Decoding algorithms in pooling designs with inhibitors and error-tolerance

My T. Thai*

Department of Computer
and Information Science and Engineering,
University of Florida,
P.O. Box 116120, Gainesville, FL 32611, USA
E-mail: mythai@cise.ufl.edu
*Corresponding author

David MacCallum

Department of Computer Science,
University of Minnesota,
Twin Cities, 200 Union Street S.E.,
Minneapolis, MN 55455, USA
E-mail: dmac@cs.umn.edu

Ping Deng and Weili Wu

Department of Computer Science,
University of Texas at Dallas,
Richardson, TX 75083, USA
E-mail: pxd010100@utdallas.edu
E-mail: weiliwu@utdallas.edu

Abstract: Pooling designs are used in DNA library screening to efficiently distinguish positive from negative clones, which is fundamental for studying gene functions and many other biological applications. One challenge is to design decoding algorithms for determining whether a clone is positive based on the test outcomes and a binary matrix representing the pools. This is more difficult in practice due to errors in biological experiments. More challenging still is a third category of clones called 'inhibitors' whose effect is to neutralise positives. We present a novel decoding algorithm identifying all positive clones in the presence of inhibitors and experimental errors.

Keywords: decoding algorithms; pooling designs; group testing; inhibitors; bioinformatics.

Reference to this paper should be made as follows: Thai, M.T., MacCallum, D., Deng, P. and Wu, W. (2007) 'Decoding algorithms in pooling designs with inhibitors and error-tolerance', *Int. J. Bioinformatics Research and Applications*, Vol. 3, No. 2, pp.145–152.

Biographical notes: My T. Thai received her PhD Degree in Computer Science from the University of Minnesota in 2005. She is an Assistant Professor at the Department of Computer and Information Science and

Engineering, the University of Florida. Her research interests include wireless networks, computational biology, applied algorithms and combinatorics. She is a Member of the IEEE Computer Society.

David MacCallum is scheduled to receive his PhD in Computer Science from the University of Minnesota in August 2006. He received his PhD in Philosophy from the University of Maryland in 1992. He is an Associate Professor in the Philosophy Department at Carleton College. His research interests include computational biology, sensor networks, quantum computation and logic.

Ping Deng received her Master Degree in Computer Science from the University of Texas at Dallas in August, 2003. She is a PhD Candidate in Computer Science at the same university. Her research interests include data mining, intrusion detection and bioinformatics.

Weili Wu received her MS and PhD Degrees in Computer Science from the University of Minnesota, in 1998 and 2002 respectively. She is currently an Assistant Professor and a Lab Director of the Database Research Lab at the Department of Computer Science and Engineering, the University of Texas at Dallas. Her research interest is mainly in database systems, especially in spatial database with applications in geographic information systems and bioinformatics, distributed database in internet systems and wireless database systems with connection to wireless communication. She is a member of the IEEE Computer Society.

1 Introduction

Recent advances in biology and technology, especially the success of the Human Genome Project, have made the study of gene functions more popular. The study of gene functions requires a high quality DNA library, which is a collection of the copies of DNA fragments, called *clones*. Unfortunately, the high quality DNA library is usually obtained through a large amount of testing and screening. Therefore, it requires techniques to reduce the number of testings and screenings. One such technique is a pooling design.

The pooling design is also called non-adaptive group testing, which is a mathematical tool to significantly reduce the number of tests in DNA library screening (Dychkov et al., 2001; Farach et al., 1997; Ngo and Du, 2000) and it also has many other biological applications (Du and Hwang, 1999, 2006; Gao et al., 2006; Li, 2006; Macula et al., 2000, 2004; Torney, 1999; Triesch, 1996). In DNA library screening, the basic problem of pooling designs is to identify the set of all positive clones in a large population of clones with the minimum number of tests. A clone is positive if it contains a given probe; otherwise, it is negative. In pooling designs, each test is performed on a subset of clones, called *pools*, rather than on an individual clone. For example, the Life Science Division of Los Alamos National Laboratories in 1998 (Marathe et al., 2000) faced 220,000 clones for testing. Testing those clones individually would require 220,000 tests whereas with pooling designs, they used only 376 tests. Each pool contains approximately 5000 clones. Clearly, pooling designs can help tremendously in reducing the number of tests.

Research on pooling designs usually contains two parts:

- designing an efficient decoding algorithm
- designing the pools to meet the requirements of decoding algorithms and to have fewer number of tests.

In this paper, we study the decoding algorithm problem, which is to determine all positive clones based on the test outcomes and the constructed pools. In the classical model, the test outcome of a pool is positive if it contains at least one positive clone; otherwise, it is negative.

However, in practice, the decoding problem becomes even more difficult due to the experimental errors in biology. With the experimental errors, the test outcomes may consist of false negatives or false positives. In the former, a test yields a negative outcome when a pool consists of at least one positive clone. Likewise, in the latter, a test yields a positive outcome when a pool does not contain any positive clone.

More challenging, in some applications, besides positive and negative clones, there is a third category of clones called 'inhibitors' whose effect is to neutralise positives (Farach et al., 1997). In other words, the presence of a single inhibitor in a pool dictates the test outcome to be negative. One example of inhibitors is an enzyme inhibitor, which is a molecule that binds to the active site of an enzyme during the reaction process, thus preventing the success of this process. Similarly, in the pooling testing, the inhibitors will spoil the clones in the pools which make the test outcomes of the pools become negative.

In this paper, we study the decoding algorithms in the Inhibitors and Error-Tolerance (IET) model, in which there are n clones with *at most* d positive clones and *at most* s inhibitors, subject to *at most* e experimental errors. There exists several decoding algorithms in the IET model (Farach et al., 1997; De Bonis and Vaccaro, 1998; Hwang and Liu, 2006). However, most of them used a sequential k-round approach for some constant $k \geq 2$. In this approach, the pools are constructed based on the test outcomes of the previous tests. Hence, the pools may be re-constructed k times. For example, in the first round, the pools of n clones are constructed and tested to identify all s inhibitors. Then in the second round, the pools of $n - s$ clones are constructed and tested to identify all positive clones. This approach is expensive since testing and conducting the pools are time-consuming. In Hwang and Liu (2006), the authors introduced the 1-round decoding algorithm. In this paper, we present a novel 1-round decoding algorithm with fewer number of tests (pools) than that of (Hwang and Liu, 2006) for the inhibitors and error-tolerance model.

2 Preliminaries and trade-offs

A pooling design consisting of t pools and dealing with n clones can be represented by a $t \times n$ binary matrix M with rows representing the pools and columns representing the clones. A cell $M[i, j] = 1$ if and only if the ith pool contains the jth clone; otherwise, $M[i, j] = 0$. Given S as a set of columns in M, then *Union*(S) is defined as the boolean sum of all the columns in S. For example, let $S = \{(1, 0, 0)^T, (0, 1, 0)^T, (1, 1, 0)^T\}$, then Union($S$) = $(1, 1, 0)^T$.

Consider a binary matrix M, we have the following definitions:

Definition 1 ***$\bar{d}$-separable*****:** M is said to be d-separable if for any two subsets S and S' of columns in M with $\max\{|S|, |S'|\} \leq d$ and $S \neq S'$, Union(S) $\neq$ Union(S').

Definition 2 ***d-disjunct*****:** M is said to be d-disjunct if for any column C_j and any set S of d columns in M such that $C_j \notin S$, C_j is not contained in Union (S).

Definition 3 ***(d, z)-disjunct*****:** M is said to be (d, z)-disjunct if for any column C_j and any set S of d columns in M such that $C_j \notin S$, C_j has at least $z + 1$ 1-entries not contained in Union (S).

Definition 4 ***($\bar{d}$, z)-separable*****:** M is said to be ($\bar{d}$, z)-separable if for any sets S and S' of at most d columns in M with $S \neq S'$, the Hamming distance H(Union (S), Union (S')) $\geq z$.

Here, the Hamming distance of two columns C_i and C_j, i.e., $H(C_i, C_j)$, is defined at the number of different components between these two columns.

Given a binary matrix $M_{t\times n}$, the test outcomes of these t pools can be represented by a t-dimensional column vector V, called the *test outcome vector*. Note that V is a binary vector, in which 1 represents a positive outcome whereas 0 represents a negative outcome. In the classical model, where there is no inhibitors and no errors, V is the union of columns corresponding to positive clones in M. Hence, one possible solution for the decoding algorithms in the classical model is that for each set S of at most d columns in matrix M, check whether the test outcome vector V matches the Union (S). In order for this algorithm to work, matrix M must be $\bar{d}$-separable. Note that the time complexity of this algorithm is $O(n^d)$.

Now let us consider another decoding algorithm in the classical model. When M is a d-disjunct matrix, we have this following lemma:

Lemma 1: *For testing based on a d-disjunct matrix, the number of clones not appearing in any negative pool is always no more than d (Du and Hwang, 2006).*

Based on Lemma 1, the authors in (Du and Hwang, 2006) presented an $O(n)$ decoding algorithm. In this algorithm, all clones appearing in negative pools are removed and the remaining clones must be positive. For this algorithm to work, matrix M must be d-disjunct.

Note that d-disjunct implies $\bar{d}$-separable. This means that d-disjunct is stronger property; therefore, the number of tests in a d-disjunct matrix is more than that of a $\bar{d}$-separable matrix. Thus there is a trade-off between the time complexity of decoding algorithms and the number of tests.

3 Main results

In this section, we study the decoding problem in the IET model and present a 1-round decoding algorithm.

Definition 5 ***Problem Definition*****:** Given a binary matrix M and a test outcome vector V from a sample of n clones with at most d positive ones and at most s inhibitors, subject to at most e experimental errors, design an efficient decoding algorithm to determine all positive clones.

Let us first consider a special case of the IET model, in which there is n clones with at most d positive ones and at most s inhibitors and no errors, i.e., $e = 0$. Define R as a set containing all clones not appearing in a positive pool. Thus set R contains all inhibitors and no positive clones. Let S be a subset of R where $|S| \leq s$. For any clone A, define:

$$t^S = \begin{cases} \infty & \text{if } A \in S \\ \#\text{of negative tests containing } A & \text{otherwise} \end{cases}.$$

Note that when $A \notin S$, the number of negative pools containing clone A is computed *after* changing a negative test outcome of a pool to a positive outcome if this pool contains a clone in S. In other words, we assume that the set S contains exactly all inhibitors. The details of how to compute the value $t^S(A)$ are shown in procedure NEGATIVE-POOLS (see Algorithm 1).

Now define $t^*(A) = \min_{S \subseteq R} t^S(A)$, we have:

Lemma 2: *If M is (d + s)-disjunct then:*

(a) $t^*(P) = 0$

(b) $t^*(Q) > 1$

(c) $t^*(I) > d$

where P represents a positive clone, Q represents a negative clone, and I represents an inhibitor.

Proof:

(a) Note that for any $S \subseteq R$, $t^S(P) > 0$. There exists an $S \subseteq R$ which contains exactly all the inhibitors. For that S, $t^S(P) = 0$ since there is no reason for P to be in a negative pool.

(b) Since M is $(d + s)$-disjunct, the union of $d + s$ columns in M must not contain any other column. Thus a negative clone Q has at least one element not covered by the up-to-d positive clones and all the clones in S. It results in $t^*(Q) > 1$.

(c) Assume that M does not have any isolated column. A column is isolated if there is a row containing only one 1-entry at the intersection with that column. This assumption is valid since in the pooling designs, each pool should contain more than one clone. Since M is $(d + s)$-disjunct and M does not have any isolated column, M is also (s, d)-disjunct. Consider an inhibitor I and a subset $S \subseteq R$. If $I \in S$, then $t^S(I) = \infty$. Otherwise, an inhibitor I has at least d 1-entries not covered by all the clones in S. Thus $t^*(I) \geq d$.

Algorithm 1 NEGATIVE-POOLS (*M*, *V*, *S*, *A*)

change V to V'' by changing every negative pool containing a clone in S to a positive pool
if $A \in S$ **then**
  $t^S(A) = \infty$
else
  $t^S(A) = \#$ of negative pools in V'' containing A
end if
return $t^S(A)$

Based on the above lemma, we can identify all positive clones *P* by finding those that have $t^*(P) = 0$ in the case where it is error-free.

Now let us consider the IET model. In this model, a pooling design must have an error correcting property. A *d*-separable matrix *M* is said to be *e*-error-correcting if the Hamming distance between two union of at most *d* columns is at least $2e + 1$.

Algorithm 2 DECODING-ALGORITHM (*M*, *V*, *n*, *d*, *s*, *e*)

1: **for** every set E of at most e pools **do**
2: change V to V' by changing the test outcomes of pools in E
3: compute set $R = \{$ clones not appearing in a positive pool $\}$
4: **for** every clone A **do**
5: $t^*(A) = min_{S \subseteq R}$ **NEGATIVE-POOLS**(M, V', R, A)
6: **end for**
7: set $D = \{A \mid t^*(A) = 0\}$
8: set $B = \{A \mid t^*(A) \geq d\} \cap R$
9: **if** (1) $Union(D) \subseteq V'$ &&
  (2) there exists $O \subseteq B$ with $|O| \leq s$ such that $Union(D \cup O) \supseteq V'$ **then**
10: **return** D
11: **end if**
12: **end for**

Without the presence of inhibitors, we can choose the *d* columns whose union is within Hamming distance *e* from the test outcome vector be the set of all positive clones. Thus we can propose a decoding algorithm for the IET model as described in Algorithm 2.

The proposed algorithm consists of two for loops. The outer for loop is to identify the errors whereas the inner loop is to identify the inhibitors. In particular, at line 2, we assume that experimental errors occur on pools in set *E*. The procedure NEGATIVE-POOLS is called at line 5 to identify the set of clones with $t^*(A) = 0$. Note that at this procedure, we assume that the set *S* is a set of inhibitors. The details of this algorithm can be seen in Algorithm 2.

Theorem 1: *The proposed algorithm outputs a set of all positive clones if matrix M is* $(d + s)$*-disjunct and* $(\overline{d + s}; 2e + 1)$*-separable.*

Proof: To prove this theorem, we prove these following three claims:

(1) the output D satisfies the following property: (*) After deleting at most s columns and rows covered by these s columns, the Hamming distance $H(\text{Union}(D), V) \leq e$.

(2) if E is exactly the set of errors tests, then D must be reached

(3) there exists exactly one D satisfying property (*).

Claim (1) follows from the conditions (1) and (2) at the if statement (line 9). Note that we compare the Hamming distance between Union (D) and the original test outcome vector V.

Claim (2) follows from Lemma 2. Also note that by Lemma 2, output D consists of all the positive clones.

Claim (3) follows from the properties of the $(\overline{d+S};\ 2e+1)$ separable matrix M, which implies that deleting at most s columns and rows covered by them, the Hamming distance between two unions of at most d columns is at least $2e + 1$.

4 Discussions

In this paper, we study the decoding algorithms in pooling designs with the presence of inhibitors and error-tolerance. We believe that there does not exist a decoding algorithm for $(d + s)$-disjunct and $(\overline{d+S};\ 2e+1)$-separable matrix running in a polynomial time with respect to n, d, t, e and s unless $NP = P$. If we strengthen the requirements of such a matrix to the $(d + s, 2e + 1)$-disjunct matrix, we may reduce the time complexity. However, the number of tests would increase as discussed in Section 2. The study also showed that the time complexity in this model is still expensive, i.e., $O(n^s)$ (Du and Hwang, 2006).

References

De Bonis, A. and Vaccaro, U. (1998) 'Improved algorithms for group testing with inhibitors', *Inform. Proc. Lett*., Vol. 67, pp.57–64.

Du, D.Z. and Hwang, F.K. (1999) *Combinatorial Group Testing and its Applications*, 2nd ed., World Scientific, Singapore.

Du, D.Z. and Hwang, F.K. (2006) *Pooling Designs: Group Testing in Molecular Biology*, World Scientific, Singapore.

Dychkov, A.G., Macula, A.J., Torney, D.C. and Vilenkin, P.A. (2001) 'Two models of nonadaptive group testing for designing screening experiments', *Proc. 6th Int. Work Shop on Model Oriented Designs and Analysis*, Physica-Verlag, New York, pp.63–75.

Farach, M., Kannan, S., Knill, E. and Muthukrishnan, S. (1997) 'Group testing problem with sequences in experimental molecular biology', *Proc. Compression and Complexity of Sequences*, pp.357–367.

Gao, H., Hwang, F.K., Thai, M.T., Wu, W. and Znati, T. (2006) 'Construction of $d(H)$ – disjunt matrix for group testing in hypergraphs', *J. Combinatorial Optimization*, Vol. 12, No. 3, pp.297–301.

Hwang, F.K. and Liu, Y.C. (2003) 'Error-tolerant pooling designs with inhibitors', *J. Comput. Biol.*, Vol. 10, No. 2, pp.231–236.

Li, Y., Thai, M.T., Liu, Z. and Wu, W. (2005) 'Protein-protein interaction and group testing in bipartite graphs', *International Journal of Bioinformatics Research and Applications (IJBRA)*, Vol. 1, No. 4, pp.414–419.

Macula, A.J., Rykov, V.V. and Yekhanin, S. (2004) 'Trivial two-stage group testing for complexes using almost disjunct matrices', *Disc. Appl. Math*., Vol. 137, No. 97, p.107.

Macula, A.J., Torney, D.C. and Villenkin, P.A. (2000) 'Two-stage group testing for complexes in the presence of errors', *DIMACS Sries in Disc. Math. and Theor. Comput. Sci.*, Vol. 55, pp.145–157.

Marathe, M.V., Percus, A.G. and Torney, D.C. (2000) *Combinatorial Optimization in Biology*, Manuscript: http://www.c3.lanl.gov/~percus/Research/Bio/.

Ngo, H.Q. and Du, D-Z. (2000) 'A survey on combinatorial group testing algorithms with applications to DNA library screening', *Discrete Mathematical Problems with Medical Applications*, New Brunswick, NJ, 1999, pp.171–182; DIMACS Ser. *Discrete Math. Theoret. Comput. Sci., Amer. Math. Soc.*, Vol. 55, Providence, RI.

Torney, D.C. (1999) 'Sets pooling designs', *Ann. Combin*., Vol. 3, pp.95–101.

Triesch, E. (1996) 'A group testing problem for hypergraphs of bounded rank', *Disc. Appl. Math.*, Vol. 66, pp.185–188.