2
d-Separable and d-Disjunct Matrices

Every nonadaptive group testing can be represented by a 0-1 matrix with certain properties. In this chapter, we introduce some concepts about those matrices.

2.1 d-Separable Matrix

For each nonadaptive group testing, one can construct a 0-1 matrix as follows: Use each row to represent a test and each column to represent an item. The cell (i, j) contains 1-entry if and only if the ith test contains the jth item. Since each column corresponds to a subset of rows, one may talk about a *union* of several columns, which is equivalent to Boolean sum of those 0-1 column vectors. Clearly, if such a matrix is constructed from a nonadaptive group testing which can identify d positive items, then it must have a property that *all unions of* d *distinct columns are distinct.* A 0-1 matrix satisfying such a property is called a *d-separable* matrix.

The definition of a 1-separable matrix is reduced to "no two columns are the same." This was also called a *separating system* studied by Rényi [10] and Katona [6].

Kautz and Singleton [7] proved the following lemma.

Lemma 2.1.1 *If a matrix is d-separable, then it is k-separable for every* $1 \leq k \leq d < n$.

Proof. Suppose that M is d-separable but not k-separable for some $1 \leq k < d < n$. Namely, there exist two distinct samples s and s' each consisting of k columns such that $P(s) = P(s')$ where $P(s)$ denotes the set of tests with positive outcomes under the sample s. Let C_x be a column in neither s nor s'. Then

$$C_x \cup P(s) = C_x \cup P(s') \ .$$

Adding a total of $d - k$ such columns C_x to both s and s' yields two distinct samples s_d and s'_d each consisting of d columns such that $P(s_d) = P(s'_d)$. Hence M is not d-separable. If there are only $\ell < d - k$ such columns C_x, then select $d - k - \ell$ pairs of columns (C_y, C_z) such that C_y is in s but not in s' and C_z is in s' but not in s. Then

$$C_z \cup P(s) = P(s) = P(s') = C_y \cup P(s') \ .$$

Therefore these pairs can substitute for the missing C_x. Since M is d-separable, a total of $d-k$ C_x and (C_y, C_z) can always be found to yield two distinct s_d and s'_d each consisting of d columns. □

Let $S(d, n)$ be the sample space consisting of all samples with n items and d positives. Consider an $t \times n$ d-separable matrix. For each sample $s \in S(d, n)$, we can obtain t test outcomes which form a t-dimensional column vector, called *test outcome vector*. Given a test outcome vector, how to decode it to recover the sample s, i.e., how to identify all positives?

First, we note that all items contained by a negative test can be identified to be negative immediately. The remainder is find d items among remaining items to hit all positive tests. This is exactly a hitting set problem as follows:

> HITTING SET PROBLEM: Given a set X and a collection $\mathcal{C}$ of subsets of X, find a minimum-cardinality subset Y of X such that $Y \cap C \neq \emptyset$.

In fact, in current situation, X is the set of items not appearing in any negative pool, $\mathcal{C}$ is the collection of all positive pools and the next lemma shows that if $|X| > d$, then the d positive items form the unique minimum solution.

Lemma 2.1.2 *Given a $t \times n$ d-separable matrix and a test outcome vector resulting from a sample in $S(d, n)$, either the number of items not appearing in any negative pool is exactly d, or the size of optimal solution for the hitting set problem reduced as above is d and it is unique.*

Proof. Let X be the set of item not appearing in any negative pool. Since X contains all positive items, we have $|X| \geq d$. If $|X| > d$, then we first show that any optimal solution for the hitting set problem has size exactly d. Note that all d positive items form a hitting set. Any optimal solution has size at least d. For contradiction, suppose there exists a hitting set H of size $h < d$. Then putting $d-h$ more other items choosing from X to H would result in a hitting set of size d. Since $|X| > d$, we can find, in this way, two distinct hitting sets of size d. Note that the union of columns corresponding any hitting set is the test outcome vector. Therefore, the two unions corresponding two hitting sets of size d are equal, contradicting the definition of d-separability.

The uniqueness of the hitting set of size d can also result from the above argument. □

The *weight* of a binary vector is the number of 1s in the vector.

Corollary 2.1.3 *In any d-separable matrix, the union of any d columns other than zero columns has weight at least d.*

Proof. Let $s \in S(d, n)$. By Lemma 2.1.2, there are two cases.

Case 1. s is the unique solution of the hitting set problem resulting from the decoding. Then, the weight of the union $U(s)$ of the d columns in s is clearly having weight at least d since, otherwise, a hitting set of size at most $d-1$ exists.

Case 2. $|X| = d$ where X is the set of items not appearing in any negative pool. For contradiction, suppose $U(s)$ has weight $w < d$. This means that w positive pools. Thus, we can choose a subset $s' \subset s$ with $|s'| = d-1$ to hit these w pools. Consider s' as a sample in $S(d-1, n)$. Then all negative pools are still negative and all positive pools are still positive. Therefore, X is still the set of items not in any negative pool. Since $|X| = d > d-1$, s' is the unique hitting set of size $d-1$ for these positive pools by Lemma 2.1.2. It follows that each column C_j in s' has a positive pool hitted only by C_j not other ones. (If $C - j$ has not such a positive pool, then it can be removed from the hitting set.) It in turn follow that the column in $s \setminus s'$ has to be a zero column, a contradiction. □

The hitting set problem is NP-hard. Even the size of optimal hitting set is known, no clever method has been found so far to compute the optimal solution. Only way that we know is exhaused search, that is, checking all possible d-subsets of unidentified items in time $O(|X|^d)$. Is there an algorithm with a polynomial time with respect to both $|X|$ and d? The answer is NO if NP≠P. In fact, if such a polynomial-time algorithm exists, then we may run the algorithm $|X|$ times on inputs with $d = 1, 2, ..., |X|$ to solve the hitting set problem in polynomial time. This implies NP=P [8].

In the real world, knowing an upper bound of the number of positive items is more practical than knowing the exact number of positives. Therefore, one is also interested in $\bar{d}$-separability. A binary matrix is *$\bar{d}$-separable* if all unions of at most d columns are distinct. $\bar{1}$-separability is the same as 1-separability. However, for $d \geq 2$, $\bar{d}$-separability is stronger than d-separability. For example, the matrix

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

is 2-separable, but not $\bar{2}$-separable.

Similarly, to decode test outcomes from a $\bar{d}$-separable matrix under a sample, we may first identify all items contained by negative pools. Next, set X to be the set of items not appearing in any negative pool and $\mathcal{C}$ to be the collections of all positive pools. Then, all positive items form a hitting set of size at most d.

Let $S(\bar{d}, n)$ be the sample space consisting of all samples with n items and at most d positives. The following lemma indicates that all positive items actually form unique hitting set of size at most d.

Lemma 2.1.4 *For any $t \times n$ $\bar{d}$-separable matrix and any test outcome vector resulting from a sample in $S(\bar{d}, n)$, there exists exactly one hitting set of size at most d for the hitting set problem reduced as above.*

Proof. Similar to the proof of Lemma 2.1.2. □

Corollary 2.1.5 *Every $\bar{d}$-separable matrix does not contain zero column.*

Proof. For contradiction, suppose there exists a $\bar{d}$-separable $t \times n$ matrix M containing a zero column. Choose $s \in S(\bar{d}, n)$ such that $|s| \leq d - 1$ and s does not contain the zero column. Then the zero column would destroy the uniqueness of hitting set of size at most d for positive pools. □

By Corollaries 2.1.3 and 2.1.5, we have

Corollary 2.1.6 *In any $\bar{d}$-separable matrix, the union of any d columns has weight at least d.*

2.2 d-Disjunct Matrix

As described in the last section, to decode positives from testing outcomes, we need to solve a hitting set problem, that is, find a minimum-cardinality subset of items hitting every positive pool. For testing based on d-separable (or $\bar{d}$-separable) matrix, the optimal solution of the hitting set problem is often unique and has size exactly d (or at most d) unless its input size is exactly d. Indeed, its input size can be reduced by negative pools, although the hitting set problem in this case is still NP-hard. The following is a characterization on the input size of this hitting set problem.

Theorem 2.2.1 *For testing based on a d-separable matrix M. the number of items not appearing in any negative pool is always no more than $d + k - 1$ if and only if any union of k columns cannot be contained by any union of other d columns.*

Proof. Note that an item does not appearing in any negative pool if and only if its corresponding column is not contained by the union of d positive columns. Therefore, the number of items not appearing in any negative pool is more than $d + k$ if and only if there are at least k non-positive items whose columns are contained by the d positive columns. Therefore, the theorem is true. □

To decode positives efficiently from testing outcomes, we are interested in the case that $k = 1$ since in this case, the number of items not appearing in any negative pool is exactly d and hence decoding can be done in $O(n)$ time.

Consider a $t \times n$ binary matrix M where R_i and C_j denote row i and column j. M will be said to be *d-disjunct* if the union of any d columns does not contain any other column. Note that d-disjunctness implies that the union of any up to d columns does not contain any other column. Therefore, every d-disjunct matrix is d-separable.

Corollary 2.2.2 *A binary matrix is d-disjunct if and only if for any sample $s \in S(d, n)$, the union of negative pools contains all negative items.*

Clearly, a binary d-disjunct matrix is also k-disjunct for any $1 \leq k \leq d$. Therefore, the following holds.

Corollary 2.2.3 *A binary matrix is d-disjunct if and only if for any sample $s \in S(\bar{d}, n)$, the union of all negative pools contains all negative items.*

This corollary means that a d-disjunct matrix also corresponds to a nonadaptive $(\bar{d}, n)$ algorithm, but with an additional property which allows the positives to be identified easily. That is, the following is true.

Corollary 2.2.4 *d-disjunct implies $\bar{d}$-separable.*

Of course, trade off always exists. Saha and Sinha [11] showed that d-disjunctness requires at least one more test than d-separability.

Lemma 2.2.5 *Deleting any row R_i from a d-disjunct matrix M yields a d-separable matrix M_i.*

Proof. Let $s, s' \in S(\bar{d}, n)$. Let $P(s)$ denote the set of positive pools under the sample s. Then $P(s)$ and $P(s')$ must differ in at least 2 rows or one would contain the other. Hence, they are different even after the deletion of a row. □

Could this result be strengthed to $\bar{d}$-separability? The answer is NO. In fact, the proof cannot applied to the case than s contains s' or s' contains s in which case $P(s)$ and $P(s')$ can differ in only one row.

The following lemmas established a basic relation between d-disjunctness, d-separability and $\bar{d}$-separability.

Lemma 2.2.6 *A binary matrix is $\overline{(d+1)}$-separable if and only if it is $(d+1)$-separable and d-disjunct.*

Proof. Since $\overline{(d+1)}$ must be $(d+1)$-separable, we need onlt to show d-disjunctness for forward direction. Suppose that M is $\overline{(d+1)}$-separable but not d-disjunct, i.e., there exists a sample s of d columns such that $P(s)$ contains another column C_j not in s, where $P(s)$ denotes the set of tests with positive outcomes under sample s. Then

$$P(s) = C_j \cup P(s),$$

a contradiction to the assumption that M is $\overline{(d+1)}$-separable.

For backward direction, consider two samples s and s' from $S(\overline{d+1}, n)$. If $|s| = |s'| = d+1$, then $P(s) \neq P(s')$ by $(d+1)$-separability. Else, assume $|s| \leq |s'|$. Then $|s| \leq d$ and s' must contain a column C_j not in s. By d-disjunctness, $P(s)$ does not contain C_j and hence $P(s) \neq P(s')$. □

From Corollary 2.2.4 and Lemma 2.2.6 any property held by a d-separable matrix also holds for a d-disjunct matrix, and any property held for a d-disjunct matrix also holds for a $\overline{d+1}$-separable matrix.

The following summarizes the relations between these properties.

$$\begin{array}{ccccc} & & \bar{d}-\text{separable} & & \\ & & \Uparrow \quad\quad \Downarrow & & \\ \overline{(d+1)}-\text{separable} & \Rightarrow & d-\text{disjunct} & \Rightarrow^{*} & d-\text{separable} \\ & & \Downarrow & & \Downarrow \\ & & k-\text{disjunct}\ (k<d) & \Rightarrow & k-\text{separable}\ (k<d) \end{array}$$

where $*$ means "delete any row".

Chen and Hwang [2] showed the following interesting results.

Theorem 2.2.7 *If a d-separable matrix contains a zero column, then deleting the zero column results in a* $(d-1)$*-disjunct matrix.*

Proof. For contradiction, suppose there exist a nonzero column C and a set s of $d-1$ other nonzero columns such that $C \subset U(s)$ where $U(s)$ is the union of columns in s. Let s' be obtained from s by putting in the zero column. Then $U(\{C\} \cup s) = U(s')$, contradicting the d-separability. □

Corollary 2.2.8 *A binary matrix containing a zero column is d-separable if and only if all nonzero columns form a* $\bar{d}$*-separable matrix.*

Proof. By Theorem 2.2.7 and Lemma 2.2.6, the forward direction is correct. For backward direction, consider two samples s and s' in $S(d,n)$. Let s^* and s'^* be obtained respectively from s and s' by deleting zero column. Then $s^* \neq s'^*$. By $\bar{d}$-separability of submatrix with nonzero columns, $U(s) = U(s^*) \neq U(s'^*) = U(s')$ where $U(s)$ denote the union of columns in s. □

Theorem 2.2.9 *Let M be a* $2d$*-separable matrix. Then we can obtain a d-disjunct matrix by adding at most one row to M.*

. *Proof.* If M is d-disjunct, then we are done. Suppose M is not. Then there exists a column C and a set s of d other columns such that C is contained by the union $U(s)$ of the d columns in s. Add a row R which has 1-entry at C and 0-entries at all collumns in s to break up the containment $C \subset U(s)$ in the new matrix. However, there may exist many pairs (C,s) such that $C \subset U(s)$. If we use the same row to break up the containment, then we must show that there is no conflicting in the definition of R, that is, we must show that for two such pairs (C,s) and (C',s'), we would not have $C \in s$.

For contradiction, suppose $C \in s$. Set $S_0 = s \cup s' \cup \{C, C'\}$. Then $|S_0| \leq 2d+1$. Set $S_1 = S_0 \setminus \{C\}$ and $S_2 = S_0 \setminus \{C'\}$. Then $|S_1| = |S_2| \leq 2d$. However, $U(S_1) = U(S_0) = U(S_2)$, contadicting $2d$-separability. □

Corollary 2.2.10 *Let M be a $2d$-separable matrix. Then we can obtain a $\overline{(d+1)}$-separable matrix by adding at most one row to M.*

. *Proof.* By Theorem 2.2.9 and Lemma 2.2.6. □

2.3 Error-Tolerance in (d, n) Space

The *Hamming distance* of two column vectors is the number of different components between them. For a d-separable matrix, the Hamming distance of any two unions of d columns is at least one. Suppose that the Hamming distance of any two unions of d columns in a d-separable matrix is at least $2k+1$. Then decoding d positive items from testing outcomes is still possible even if there exist at most k error-tests. Indeed, we can choose the d columns whose union is within Hamming distance k from the test-outcome vector. Therefore, a d-separable matrix is said to be *k-error-correcting* if the Hamming distance of any two unions of d columns is at least $2k+1$.

For a k-error-correcting d-separable matrix, if the number of error-tests is more than k and less than $2k+1$, then we cannot decode the d positive items from a test-outcome column vector, however we can detect whether error exists or not. Indeed, if the test-outcome column vector is identical to a union of d columns, then these d columns corresponds to the d positive items. Otherwise, error exists. Therefore, we also call a k-error-correcting d-separable matrix as a $2k$-error-detecting d-separable matrix. In general, a d-separable matrix is said to be *k-error-detecting* if the Hamming distance of any two unions of d columns is at least $k+1$.

The following are basic properties of k-error-correcting matrices.

Lemma 2.3.1 *Deleting any k rows from a k-error-correcting d-separable matrix results in a d-separable matrix.*

Proof. Assume these k rows always make certain errors to disturb the identification. The separability has to come from the remaining rows and hence they form a d-separable matrix. □

Corollary 2.3.2 *The union of any d columns in a k-error-correcting d-separable matrix has weight at least $d+k-1$.*

Proof. Since a d-separable matrix is also $(d-1)$-separable, the union of any $d-1$ columns other than zero column in a d-separable matrix has weight at least $d-1$. It follows that the union of any d columns has weight at least d. By Lemma 2.3.1, this corollary holds. □

There is a similar issue about the d-disjunct matrix. A binary matrix is (d, k)-disjunct if for any union of d columns, every other column has at least $k+1$ 1-components not contained by the union.

Theorem 2.3.3 *For every (d,k)-disjunct matrix, the Hamming distance between any two unions of d columns is at least $2k+2$.*

Proof. Let $C_1 \cup \cdots \cup C_d$ and $C_{1'} \cup \cdots \cup C_{d'}$ be two different unions of d columns. Suppose C_1 is not in $\{C_{1'}, ..., C_{d'}\}$ and $C_{1'}$ is not in $\{C_1, ..., C_d\}$. By the definition of (d,k)-disjunctness, C_1 has at least $k+1$ 1-components not contained by $C_{1'} \cup \cdots \cup C_{d'}$ and $C_{1'}$ has at least $k+1$ 1-components not contained by $C_1 \cup \cdots \cup C_d$. Therefore, the Hamiming distance between the two unions

$$H(C_1 \cup \cdots C_d, C_{1'} \cup \cdots \cup C_{d'}) \geq 2(k+1).$$

□

Consequently, we have the following.

Corollary 2.3.4 *Every (d,k)-disjunct matrix is a k-error-correcting d-separable matrix and moreover, deleting any row, the remainder is still k-error-correcting d-separable.*

(d,k)-disjunctness via k-error-correcting property is similar to the d-disjunctness via d-separability. Decoding positives from testing based on (d,k)-disjunct matrix is much easier than from testing based on k-error-correcting d-separable matrix. To see this, let us first show the following lemma.

Lemma 2.3.5 *Suppose testing is based on a (d,k)-disjunct matrix. If the number of error tests is not more than k, then the number of negative pools containing a positive item is always smaller that the number of negative pools containing a negative item.*

Proof. Let i be a positive item and j a negative item. Suppose the number of negative pools containing i is m. Then these m pools must receive error tests. Therefore, there are at most $k-m$ error tests turning negative outcome to positive outcome. Moreover, we note that if no error exists, the number of negative pools containing j is at least $k+1$ by the definition of (d,k)-disjunctness. Hence, the number of negative pools containing j is at least $(k+1)-(k-m) = m+1 > m$. □

From the above lemma, we see that to decode positives from testing based on (d,k)-disjunct matrix, we only need to compute the number of negative pools containing each item and select d smallest ones. This runs in time $O(nt)$.

Hwang [4] considered a special case that errosr occur only in testing on negative pools, i.e., all negative outcomes are correct. In this case, if the number of errors is at most k in testing with (d,k)-disjunct matrix, then every negative item appears in a negative pool. Therefore, after remove all items appearing in negative pools, the remaining items all are positive. The running time for decoding in this way is $O(n)$.

Huang and Weng [3] consider another special case that the number of errors is at most $\lfloor k/2 \rfloor$ in testing with (d,k)-disjunct matrix. In this case, a positive items appears in at most $\lfloor k/2 \rfloor$ negative pools. But, a negative items appears in at least $k+1-\lfloor k/2 \rfloor (> \lfloor k/2 \rfloor)$ negative pools. Therefore, an items is positive if and only if it appeas in at most $\lfloor k/2 \rfloor$ negative pools. Using this condition, decoding runs in time $O(kn)$.

Now, we give a relation between (d,k)-disjunct matrix and $(d+k)$-disjunct matrix without isolated column. A column is *isolated* if there is a row containing only one 1 at the intersection with the column. In other words, there is a pool containing only one item. Usually, such pool (row) and item (column) can be eliminated from our consideration when we study construction of disjunct matrix.

Theorem 2.3.6 *Every $(d+k)$-disjunct matrix without isolated column is (d,k)-disjunct matrix.*

Proof. Consider a column C_0 and a union of d other columns, $C_1 \cup \cdots \cup C_d$. If the number of 1-components in C_0 but not in $C_1 \cup \cdots \cup C_D$ is less than $k+1$, then we may find at most k other columns to contain those components. (Since no isolated column exists, every row contains at least two 1s and hence the k other columns exist.) The union of these k columns together with $C_1, ..., C_d$ will contain C_0, contradicting $(d+k)$-disjunctness. □

Although constructing (d,k)-disjunct matrix can be reduced to construction of $(d+k)$-disjunct matrix, we will still introduce some special ways to construct (d,k)-disjunct matrix directly since the number of rows can be significantly smaller.

3 Error-Tolerance in $(\bar{d}, n)$ Space

The d-disjunct matrix serves for both (d,n) and $(\bar{d},n)$ spaces in the case of error-free. However, the situation in error-tolerance is different. In $(\bar{d},n)$ space, if we want a $\bar{d}$-separable matrix M to be able to identify sample from test outcomes with up to k errors, then M must have a property that the Hamming distance of two unions of at most d columns is at least $2k+1$. Such a matrix is called a *k-error-correcting $\bar{d}$-separable* matrix. In later chapter, we will see some examples which show that (d,k)-disjunct matrix may not be k-error-correcting $\bar{d}$-separable. Therefore, the decoding method of Wu *et al.* introduced in Section 2.3 does not work for $(\bar{d},n)$ space. Indeed, Theorem 2.3.5 holds also for $(\bar{d},n)$ space. However, we do not know how many smallest ones should be determined to be positive.

On the other hand, the decoding methods of Huang and Weng [3] holds for $(\bar{d},n)$ space. Actually, we have the following.

Theorem 3.0.7 *Every (d,k)-disjunct matrix is $\lfloor k/2 \rfloor$-error-correcting d-disjunct.*

Proof. For two different subsets of at most d columns, there must be one not contained by the other. Then, the union of the former contains at least $k+1$ 1-entries not appearing the union of the latter. This means that the Hamming distance of the two unions is at least $k+1$. Hence, it is $\lfloor k/2 \rfloor$-error-correcting for $\bar{d}$-separability. Moreover, (d,k)-disjunct matrix must be d-disjunct. Thus, it is k-error-correcting d-disjunct. □

The unsucess of (d,k)-disjunctness in $(\bar{d},n)$ space encourage one to look for other type of matrices.

A d-disjunct matrix is said to be *k-error-correcting* if the Hamming distance between two union of at most d columns is at least $2k+1$.

An interesting question is whether the time complexity of decoding for k-error-correcting d-disjunct matrix can be a polynomial with respect to both d and n where n is the number of columns in the considered matrix. Wu et al [17] gave a positive answer to the question.

Theorem 3.0.8 *There exists a decoding algorithm for k-error-correcting d-disjunct matrix, running in time $O((n+t)t^k)$.*

Proof. Let M be a k-error-correcting d-disjunct $t \times n$ matrix. Given outcomes from t tests on a sample of n clones with at most d positive ones, assume that the number of errors is at most k. We describe a decoding approach as follows.

Let E denote the set of error tests. The $|E| \leq k$. For each possible E, we remove all clones in negative pools not in E and all clones in positive pools in E. If the number of remaining clones is at most d, then we compute the Hamming distance between the given test outcomes and the true outcomes for the sample with all remaining clones as positives. If this Hamming distance does not exceed k, then we accept the result that all remaining ones are positive and all removed ones are all negative. Clearly, this method runs in time $O((n+t)t^k)$.

To show correctness of this method, we need to prove the following two claims:

(1) Theere exists an E such that the number of remaining clones does not exceed d.

(2) If the number of remaining clones does not exceed d, then the set of remaining clones remains the same.

Claim (1) follows from the d-disjunctness of M. Indeed, if E is the set of all pools on each of which test outcome is wrong, then all remaining clones should be positive and hence there exist at most d of them.

Claim (2) follows from the k-error-correcting property of M, by which, there exists exactly one sample with at most d positive clones such that the Hamming distance between the given test outcomes and the true outcomes on the sample does not exceed k. □

Is there a decoding method for k-error-correcting d-disjunct matrix running in a polynomial time with respect to n, d, t and k? The following result gives a negative answer.

Theorem 3.0.9 *There does not exist a decoding algorithm for k-error-correcting d-disjunct matrix running in a polynomial time with respect to n, d, t, and k unless NP=P.*

To show this theorem, let us first study the decoding for k-error-correcting $\bar{d}$-separable matrix.

The decoding for k-error-correcting $\bar{d}$-separable matrix is looking for a subset of at most d clones such that the number of positive pools not hit by the subset plus the number of negative pools hit by the subset does not exceed k. Thus, this decoding problem is closedly related to the following problem.

> DOUBLE-HITTING: Given two collections $\mathcal{C}$ and $\mathcal{D}$ of subsets of X and an positive integer d, find a subset A of at most d elenments of X to minimize the total number of subsets in $\mathcal{C}$ not hit by A and subsets in $\mathcal{D}$ hit by B.

Indeed, $\mathcal{C}$ is the collection of positive pools and $\mathcal{D}$ is the collection of negative pools. What we minimize is the number of error tests for the "hitting" subset over all possible set of positive clones. The difference is that for the decoding problem, we know that the minimum value of the objective function is at most k and want to find the subset to achieve this value.

Similarly, the decoding for k-error-correcting d-disjunct matrix is closedly related to a variation of DOUBLE-HITTING. Note that by d-disjunctness, the target set of positive clones is complement of the union of pools not hit by the target set. Thus, the minimization should be over all subsets A of cardinality at most d, satisfying the following property:

(*) $A = X - \cup_{B \in \mathcal{C} \cup \mathcal{D}, A \cap B = \emptyset} B$.

Formally, we state this variation as follows.

> DOUBLES-HITTING-(*): Given two collections $\mathcal{C}$ and $\mathcal{D}$ of subsets of X and an positive integer d, find a subset A of at most d clones, satisfying (*), to minimize the total number of subsets in $\mathcal{C}$ not hit by A and subsets in $\mathcal{D}$ hit by B.

Lemma 3.0.10 *If decoding for k-error-correcting d-disjunct matrix can be done in polynomial-time with respect to n, t, d, and k, then* DOUBLE-HITTING-(*) *can be solved in polynomial time.*

Proof. The decoding for k-error-correcting d-disjunct matrix is equivalent to the problem that knowing the minimum value of DOUBLE-HITTING-(*) is at most k, find a subset A of at most d clones, satisfying (*), to achieve the minimum. If there exits an algorithm K solving this problem in polynomial time with respect to n, t, d and k, then we may solve DOUBLE-HITTING-(*) by applying this algorithm K repeatly for $k = 1, 2, ..., t$. For each $k = 1, 2, ..., t$, if the algorithm K cannot find, in the polynomial time, a subset A of at most d clones, satisfying (*) to achieve the k-value, then restart the algorithm for next k-value, until a k-value is achieved by a subset A of at most d clones, satisying (*). This clearly still runs in polynomial time. □

Next, we show that BOUBLE-HITTING-(*) is NP-hard. To do so, we first study a variation of the vertex-cover problem as follows:

VERTEX-COVER: Given a graph G with n vertices and a positive integer h, $0 < h \le n$, determine whether G has a vertex-cover of size h.

VERTEX-COVER-(*): Given a graph G and two positive integer s m and h, determine whether G has a vertex subset of size at most h, covering at least m edges, such that the complement of the vertex-cover has no isolated vertex.

Lemma 3.0.11 VERTEX-COVER-(*) *is NP-complete.*

Proof. VERTEX-COVER-(*) is clearly in NP since we can guess the subset and verify in polynomial-time. We next construct a polynomial-time reduction from the well-known NP-complete problem VERTEX-COVER to VERTEX-COVER-(*).

Consider a graph G with n vertices and a positive integer h, $0 < h \le n$. Let m be h plus the number of edges of G. We construct another graph G' from G by additing $h + 1$ new vertices $v_0, v_1, ..., v_h$ and connecting v_0 to all vertices of G and $v_1, ..., v_h$. (Fig. 3.1) Next, we show that G has a vertex-cover of size h if and only if G has a

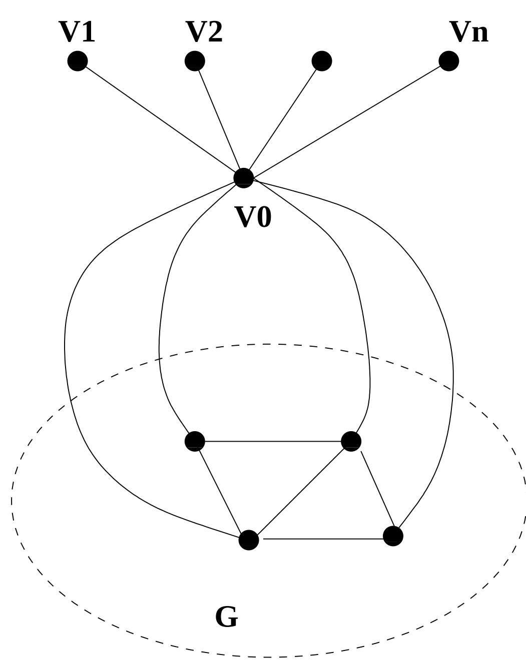

Figure 3.1: Construction of G' from G

vertex subset of size h, covering at least m edges, such that the complement of the subset has no isolated vertex.

First, if G has a vertex-cover of size at most h, then this vertex cover in G' has the required property.

Next, assume that G' has a vertex subset A of size at most h, covering at least m edges, such that the complement of A contains no isolated vertex. Note that v_0 cannot be in A. In fact, if v_0 is in A, then each v_i for $i = 1, ..., h$ has to be in A; otherwise, v_i would be isolated in the complement. However, all $v_0, v_1, ..., v_h$ being in A contradicts the size of A. Since v_0 is not in A and each vertex other than v_0 covers exactly one edge not in G, A covers at least $m - h$ edges in G. Hence, A covers all edges of G. Therefore, all vertices of G in A is a vertex-cover of size at most h. If this vertex-cover has size smaller than h, then we can simply add in more vertice to achieve the size h. □

Finally, we finish the proof of Theorem 3.0.9 by proving the following lemma.

Lemma 3.0.12 DOUBLE-HITTING-(*) *is NP-hard.*

Proof. Consider decision version of DOUBLE-HITTING-(*):

> Decision version of DOUBLE-HITTING-(*): Given two collections $\mathcal{C}$ and $\mathcal{D}$ of subsets of X and two positive integers d and k, determine whether or not there exists a subset A, satisfying (a) and (b), and the total number of subsets in $\mathcal{C}$ not hit by A and subsets in $\mathcal{D}$ hit by B is at most k.

Now, we construct a polynomial-time reduction from VERTEX-COVER-(*) to the decision version of DOUBLE-HITTING-(*). Consider an instance of VERTEX-COVER-(*), consisting of a graph G and two positive integers m and h. Let X be the vertex set of G and $\mathcal{C}$ the edge set of G. Set $\mathcal{D} = \emptyset$, $k = |\mathcal{C}| - m$, and $d = h$. We show that G has a vertex subset H of size at most h, hitting at least m edges, such that its complement contains no isolated vertex if and only if X has a subset A of size at most d satisfying condition (b) and the number of subsets in $\mathcal{C}$ not hit by A is at most k.

If G has such a vertex subset H, then set $A = H$ which is required A for X. Conversely, if X has such a subset A, then set $H = A$ and H is a required vertex subset for G. □

Every (d, k)-disjunct matrix is k-error-correcting d-separable. However, for $\bar{d}$-separability, only a weaker result exists.

We already saw that the (d, k)-disjunct matrix and the k-error-correcting d-disjunct matrix are quite different in term of decoding. Which one is the natural extension of the d-disjunct matrix in error-tolerant case? Wu and Li [18] indicated that the (d, k)-disjunct matrix is more like a natural generalization of d-disjunct matrix. Indeed, if we use (d, k)-disjunctness to replace d-disjunct, then many results for error-free can be generalized to error tolerant case.

Theorem 3.0.13 *A binary matrix M is k-error-correcting $\bar{d}$-separable if and only if it is k-error-correcting d-separable and $(d-1, 2k)$-disjunct.*

Proof. For forward direction, consider d distinct columns $C_0, C_1, ..., C_{d-1}$. By k-error-correcting $\bar{d}$-separability, the Hamming distance between $C_0 \cup C_1 \cup \cdots \cup C_{d-1}$ and $C_1 \cup \cdots \cup C_{d-1}$ must be at least $2k+1$. Hence, C_0 must have at least $2k+1$ 1-entries not in $C_1 \cup \cdots \cup C_d$. This means that the matrix M must be $(d-1, 2k)$-disjunct.

Conversely, assume M is k-error-correcting d-separable and $(d-1, 2k)$-disjunct. For two samples s and s' in $S(\bar{d}, n)$. If $|s| = |s'| = d$, then by the k-error-correcting d-separability of M, $H(U(s), U(s')) \geq 2k+1$, where $U(s)$ denotes the union of columns in s. Otherwise, we may assume, without loss of generality, that $|s| \geq |s'|$ and $|s'| \leq d-1$. Then s must have a column C_j not in s'. By the $(d-1, 2k)$-disjunctness of M, C_j contains at least $2k+1$ 1-entries not in $U(s')$. Thus, $H(U(s), U(s')) \geq 2k+1$. □

Theorem 3.0.14 *Any k-error-correcting $2d$-separable matrix M can be modified into a (d, k)-disjunct matrix by adding at most $k+1$ rows.*

Proof. Similar to the proof of Theorem 2.2.9. □

Corollary 3.0.15 *Any $2k$-error-correcting $2d$-separable matrix can be modified into a k-error-correcting $\overline{(d+1)}$-separable matrix by adding at most $k+1$ rows.*

Now, we have a summary for relations between concepts about error-tolerance.

$$\begin{array}{ccccc} & & (d+k)-\text{disjunct} & & \\ & & \Downarrow & & \\ \lceil k/2\rceil-\text{error-correcting} & & & & k-\text{error-correcting} \\ \overline{(d+1)}-\text{separable} & \Rightarrow & (d,k)-\text{disjunct} & \Rightarrow & d-\text{disjunct} \\ & & \Downarrow^{*} & & \Downarrow \\ & & k-\text{error-correcting} & & k-\text{error-correcting} \\ & & d-\text{separable} & \Leftarrow & \bar{d}-\text{separable} \end{array}$$

where $*$ means "delete any row".

References

[1] P. Busschbach, Constructive methods to solve the problems of: s-subjectivity, conflict resolution, coding in defective memories, unpublished manuscript, 1984.

[2] H. B. Chen and F. K. Hwang, Exploring the missing link among d-separable, $\bar{d}$-separable, and d-disjunct matrices, preprint, 2003.

[3] T. Huang and C.-W. Weng, A note on decoding of superimposed codes, *Journal of Combinatorial Optimization*, 7 (2003).

[4] F.K. Hwang, On Macula's error-correcting pooling design, to appear in *Discrete Mathematics*, 2003.

[5] S. M. Johnson, A new upper bound for error correcting codes, *IEEE Trans. Inform. Thy.* 8 (1962) 203-207.

[6] G. Katona, On separating systems of a finite set, *J. Combin. Thy.* 1 (1966) 174-194.

[7] W. H. Kautz and R. R. Singleton, Nonrandom binary superimposed codes, *IEEE Trans. Inform. Thy.* 10 (1964) 363-377.

[8] M. Li, Private communication.

[9] B. Lindstrom, Determination of two vectors from the sum, *J. Combin. Thy.* A6 (1969) 402-407.

[10] A. Rényi, On random generating elements of a finite Boolean algebra, *Acta Sci. Math.* (szeged) 22 (1961) 75-81.

[11] G. M. Saha and B. K. Sinha, Some combinatorial aspects of designs useful in group testing experiments, unpublished manuscript.

[12] D. J. Schultz, Topics in Nonadaptive Group Testing, Ph.D. Dissertation, Temple University, 1992.

[13] D. J. Schultz, M. Parnes and R. Srinivasan, Further applications of d-complete designs to group testing, *J. Combin. Inform. & System Sci.* 8 (1993) 31-41.

[14] E. Sperner, Ein Satz Uber Untermengen einer endliche Menge, *Math. Zeit.* 27 (1928) 544-548.

[15] I. Wegener, On separating systems whose elements are sets of at most k elements, *Disc. Math.* 28 (1979) 219-222.

[16] W. Wu, C. Li, X. Wu, and X. Huang, Decoding in pooling designs, *Journal of Combinatorial Optimization*, 7 (2003).

[17] W. Wu, C. Li, X. Huang, and Y. Li, On error-tolarent DNA screening, submitted to in *Discrete Applied Mathematics*.

[18] W. Wu and Y. Li, (d, k)-disjunctness via k-error-correcting d-disjunctness in error-tolerant DNA screening, preprint.