in the complex pooling design. However, in [6] a completely random matrix was used, while in this paper, a randomly augmented non-random α-almost k-disjunct matrix (a notion introduced in this paper) is used to construct the complex pooling design. The deterministic nature of the non-random matrix used here gives a two-stage algorithm with a less complex decoding procedure *and produces fewer candidate positive complexes at the end of the first stage* (at most one candidate for each $o_i(j)$). At the end of the first stage of the algorithm in [6], all k-infs of columns of a random Ω must be compared to each $o_i(j)$ while in Algorithm 1 here only the columns of the non-random Ω need to be compared to $o_i(j)$. Then in [6] all k-sets of columns of Ω with infs equal to some $o_i(j)$ are tested. Since many distinct k-sets of columns of a random matrix can have the same k-inf, the number of candidate positives tends to be larger.

A key feature in any group testing for complexes algorithm is how well the pooling design "separates" the positive complexes. Our Lemma 1 is the best known lower bound on the probability that a random design (i.e., the random part of $\Omega(m,p,t)$) "separates" randomly distributed positive complexes. It is a considerable improvement over Proposition 1 in [6]. Moreover, for situations where the k-complexes are not randomly distributed, the proof of Lemma 1 can easily be modified to yield a more general result and/or several corollaries. Here is a more general result.

Lemma 2. *Let $\Gamma = \{S_1,\ldots,S_\ell,\ldots,S_d\}$ be a family of k-complexes of columns of $\Omega^*(m,p,t)$ such that for each ℓ with $1 \leqslant \ell \leqslant d$, $\ell \neq \ell_0$, the $d-1$ random variables $X_{\ell,\ell_0} = |S_\ell \setminus S_{\ell_0}|$ are independent with density function $f_{\ell,\ell_0}(y)$ where $1 \leqslant y \leqslant k$. We define $\Phi'(\ell_0,d,p)$ to be the probability that the random part of $\Omega(m,p,t)$ separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$. Then*

$$\Phi'(\ell_0,\Gamma,p) \geqslant 1 - \left(1 - p^k \prod_{\ell=1,\ell\neq\ell_0}^{d} E(1-p^{X_{\ell,\ell_0}})\right)^m.$$

One corollary is.

Corollary 1. *Let $\Gamma = \{S_1,\ldots,S_\ell,\ldots,S_d\}$ be a family of k-complexes of columns of $\Omega^*(m,p,t)$ such that $|S_\ell \setminus S_{\ell_0}| \geqslant r$ with $\ell \neq \ell_0$. Let $\Phi''(\ell_0,d,p)$ to be the probability that the random part of $\Omega(m,p,t)$ separates S_{ℓ_0} from $S_1,\ldots,S_\ell,\ldots,S_d$ with $\ell_0 \neq \ell$ in $\Omega(m,p,t)$. Then*

$$\Phi''(\ell_0,d,p) \geqslant 1 - (1 - p^k(1-p^r)^{d-1})^m.$$

In a manner analogous to what is discussed in the previous sections, these results can be used to analyze the performance of our trivial-two stage method when the k-complexes are not randomly distributed.

Finally, the performance of our method discussed in this paper can be analyzed to take testing errors into account in a manner similar to that in [6]. This issue will be the topic of a forthcoming paper.

References

[1] D.J. Balding, et al., A comparative survey of non-adaptive pooling designs, Genetic Mapping and DNA Sequencing, 133–155, IMA Volumes in Mathematics and its Applications, Springer, Berlin, 1995.

[2] W.J. Bruno, et al., Design of efficient pooling experiments, Genomics 26 (1995) 21–30.

[3] D.-Z. Du, F. Hwang, Combinatorial Group Testing and Its Applications, 2nd Edition, World Scientific, Singapore, 2000.

[4] A. Dyachkov, A. Macula, V. Rykov, New construction of superimposed codes, IEEE Trans. Inform. Theory 46 (1) (2000) 284–291.

[5] Farach, et al., Group testing problems with sequences experimental molecular biology, in: B. Carpentieri, et al. (Eds.), Proceedings of Compression and Complexity of Sequences, IEEE Press, New York, 1997, pp. 357–367.

[6] A. Macula, P. Vilenkin, D. Torney, Two stage group testing for complexes in the presence of errors, DIMACS Proceeding of Medical Applications of Discrete Mathematics, 2000, DIMACS Series in Discrete Math and Theoretical Computer Science, Vol. 55, American Mathematical Society, Rutgers University, New Brunswick, NJ, 2000, pp. 145–157.
[7] F.J. MacWilliams, N.J.A. Sloane, The Theory of Error-Correcting Codes, North-Holland, Amsterdam, 1977.