Discrete Applied Mathematics 157 (2009) 1581–1585

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Nonadaptive algorithms for threshold group testing

Hong-Bin Chen *, Hung-Lin Fu

Department of Applied Mathematics, National Chiao Tung University, Hsinchu, 30050, Taiwan

ARTICLE INFO

Article history:
Received 11 June 2007
Received in revised form 31 May 2008
Accepted 3 June 2008
Available online 9 July 2008

Keywords:
Threshold group testing
Nonadaptive algorithms
Graph search

ABSTRACT

Threshold group testing first proposed by Damaschke is a generalization of classic group testing. Specifically, a group test is positive (negative) if it contains at least u (at most l) positives, and if the number of positives is between l and u, the test outcome is arbitrary. Although sequential group testing algorithms have been proposed, it is unknown whether an efficient nonadaptive algorithm exists. In this paper, we give an affirmative answer to this problem by providing efficient nonadaptive algorithms for the threshold model. The key observation is that disjunct matrices, a standard tool for group testing designs, also work in this threshold model. This paper improves and extends previous results in three ways:
1. The algorithms we propose work in one stage, which saves time for testing.
2. The test complexity is lower than previous results, at least for the number of elements which need to be tested is sufficiently large.
3. A limited number of erroneous test outcomes are allowed.

1. Introduction

In the classic group testing problem, we consider a set N of n items consisting of at most d positive items with the other being negative items. Typically d is much smaller than n. A group test is an arbitrary subset of items, called a pool. A test yields a positive outcome if and only if the tested subset of items contains at least one positive item. The task is to identify all positive items with group tests as few as possible.

Group testing is a basic tool which can be applied to a variety of problems such as blood testing, multiple access communication, coding theory, and others. Recently group testing procedures have proved to be useful in the area of Computational Molecular Biology, for example, in screening clone libraries [6] and sequencing by hybridization [15]. Here we refer to the book by Du and Hwang [10] for an overview.

Biological applications not only call for more complex models but also require search strategies to fulfill different criterions of performance. Typically, the main concern of group testing is to minimize the number of tests required. However, things become more complicated when it comes to molecular biology applications. While minimizing the number of tests is still important in the applications to molecular biology, two other goals emerge. There are many biological experiments which can be time-consuming during the testing procedure. Screening the pools one by one is far more expensive than screening many pools in parallel. For this reason, it is impractical to screen the pools sequentially, and thus nonadaptive algorithms for testing are desirable, i.e., all tests are specified in advance without knowing the outcomes of other tests. Besides, there is another feature that differentiate biology-motivated group testing problems from traditional ones. In the classical scenario it is supposed that test outcomes are quite reliable. In biological applications, however, experimental errors may be made and cannot be ignored. With experimental errors, test outcomes are unreliable and may consist of

* Corresponding author. Fax: +886 3 572 4679.
E-mail address: andan.am92g@nctu.edu.tw (H.-B. Chen).

0166-218X/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2008.06.003

false negative outcomes and false positive outcomes. In practice, the decoding issue becomes even more difficult; hence algorithms with ability of error tolerance are much preferable.

In this paper we discuss a generalization of group testing which is a novel model first proposed by Damaschke [9]. The problem is described as follows. Consider a set N of n items consisting of at most d positive items with the other being negative items. Let l and u be nonnegative integers with $l < u$, called the *lower* and *upper threshold*, respectively. A group test for a subset S of items is positive if S contains at least u positives, and negative if at most l positives are present. If the number of positives in S is between l and u, the test outcome is arbitrary. Clearly, the classic group testing problem is a special case with $l = 0$ and $u = 1$. Denote by P the set of positive items. In the threshold model, some lower bound on $|P|$ is necessary as a prior assumption. For example, if $|P| < u - 1$ and every test yields a negative outcome, we can conclude nothing but $|P| < u - 1$, even if we test all possible 2^n subsets. Thus, throughout this paper, we assume that $u \le |P| \le d$.

As in the classic group testing problem, it is desirable that there is a way to identify the set P exactly. Unfortunately, this goal cannot always be achieved in the threshold model. Let $g \equiv u - l - 1$ denote the *gap* between the lower and upper thresholds. For the general case, it has been shown [9] that the $|P|$ positives can be determined with up to a constant number of misclassifications, bounded by the gap g. Indeed, Damaschke [9] proved that some set P' with $|P' \backslash P| \le g$ and $|P \backslash P'| \le g$ can be identified by simply testing all u-subsets, i.e., in $\binom{n}{u}$ tests. In addition, Damaschke also studied the special case without gap, $g = 0$. For this case he proved that there is an algorithm with test complexity $O(d \log n)$ for any fixed u. Nevertheless, he only considered sequential algorithms; specifically, tests cannot be performed in parallel (because some tests depend on the outcomes of other tests). It remains an open question whether there exists an efficient nonadaptive algorithm for the threshold model.

In this paper, we extend threshold group testing to the error-tolerant version where at most e erroneous outcomes are allowed. To understand and construct an efficient nonadaptive algorithm for the threshold model, we start with the special case that $g = 0$. Unexpectedly, we find a relation between the special case of the threshold model and graph search problems. Establishing such a connection leads to the consequence that disjunct matrices, a standard tool for graph search, can also be useful in the special case of threshold group testing. Furthermore, this idea can also be extended to the general threshold model with a slight modification. Consequently, we answer Damaschke's question by providing an efficient nonadaptive algorithm for the general threshold model.

The remainder of this paper is organized as follows. Section 2 introduces the main structure, disjunct matrices, on which our results depend. Section 3 focuses on the special case, the threshold model without gap, and establishes a connection to graph search problems. In Section 4, we provide an efficient nonadaptive algorithm for the general error-tolerant threshold model. Mainly, we prove that there exists a nonadaptive algorithm for the general error-tolerant threshold model in $(2e+1)\left(\frac{d+u-l}{u}\right)^u\left(\frac{d+u-l}{d-l}\right)^{d-l}\left[1 + (d+u-l)\left(1 + \ln(\frac{n}{d+u-l} + 1)\right)\right]$ tests such that some set P' with $|P' \backslash P| \le g$ and $|P \backslash P'| \le g$ can be identified. Note that for the case $e = 0$, our result requires much fewer tests than that of Damaschke [9] when n is sufficiently large.

2. Preliminaries

First, we adopt some notations and definitions discussed in the rest of this paper. A nonadaptive group testing algorithm can be represented by a 0–1 matrix where the columns are the set of items, the rows are the set of tests, and cell $(i, j) = 1$ signifies that item j is in test i and $(i, j) = 0$ for otherwise. For convenience, a column can be treated as the set of row indices where the column has a 1. Then we can talk about the union and the intersection of columns. We say that a set X of columns appears (or is contained) in a row if all columns in X have a 1 in the row. A pool with a negative (positive) outcome is called a negative (positive) pool, respectively. For a subset X of columns, define $t_0(X)$ to be the number of negative pools in which all columns in X appear.

A 0–1 matrix is said to be $(d, r; z]$-disjunct if for any $d + r$ columns $C_1, C_2, \ldots, C_{d+r}$,

$$\left| \bigcap_{i=1}^{r} C_i \backslash \bigcup_{i=r+1}^{d+r} C_i \right| \ge z,$$

i.e., for any $d+r$ columns there exist at least z rows where the r designated columns have 1-entries and the other d columns have 0-entries.

For the particular case $r = z = 1$, the $(d, r; z]$-disjunct matrices were first introduced by Kautz and Singleton [13] in the context of superimposed binary codes. On the other hand, this structure has also been studied elsewhere under other name such as cover-free families. Stinson and Wei [16] first studied the generalized cover-free families ($(d, r; z]$-disjunct matrices), and provided some bounds and constructions. Further studies on this structure can be found in [7,8,11,14,16,17]. Let $t(n, d, r; z]$ denote the minimum number of rows among all $(d, r; z]$-disjunct matrices with n columns. A recent upper bound by Chen, Fu and Hwang [8] is that

$$t(n, d, r; z] < z(k/r)^r (k/d)^d [1 + k(1 + \ln(n/k + 1))], \quad (1)$$

where $k = d + r$.

The $(d, r; z]$-disjunct matrices have been applied to a variety of problems such as graph search problems, DNA complex screening, superimposed codes and secure key distribution [7]. In recent group testing literature, this structure has become a major tool in understanding and constructing nonadaptive algorithms. In the rest of this paper, we will first establish a connection between threshold group testing and graph search problems, and then exploit the $(d, r; z]$-disjunct matrices to construct efficient nonadaptive algorithms for the threshold group testing problems.

3. The case without gap

In this section we deal with the case when the gap between the upper and lower thresholds is zero, that is, a pool is positive if and only if it contains at least u positives. For the particular case, we establish a connection to a seemingly unrelated problem: graph search. As a consequence, we propose a nonadaptive algorithm for the threshold group testing without gap by using disjunct matrices, a standard tool for graph search.

To present our results, we now introduce the graph search problem we consider here. For a given vertex set $V = \{1, 2, \ldots, n\}$, the goal is to reconstruct a hidden graph H defined on V by asking queries of the following form: for $F \subseteq V$, the query Q_F is that "Does F contain at least one edge of H?". Precisely, a pool is positive if and only if F contains at least one edge of H. The task here is to find out all edges of the hidden graph in an efficient fashion by using the above query model. There are different graph search problems according to prior knowledge of the hidden graph H. The usual assumption is that the number of edges in H, denoted by $\|H\|$, is upper bounded by a constant d; but it can also be H is a matching [2,4], a Hamiltonian circuit [12] or some others [1,3,5]. Our interest here will be the assumption that $|H|$ is upper bounded by a constant d, where $|H|$ denote the number of vertices induced by the edges of H.

Of particular note is that graphs we discuss here are hypergraphs. Given a finite set V, a hypergraph $\mathcal{H} = (V, \mathcal{F})$ is a family $\mathcal{F} = \{E_1, E_2, \ldots, E_m\}$ of subsets of V. The elements of V are called vertices, and the subsets E_i's are the edges of the hypergraph $\mathcal{H}$. A hypergraph is called a u-hypergraph if each edge consists of exactly u vertices. Let M be a subset of V. A hypergraph is u-complete with respect to M if and only if every u-subset of M is an edge of the hypergraph.

In [7], Chen and Hwang proved that a $(d, u; 2e+1]$-disjunct matrix can be used to reconstruct any hidden u-hypergraph $\mathcal{H}$ with at most e erroneous tests under the assumption that $\|\mathcal{H}\| \le d$. Using a similar argument to that of Chen and Hwang, we obtain that a $(d-u+1, u; 2e+1]$-disjunct matrix can be used to reconstruct any hidden u-hypergraph $\mathcal{H}$ with at most e erroneous tests under the assumption that $|\mathcal{H}| \le d$.

Theorem 3.1. *Assume that erroneous outcomes are upper bounded by a constant e. Then, any hidden u-hypergraph $\mathcal{H} = (V, \mathcal{F})$ with $|\mathcal{H}| \le d$ can be reconstructed by using a $(d-u+1, u; 2e+1]$-disjunct matrix.*

Proof. Obviously, every pool containing any edge in $\mathcal{H}$ should be positive except erroneous pools. Even for the worst case that e pools are erroneous, we have $t_0(X^+) \le e$ for every u-subset $X^+ \in \mathcal{F}$.

On the other hand, it suffices to show that $t_0(X) \ge e+1$ for every u-subset $X \notin \mathcal{F}$. Let M be the set of vertices induced by the edges in $\mathcal{F}$. For each u-subset $X \notin \mathcal{F}$, we have that either $|X \cap M| \le u-1$ or $|X \cap M| = u$. If $|X \cap M| \le u-1$, then we can choose a $(d-u+1)$-subset Y disjoint from X such that the number of vertices in M but not in Y is at most $u-1$. If $|X \cap M| = u$, then let Y be any $(d-u+1)$-subset, disjoint from X, containing all vertices in $M \backslash X$. In any case, there is no edge induced by the vertices outside Y. By the $(d-u+1, u; 2e+1]$-disjunctness property, there are at least $2e+1$ rows each containing X but none of Y. The pools corresponding to these rows should be negative since these pools contain no edge of $\mathcal{F}$. Therefore, even for the worst case that e pools are erroneous, X still appears in at least $e+1$ negative pools. Hence, $t_0(X) \ge e+1$ for every u-subset $X \notin \mathcal{F}$.

From the above discussion, we can determine whether an edge X belongs to $\mathcal{F}$ or not. ■

We now turn our attention back to the threshold model without gap. Recall that our goal is to identify all positives in P from a given set N by testing whether a pool contains at least u positives or not. By treating N as a vertex set V and the set of all u-subsets of P as an edge set $\mathcal{F}$, the threshold model without gap easily fits into the framework of reconstructing a hidden hypergraph $\mathcal{H} = (V, \mathcal{F})$ which is u-complete with respect to P. It is easy to see that a hypergraph $\mathcal{H} = (V, \mathcal{F})$ which is u-complete with respect to P is also a u-hypergraph with $|\mathcal{H}| \le d$. By Theorem 3.1, we then have the following result.

Theorem 3.2. *For the error-tolerant threshold model without gap, a $(d-u+1, u; 2e+1]$-disjunct matrix can be used to identify the positive set P.*

For $u = 1$ and $e = 0$, our result agrees with the well-known result that a d-disjunct matrix can be used to identify up-to-d positives on the classic group testing problem.

The decoding algorithm for computing the set P is described in the following.

Algorithm 1 (*For the Error-tolerant Threshold Model Without Gap*).

Tool: Use a $(d-u+1, u; 2e+1]$-disjunct matrix.

Decoding: Let X be a variable for sets of u elements. Output the set S, which is the union of elements in X with $t_0(X) \le e$, i.e., $S = \{x : x \in X \text{ and } t_0(X) \le e\}$.

To compute the set S, it suffices to verify the condition $t_0(X) \le e$ for each u-subset X separately. Clearly, the decoding complexity is $t_0 u \binom{n}{u}$, where t_0 is the number of negative pools. Remark that we cannot decode faster than by testing the condition $t_0(X) \le e$ for each X separately as seen by the following example. In the case when $d = u$, we cannot avoid verifying the above condition separately to find the unique u-subset P.

Theorem 3.3. *For the error-tolerant threshold model without gap, there exists a nonadaptive algorithm that successfully identifies up-to-d positives, using a number of tests no more than* $(2e+1)\left(\frac{d+1}{u}\right)^u\left(\frac{d+1}{d-u+1}\right)^{d-u+1}\left[1+(d+1)\left(1+\ln(\frac{n}{d+1}+1)\right)\right]$. *Moreover, the decoding complexity is* $O(n^u \log n)$ *for fixed parameters* (d, e).

Proof. The theorem is obtained by applying the upper bound of inequality (1) to $t(n, d-u+1, u; 2e+1]$ and by the above discussion. ■

4. The general case

In this section, we consider the general case for the threshold model. Things become more complicated if there is a gap between the upper and lower thresholds. This is much different from what happens in the threshold model without gap for which the positive set P can be identified exactly. In [9], Damaschke showed that to find some set P' with $|P'\backslash P| \le g$ and $|P\backslash P'| \le g$ is the best possible result. In view of this, for the general case of the threshold model, we attempt to find some set P' with the properties mentioned above, instead of identifying the set P exactly. Fortunately, the idea of using disjunct matrices as a tool also works in the general threshold model. We first state our algorithm, and then prove its correctness in the following.

Algorithm 2 (*For the General Error-tolerant Threshold Model*).

Tool: Use a $(d-l, u; 2e+1]$-disjunct matrix.

Decoding:

Step 1: Construct a hypergraph $\mathcal{H}_u = (V, \mathcal{F})$ where $V = N$ is the vertex set and a u-subset $X \subseteq N$ is an edge in $\mathcal{F}$ if and only if $t_0(X) \le e$.

Step 2: We want to establish increasing vertex-sets P_i's, $|P_1| < |P_2| < \cdots < |P_m|$, such that the hypergraph $\mathcal{H}_u = (V, \mathcal{F})$ is u-complete with respect to each P_i. As an initial P_1, we may choose all u vertices of an arbitrary edge. To find P_2, we check all possible cases to obtain some $(g+1)$-set A in $V(\mathcal{H}_u) \setminus P_1$ and a g-set B in P_1 such that $\mathcal{H}_u$ is u-complete with respect to $(P_1 \cup A) \setminus B$. If such a pair A, B exists, then set $P_2 = (P_1 \cup A) \setminus B$. Continue this process till either P_m is not extendible or $|P_m| \ge d$. Output the set $P' = P_m$.

Lemma 4.1. *After Step* 1*, for every u-subset* $X \in \mathcal{F}$*, X contains no more than g items not in P. Moreover, every u-subset* $X^+ \subseteq P$ *must be in* $\mathcal{F}$.

Proof. It is easy to see that every u-subset $X^+ \subseteq P$ should appear in positive pools except erroneous pools. Even for the worst case that e pools are erroneous, we have $t_0(X^+) \le e$. Hence, every u-subset $X^+ \subseteq P$ is in $\mathcal{F}$ after Step 1.

We now want to show that $t_0(X) > e$ for every u-subset X containing more than g items not in P. A u-subset X containing more than g items not in P implies that at most l positives are in X. Thus, we can choose a $(d-l)$-subset Y disjoint from X such that the number of positives not in Y is at most l. Such Y can always be chosen by putting as many positives as possible in Y. By the $(d-l, u; 2e+1]$-disjunctness property, there exist at least $2e+1$ rows each containing X but none of Y. Without erroneous outcomes, the pools corresponding to these rows should be negative because each of them contains at most l positives. Thus, even for the worst case that e pools are erroneous, X still appears in at least $e+1$ negative pools. Therefore, $t_0(X) > e$ for every u-subset X containing more than g items not in P. ■

Theorem 4.2. *Algorithm 2 outputs some set* P' *satisfying both* $|P'\backslash P| \le g$ *and* $|P\backslash P'| \le g$.

Proof. By Lemma 4.1, every P_i during Step 2 contains no more than g vertices not in P since any u-subset containing any $g+1$ of those vertices cannot be an edge in $\mathcal{F}$. Therefore, we obtain $|P'\backslash P| \le g$.

On the other hand, observe that Algorithm 2 terminates when P' is not extendible or $|P'| \ge d$. If $|P'| \ge d$, then we have $|P\backslash P'| \le g$ since $|P| \le d$ and $|P'\backslash P| \le g$. Now, it suffices to show that if P' is not extendible, then $|P\backslash P'| \le g$. Assume that $|P\backslash P'| > g$. Set $A \subseteq P\backslash P'$ with $|A| = g+1$, and let B be any subset with $P'\backslash P \subseteq B \subseteq P'$ and $|B| = g$. The existence of B follows by the fact that $|P'\backslash P| \le g$ and the initial P' has $u > g$ elements. It is easy to see that $(P' \cup A) \setminus B$ is contained in P. Consequently, $\mathcal{H}_u$ is u-complete with respect to $(P' \cup A)\backslash B$, which is larger than P', a contradiction to that P' is not extendible. ■

The following result is an immediate consequence of Theorem 4.2.

Corollary 4.3. *For the general error-tolerant threshold model, a* $(d-l, u; 2e+1]$*-disjunct matrix can be used to identify some set* P' *with* $|P'\backslash P| \le g$ *and* $|P\backslash P'| \le g$.

Our next concern is to compute some set P' with $|P'\backslash P| \le g$ and $|P\backslash P'| \le g$ from the test outcomes efficiently. In Step 1, it suffices to simply count $t_0(X)$ for each u-subset X. This can be done in time $t_0 u \binom{n}{u} = O(n^u \log n)$ for fixed parameters (d, e), where t_0 is the number of negative pools. Next, to find a pair A, B with the desired property in Step 2, we may verify all $\binom{n-u}{g+1}$ candidates for A. For any fixed A, to find a suitable B, we need to verify at most $\binom{d}{g}$ candidates in P_i since $|P_i| \leq d$ for each P_i's. Therefore, a pair A, B with the desired property can be found, if it exists, in $\binom{n-u}{g+1}\binom{d}{g}\binom{d}{u}$ time in the worst case. This process has to be executed at most $d-u$ times since Step 2 starts with $|P_1| = u$ and terminates with $|P_m| \leq d$. Consequently, in Step 2, some set P' with both $|P' \backslash P| \leq g$ and $|P \backslash P'| \leq g$ can be computed in $(d-u)\binom{n-u}{g+1}\binom{d}{g}\binom{d}{u} = O(n^{g+1})$ time for fixed d. Hence, the decoding complexity is dominated by Step 1, $O(n^u \log n)$ for fixed parameters (d, e). Remark that the idea of Step 2, using such a pair A, B for exchanging, is similar to that of Damaschke [9]. We now formulate the final result of this section in the following.

Theorem 4.4. *For the general error-tolerant threshold model, there exists a nonadaptive algorithm that successfully identifies some set P' with $|P' \backslash P| \leq g$ and $|P \backslash P'| \leq g$, using no more than* $(2e+1)\left(\frac{d+u-l}{u}\right)^u \left(\frac{d+u-l}{d-l}\right)^{d-l} \left[1 + (d+u-l)\left(1 + \ln\left(\frac{n}{d+u-l} + 1\right)\right)\right]$ *tests. Moreover, the decoding complexity is $O(n^u \log n)$ for fixed parameters (d, e).*

Proof. The theorem is obtained by applying the upper bound of inequality (1) to $t(n, d-l, u; 2e+1]$ and by the above discussion. ■

References

[1] N. Alon, V. Asodi, Learning a hidden subgraph, SIAM J. Discrete Math. 18 (2005) 697–712.
[2] N. Alon, R. Beigel, S. Kasif, S. Rudich, B. Sudakov, Learning a hidden matching, SIAM J. Comput. 33 (2004) 487–501.
[3] M. Aigner, Combinatorial Search, John Wiley and Sons, 1988.
[4] R. Beigel, N. Alon, M.S. Apaydin, L. Fortnow, S. Kasif, An optimal procedure for gap closing in whole genome shotgun sequencing, in: Proc. 2001 RECOMB, ACM Press, pp. 22–30.
[5] M. Bouvel, V. Grebinski, G. Kucherov, Combinatorial search on graphs motivated by bioinformatics applications: A brief survey, in: Graph-Theoretic Concepts in Computer Science, in: LNCS, vol. 3787, 2005, pp. 16–27.
[6] W.J. Bruno, D.J. Balding, E. Knill, D. Bruce, C. Whittaker, N. Dogget, R. Stalling, D.C. Torney, Design of efficient pooling experiments, Genomics 26 (1995) 21–30.
[7] H.B. Chen, D.Z. Du, F.K. Hwang, An unexpected meeting of four seemingly unrelated problems: Graph testing, DNA complex screening, superimposed codes and secure key distribution, J. Combin. Optim. 14 (2007) 121–129.
[8] H.B. Chen, H.L. Fu, F.K. Hwang, An upper bound of the number of tests in pooling designs for the error-tolerant complex model, Optim. Lett., in press (doi:10.1007/s11590-007-0070-5).
[9] P. Damaschke, Threshold group testing, General Theory of Information Transfer and Combinatorics, in: LNCS, vol. 4123, 2006, pp. 707–718.
[10] D.Z. Du, F.K. Hwang, Pooling Designs and Nonadaptive Group Testing - Important Tools for DNA Sequencing, World Scientific, 2006.
[11] A.G. D'yachkov, P.A. Vilenkin, A.J. Macula, D.C. Torney, Families of finite sets in which no intersection of ℓ sets is covered by the union of s others, J. Combin. Theory Ser. A 99 (2002) 195–218.
[12] V. Grebinski, G. Kucherov, Reconstructing a Hamiltonian cycle by querying the graph: Application to DNA physical mapping, Discrete Appl. Math. 88 (1998) 147–165.
[13] W.H. Kautz, R.R. Singleton, Nonrandom binary superimposed codes, IEEE Trans. Inform. Theory 10 (1964) 363–377.
[14] H.K. Kim, V. Lebedev, On optimal superimposed codes, J. Combin. Designs 12 (2004) 79–91.
[15] P.A. Pevzner, R. Lipshutz, Towards DNA sequencing chips, in: Mathematical Foundations of Computer Science, in: LNCS, vol. 841, 1994, pp. 143–158.
[16] D.R. Stinson, R. Wei, Generalized cover-free families, Discrete Math. 279 (2004) 463–477.
[17] D.R. Stinson, R. Wei, L. Zhu, Some new bounds for cover-free families, J. Combin. Theory Ser. A 90 (2000) 224–234.