A Construction of Pooling Designs with Some Happy Surprises

A. D'yachkov* Frank Hwang † Antony Macula*‡

Pavel Vilenkin§ Chih-wen Weng¶

February 9, 2007(revised)

*A. D'yachkov, Department of Probability Theory, Faculty of Mechanics and Mathematics, Moscow State University, Moscow, 119992, Russia, email: agd-msu@yandex.ru

†F. Hwang, Department of Applied Mathematics, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 30050, Taiwan, email: fhwang@math.nctu.edu.tw. This research was partially supported by a Republic of China NSC grant 92-2115-M-009-014.

‡A. Macula(Corresponding author), Department of Mathematics, College at Geneseo, State University of New York, Geneseo, NU 14454, USA, email: macula@geneseo.edu. This research was partially supported by NSF-DMS 0107179.

§P. Vilenkin, Department of Probability Theory, Faculty of Mechanics and Mathematics, Moscow State University, Moscow, 119992, Russia, email: pavvil@yandex.ru

¶C. Weng(Corresponding author), Department of Applied Mathematics, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 30050, Taiwan, email: weng@math.nctu.edu.tw, fax: +886-3-5724679. This research was partially supported by a Republic of China NSC grant 91-2005-M-009-008.

Abstract

The screening of data sets for "positive data objects" is essential to modern technology. A (group) test that indicates whether or not a positive data object is in a specific subset or pool of the data set can greatly facilitate the identification of all the positive data objects. A collection of tested pools is called a pooling design. Pooling designs are standard experimental tools in many biotechnical applications. In this paper, we use the (linear) subspace relation coupled with general concept of a "containment matrix" to construct pooling designs with surprisingly high degrees of error-correction (detection.) Error-correcting pooling designs are important to biotechnical applications where error rates often are as high as 15%. What is also surprising is that the rank of the pooling design containment matrix is independent of the number of positive data objects in the data set.

Keywords: pooling designs, error-correction.

1 Introduction

The screening of biological sets of objects, e.g., blood samples, cells, clones, macromolecules, is an essential but often laborious aspect of modern biotechnology. In a few instances, the screening of large libraries, e.g., peptide, cDNA, monoclonal antibody, for a relatively few number of positive objects has become a routine experimental procedure. See [2]. Similar approaches have also been proposed for contig sequencing [8], determination of exon

boundaries in eukaryotic genes [17], detecting gene complex[18], micro-array quality control [3] and disease gene mapping [7].

Whenever the objective is to find "needles in a haystack" a test indicating whether at least one needle is in a specific part of the haystack can greatly facilitate the isolation of the "needles". Such tests are called *binary group tests* and the general mathematical method behind the identification of the "needles" using such tests is called *group testing* [6]. If we have a finite ground set or *population* containing elements that can be uniquely characterized as positive or negative, we refer to the collection of *positive elements* as the *positive subset* P. In the abstract group testing problem, P must be identified by performing 0, 1 tests on subsets of the population.

One applied aim is to consider screening situations where we have a biological set of objects containing a relatively small number data points (e.g., clones) which have a measurable attribute or function that can characterize them as "positive". This subcollection is initially unknown to the experimenter and it is the object of the search. A group of biological objects taken from a larger set of objects is called a *pool*. A *pool assay* is a 0, 1 test to determine if at least one member of the pool is positive. The practical goal here is to determine a large portion of P from the pool assays. The collection of pools taken from a biological set of objects is called a *pooling design*.

The following comes from [2].

"Much of the current effort of the Human Genome project involves the screening of a large recombinant DNA libraries to isolate clones containing

a particular DNA sequence." "This screening is important for disease-gene mapping and also for large-scale clone mapping." "More generally, efficient screening techniques can facilitate a broad range of basic and applied biological research."

For example, using probes to screen DNA libraries of clones fits the group testing paradigm in the following way: The population is the DNA library which consists of thousands of separate recombinant DNA clones each of which represents some contiguous piece of a contiguous superpiece of DNA. A unique, identifiable, predetermined, and contiguous DNA subpiece is called a *sequenced tagged site* (STS). A clone is called *positive* for an STS if it contains that STS. A pool is a subset of the clones that are mixed together and tested by exposing the entire group to a chemical probe. A pool is *labeled* positive for an STS if the probe chemically indicates its presence. In other words, if the tests are error-free, then a pool is labeled positive for an STS if and only if that pool contains at least one clone that contains that STS.

Generally because bioinformatic applications are often automated, parallel rather than sequential screening methods are generally preferred. See [6] for other screening cost factors. Long before the advent of bioinformatics, consideration of analogous factors in other testing, screening, or coding situations lead to the development of *nonadaptive group testing*. See [4]. In NGT, one must decide exactly which pools to test before any testing occurs. A NGT algorithm is sometimes referred to as a *one-stage* algorithm. A two-stage algorithm is a nearly nonadaptive algorithm. In a trivial two-stage algorithm, all non-trivial pools occur in the first stage. After the first stage

is complete, one has a set the *candidate positives*. In the second stage, each candidate positive is individually tested to see whether or not it is an actual positive.

When screening biological sets errors almost always occur during the testing procedure. This paper addresses a new class of pooling designs that can cope with large numbers of errors.

2 d-disjunct matrices as nonadaptive pooling design models

We will use the terminology of clone library screening for convenience. Suppose there are n clones including at most d positive ones (others are negative). A (group) test is applicable to an arbitrary subset of clones with two possible outcomes: a negative outcome indicates all clones in the subset are negative, and a positive outcome indicates otherwise. A pooling design is a specification of all tests so that they can be performed simultaneously with the goal to identify all positive clones with a small number of tests. A pooling design M can be represented by a binary incidence matrix where the columns represent clones, the rows represent tests, and $m_{ij} = 1$ if and only if clone j is contained in the subset of test i.

Suppose M has t rows. Then the t outcomes can also be represented by a t-vector $V = (v_1, \cdots, v_t)^t$, where $v_i = 1$ if and only if the outcome of test i is positive ($v_i = 0$ otherwise). Note that V is the boolean sum of the set of positive clones. Therefore it is convenient to view a column vector C as

a subset S of the base set $\{1, 2, \cdots, t\}$, where $i \in S$ if and only if C has an 1-entry in row i. Then we can say that V is the union of the set of positive clones.

M is called *d-disjunct* if no union of any d columns covers another column. A d-disjunct matrix not only identifies the up-to-d positive clones, but with a simple decoding. Namely, a clone is positive if and only if it (as a column) is contained by V. This is because a negative clone (column) has at least one row not covered by the union of the up-to-d positive clones; such a row then has a negative outcome which identifies the clone as negative. The notion of d-disjunctness was first raised by Kautz and Singleton[11] in the study of superimposed codes. It was also studied by Erdös, Frankl and Füredi[5] under the name of d-cover-free family in extremal set theory. d-disjunct matrices have become the most important tool in the construction of deterministic pooling designs. Although many constructions have been proposed, the existence of d-disjunct matrices is still sparse.

Macula [13] proposed a novel way of constructing d-disjunct matrices which uses the containment relation in a structure. More specifically, let $[m] := \{1, 2, \cdots, m\}$ be the base set. Then each of the n columns is labeled by a (distinct) k subset of $[m]$, assuming $n \leq \begin{pmatrix} m \\ k \end{pmatrix}$, and each of the $\begin{pmatrix} m \\ d \end{pmatrix}$ rows is labeled by a (distinct) d-subset of $[m]$, where $d < k < m$. $m_{ij} = 1$ if and only if the label of row i is contained in the label of column j. He proved that M is d-disjunct.

Huang and Weng [9] generalized the construction to arbitrary atomic semi-lattice where the elements can be ranked. Again, label the columns by a subset of the rank k elements and label the rows by all rank d elements, $d < k$, then M is d-disjunct.

Ngo and Du [16] further extended the construction to some geometric structures like simplicial complexes, and some graph properties like matchings. It is safe to say the "containment matrix" method has opened a new door for constructing d-disjunct matrices from many mathematical structures. However, the basic result in all these constructions is invariably that, to obtain a d-disjunct matrix, use all rank d elements for rows.

One practical problem with this type of construction is that a large n forces d to be large. Then the number of tests could be too large as there are too many rank d elements. This led Macula [15] to propose using the rank 2 elements for rows, regardless of the real d. He showed that while there is no guarantee to identify all positive clones, the probability of success is still satisfactory when d does not deviate too much from 2. Ngo and Du made a similar comment.

In this paper, we show that the containment matrix which use rank r of elements for rows has the degree d of disjunctness, where r can be much less than d. In fact r can be any number from 1 to $k-1$ (k is the lever for columns), while $d \leq q^r$ for some constant q. This is the first happy surprise. Since we can choose $r = 1$, we also have better control on the number of tests.

3 The error-correcting capability

Biological experiments are notorious for producing erroneous outcomes. Therefore it would be wise for pooling designs to allow some outcomes to be affected by errors. Macula[14] proposed the notion of d^e-disjunct to reflect the error-correcting capability of a d-disjunct matrix. A d-disjunct matrix is d^e-disjunct if a column has at least $e+1$ 1-entries not covered by the union of any other d columns. d^0-disjunct would then be the regular d-disjunct.

In [10] it was misclaimed that a d^e-disjunct matrix can correct e errors. The argument was that if we try all subsets E of up to e elements as the candidate set of errors and adjust the outcome set V to $V \cup E$, then when E is the true error set, a positive clone C must be contained in $V \cup E$. On the other hand, a negative clone C has at least $e+1$ 1-entries not covered by the set of up to d positive clones, i.e., C has at least $e+1$ negative outcomes. At most e of them can be converted to positive by errors, thus at least one negative outcome is not covered by V. The problem of this argument is that we need to show that C has at least one negative outcome not covered by $V \cup E$. The following is a counterexample.

Example 3.1. $d = 2$, $e = 1$. *Column* 1 *is the only positive clone while* v_3 *is*

an error.

$$M = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \qquad V = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

When $E = \{4\}$, $V \cup E = (1, 1, 1, 1, 0, 0)^t$ *covers column* 1 *and* 2.

Thus the correct version should be

Theorem 3.2. *A* d^{2e}*-disjunct matrix is* e*-error-correcting.*

Proof. For a positive clone C the argument is as before that there exists a candidate set E such that $C \subseteq V \cup E$.

A negative clone C has at least $2e + 1$ 1-entries not covered by the set D of up to d positive clones, hence at least $2e + 1$ negative outcomes. e of them may be converted to positive by errors and another e of them by E, but at least one negative outcome is not covered by $V \cup E$. □

For the reason that a d^e-disjunct matrix is not really e-correcting, and also that d^0-disjunct= d-disjunct, is kind of uncustomary, we propose to use the term d^z-disjunct while z is the minimum (over C) number of 1-entries in C not covered by the union of any other d columns. Theorem 3.2 then can be restated as

Theorem 3.3. *A* d^z*-disjunct matrix can detect* $z - 1$ *errors and correct* $\lfloor \frac{z-1}{2} \rfloor$ *errors.*

In particular, a d-disjunct matrix has no error-tolerance.

If an extra round of confirmatory tests is allowed, then a d^z-disjunct matrix can indeed correct $z-1$ errors. First, we need a lemma. Let $H(X,Y)$ denote the Hamming distance between two binary vectors X, Y of the same length.

Lemma 3.4. *Let M be a d^z-disjunct matrix and let S_1 S_2 be two distinct subsets of columns with $|S_1| \leq d, |S_2| \leq d$. Let U_i be the union of the set S_i for each $i = 1, 2$. Then*

1. *$H(U_1, U_2) \geq z$ if either $S_1 \subseteq S_2$ or $S_2 \subseteq S_1$;*

2. *$H(U_1, U_2) \geq 2z$ if otherwise.*

Proof. These are trivial by using the d^z-disjunct property. □

Theorem 3.5. *A d^z-disjunct matrix corrects $z-1$ errors with an extra round of at most d confirmatory tests.*

Proof. Take all subsets S of columns of M with $|S| \leq d$ and $H(U,V) \leq z-1$, where U is the union of S. Let S_1, S_2 be two such sets. The $H(U_1, U_2) \leq 2(z-1) < 2z$. By Lemma 3.4, either $S_1 \subseteq S_2$ or $S_2 \subseteq S_1$. Therefore the set $\{S\}$ is a chain. Hence $\{S\}$ has at most d members. Since $H(D,V) \leq z-1$, $D \in \{S\}$, D can be identified by testing at most d columns in the maximal chain of $\{S\}$. □

Not many constructions of d^z-disjunct matrices have been known. Macula [14], and also see [10], gave a construction for d^4, and recently Ngo and Du gave a construction for d^{d+1}. We will show that the construction delivering

the first happy surprise mentioned in section 1 not only yields d-disjunct matrices, but also d^z-disjunct matrices with the z-value much greater than 4 or $d+1$. This is the second happy surprise.

4 The construction

Consider the m-dimensional space, or simply m-space, of $GF(q)$ where q is a prime or a prime power. Let $\left[\begin{array}{c} m \\ k \end{array} \right]_q$ denote the number of k-dimensional subspaces, or simply k-space. It is well known [12, p. 291]

Lemma 4.1.

$$\left[\begin{array}{c} m \\ k \end{array} \right]_q = \frac{(q^m-1)(q^{m-1}-1)\cdots(q^{m-k+1}-1)}{(q^k-1)(q^{k-1}-1)\cdots(q-1)}$$

and

$$\left[\begin{array}{c} m \\ k \end{array} \right]_q = \left[\begin{array}{c} m \\ m-k \end{array} \right]_q .$$

Definition 4.2. Fix integers $1 \leq r < k < m$. Let $M(m,k,r)$ be the 01-matrix by taking all k-spaces (from an underlying m-space) as columns and all r-spaces as rows. $M(m,k,r)$ has a 1 in row i and column j if and only if i is contained in j.

$M(m,k,r)$ was first studied in [19] from a linear algebra point of view and in [16] from a pooling design point of view. $M(m,k,r)$ is easily checked to be a ranked atomic semi-lattice, thus the matrix is r-disjunct, hence [9]

d^z-disjunct for some $1 \leq d \leq r$ and

$$z = \begin{bmatrix} k-d \\ r-d \end{bmatrix}_q .$$

Note that the construction still requires the row rank r being at least as large as the upper bound d of the number of positive clones. We now show that r can be much less than d. First, we give a lemma.

Lemma 4.3.

$$\begin{bmatrix} k \\ r \end{bmatrix}_q - \begin{bmatrix} k-1 \\ r \end{bmatrix}_q = q^{k-r} \begin{bmatrix} k-1 \\ r-1 \end{bmatrix}_q \qquad (0 \leq r < k).$$

Proof.

$$\begin{aligned} & \begin{bmatrix} k \\ r \end{bmatrix}_q - \begin{bmatrix} k-1 \\ r \end{bmatrix}_q \\ = \; & \frac{(q^k-1)(q^{k-1}-1)\cdots(q^{k-r+1}-1)}{(q^r-1)(q^{r-1}-1)\cdots(q-1)} - \frac{(q^{k-1}-1)(q^{k-2}-1)\cdots(q^{k-r}-1)}{(q^r-1)(q^{r-1}-1)\cdots(q-1)} \\ = \; & \frac{(q^k-1)-(q^{k-r}-1)}{q^r-1} \cdot \frac{(q^{k-1}-1)\cdots(q^{k-r+1}-1)}{(q^{r-1}-1)\cdots(q-1)} \\ = \; & q^{k-r} \begin{bmatrix} k-1 \\ r-1 \end{bmatrix}_q . \end{aligned}$$

□

Theorem 4.4. *Suppose* $k - r \geq 2$ *and set* $p := \dfrac{q(q^{k-1}-1)}{q^{k-r}-1} + 1$. *Then* $M(m,k,r)$ *is* d^z*-disjunct for* $1 \leq d < p$ *and*

$$z = q^{k-r} \begin{bmatrix} k-1 \\ r-1 \end{bmatrix}_q - (d-1)q^{k-r-1} \begin{bmatrix} k-2 \\ r-1 \end{bmatrix}_q .$$

Proof. Let $C, C_1, \cdots, C_d$ be $d+1$ distinct columns (k-spaces) of M. By Lemma 4.1, C contains $\begin{bmatrix} k \\ r \end{bmatrix}_q$ r-spaces. To obtain the maximum elements in

$$C \cap \bigcup_{i=1}^{d} C_i = \bigcup_{i=1}^{d} (C \cap C_i),$$

we may assume that each C_i intersects C at a $(k-1)$-space. Then each C_i covers $\begin{bmatrix} k-1 \\ r \end{bmatrix}_q$ r-spaces of C. However, the coverage of each pair of C_i and C_j overlaps at a $(k-2)$-space. Therefore only C_1 covers the full $\begin{bmatrix} k-1 \\ r \end{bmatrix}_q$ r-spaces, while each of $C_2, \cdots, C_d$ can cover a maximum of $\begin{bmatrix} k-1 \\ r \end{bmatrix}_q - \begin{bmatrix} k-2 \\ r \end{bmatrix}_q$ r-spaces not covered by C_1. Consequently the number of r-spaces of C not covered by $C_1, \cdots, C_d$ is at least

$$\begin{aligned} z &= \begin{bmatrix} k \\ r \end{bmatrix}_q - \begin{bmatrix} k-1 \\ r \end{bmatrix}_q - (d-1)\left(\begin{bmatrix} k-1 \\ r \end{bmatrix}_q - \begin{bmatrix} k-2 \\ r \end{bmatrix}_q \right) \\ &= q^{k-r} \begin{bmatrix} k-1 \\ r-1 \end{bmatrix}_q - (d-1) q^{k-r-1} \begin{bmatrix} k-2 \\ r-1 \end{bmatrix}_q . \end{aligned}$$

Note that for $M(m,k,r)$ to be d^z-disjunct, z must be positive, which implies

$$d < \frac{q^{k-r} \begin{bmatrix} k-1 \\ r-1 \end{bmatrix}_q}{q^{k-r-1} \begin{bmatrix} k-2 \\ r-1 \end{bmatrix}_q} + 1,$$

or $d < p$. □

Suppose $d \leq q+1$. The following corollary shows the above z is optimal.

Corollary 4.5. *Suppose $k - r \geq 2$ and suppose*

$$1 \leq d \leq \begin{cases} q+1, & \text{if } r > 1; \\ q, & \text{if } r = 1. \end{cases} \tag{4.1}$$

Then $M(m,k,r)$ is not d^{z+1}-disjunct, where z is as in Theorem 4.4.

Proof. We prove this by showing that a maximum coverage of r-spaces in the proof of Theorem 4.4 is obtained. We reverse the arguments. Let U be a $(k-2)$-space contained in C. Then the number of $(k-1)$-spaces between U and C is

$$\begin{bmatrix} k-(k-2) \\ k-1-(k-2) \end{bmatrix}_q = q+1.$$

We choose d distinct ones among them, say T_i $(1 \leq i \leq d)$. Here $d \leq q$ if $r = 1$ to ensure $d < p$ in the assumption of Theorem 4.4. For each T_i, we choose a k-space C_i such that $C \cap C_i = T_i$. Hence each pair of C_i and C_j overlaps at the same $(k-2)$-space U. □

Lemma 4.6. *Suppose $r \leq \dfrac{k}{2}$. Then with referring to the definition of p in Theorem 4.4, $d = q^r$ is the largest integer less or equal to p.*

Proof. Note that $q^{r-1} < q^r \leq q^{k-r}$. Hence

$$\begin{aligned} p - q^r &= \frac{q(q^{k-1}-1)}{q^{k-r}-1} - q^r \\ &= \frac{q^k - q - q^k + q^r}{q^{k-r}-1} \\ &= \frac{q(q^{r-1}-1)}{q^{k-r}-1} \\ &< \frac{qq^{r-1}}{q^{k-r}} \\ &\leq 1. \end{aligned}$$

Then $p-1 < q^r \leq p$. □

Corollary 4.7. *Suppose $k-r \geq 2$ and $d = q^r$. Then $M(m,k,r)$ is d^z-disjunct with*

$$z = \left[\begin{array}{c} k-1 \\ r-1 \end{array} \right]_q + (q^r-1) \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q .$$

Proof. Setting $d = q^r$ in Theorem 4.4 and referring to Lemma 4.1, Lemma 4.3,

$$\begin{aligned}
z &= q^{k-r} \left[\begin{array}{c} k-1 \\ r-1 \end{array} \right]_q - (q^r-1)q^{k-r-1} \left[\begin{array}{c} k-2 \\ r-1 \end{array} \right]_q \\
&= \left[\begin{array}{c} k \\ r \end{array} \right]_q - \left[\begin{array}{c} k-1 \\ r \end{array} \right]_q - (q^r-1)(\left[\begin{array}{c} k-1 \\ r \end{array} \right]_q - \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q) \\
&= \left[\begin{array}{c} k \\ r \end{array} \right]_q - q^r \left[\begin{array}{c} k-1 \\ r \end{array} \right]_q + (q^r-1) \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q \\
&= \left[\begin{array}{c} k \\ k-r \end{array} \right]_q - q^{k-(k-r)} \left[\begin{array}{c} k-1 \\ r \end{array} \right]_q + (q^r-1) \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q \\
&= \left[\begin{array}{c} k-1 \\ k-r \end{array} \right]_q + (q^r-1) \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q \\
&= \left[\begin{array}{c} k-1 \\ r-1 \end{array} \right]_q + (q^r-1) \left[\begin{array}{c} k-2 \\ r \end{array} \right]_q .
\end{aligned}$$

□

When $r = 1$, the z in Theorem 4.4 is in a neater form.

Corollary 4.8. *Suppose $k \geq 3$, $d \leq q$ and $z = q^{k-2}(q-d+1)$. Then $M(m,k,1)$ is d^z-disjunct, but is not d^{z+1}-disjunct.*

Proof. Setting $r = 1$ in the z formula of Theorem 4.4, we obtain

$$\begin{aligned} z &= q^{k-1} \begin{bmatrix} k-1 \\ 0 \end{bmatrix}_q - (d-1)q^{k-2} \begin{bmatrix} k-2 \\ 0 \end{bmatrix}_q \\ &= q^{k-2}(q-d+1). \end{aligned}$$

The second statement follows from Corollary 4.5. □

Example 4.9. *Fix* $q = 5$. *Then* $M(8, 4, 1)$ *is a* 5^{25}*-disjunct matrix with* 97656 *rows and* 200525284806 *columns. This means that in our method to identify* 5 *positives from* 2×10^{11} *clones, at most* 10^5 *pools are necessary and* 25 *errors are allowed.*

References

[1] Barrllot, E., Lacroix, B. and Cohen, D. 1995. Theoretical analysis of library screening using an n-dimensional pooling strategy. *Nucleic Acids Res.* 19:6241-6247.

[2] Bruno, W. J., Knill, E., Balding, D. J., Bruce, D. C., Doggett, N. A., Sawhill, W. W., Stallings, R. L., Whittaker C. C., and Torney E. C. 1995. Effective pooling designs of library screening. *Genomics* 26:21-30.

[3] Colburn, C., Ling, A., Tompa, M. 2002. Construction of optimal quality control ogilio arrays, *Bioinformatics*, 18, no. 4, 529-535.

[4] Du, D. and Hwang, F. K. 2000. *Combinatorial Group Testing and Its Applications*, 2nd Ed., World Scientific, Singapore.

[5] Erdös, P., Frankl, and Füredi, D. 1985. Families of finite sets in which no set is covered by the union of r others. *Israel J. Math.* 51:79–89.

[6] Farach, et al. 1997. Group testing problems with sequences experimental molecular biology, *Proceedings of Compression and Complexity of Sequences, 1997, B. Carpentieri, et al. (Eds.)* IEEE Press, 357-367.

[7] Flodman, P., Macula, A., Spence, A. and Torney, D. 2001. A new data mining technique for the analysis of simulated genetic data. *Proceedings of Genetic Analysis Workshop 12, Wiley-Liss, 2001*; *Genet Epidemiol (Genetic epidemiology.)* 21 Suppl 1: S390-5 (2001.)

[8] Grebinski, V. and Kucherov, G. 1998. Reconstructing a Hamiltonian cycle by querying the graph:application to DNA physical mapping. *Disc. Appli. Math.*. 88:147-165.

[9] Huang, T. and Weng, C. 2004. Pooling Spaces and Non-Adaptive Pooling Designs. *Disc. Math.* 282(1-3):163-169.

[10] Hwang, F. K. 2003. On Macula's Error-Correcting Pool Designs. *Disc. Math.* 267:311-314.

[11] Kautz, W. H. and Singleton, R. C. 1964. Nonadaptive binary superimposed codes. *IEEE Trans. Inform. Theory* 10:363–377.

[12] van Lint, J. H., and Wilson, R. M. 1992. *A Course in Combinatorics*. Cambridge, Victoria.

[13] Macula, A. J. 1996. A simple construction of d-disjunct matrices with certain constant weights. *Disc. Math.* 162:311–312.

[14] Macula, A. J. 1996. Error-correcting nonadaptive group testing with d^e-disjunct matrices. *Disc. Appl. Math.* 80:217-222.

[15] Macula, A. J. 1999. Probabilistic nonadaptive and two-stage group testing with relatively small pools and DNA library screening. *J. Comb. Opt.* 2:385–397.

[16] Ngo, H., and Du, D. 2002. New Constructions of Non-Adaptive and Error-Tolerance Pooling Designs. *Disc. Math.* 243:161–170.

[17] Pevzner, P. A. 2000. *Computational Molcular Biology: an algorithmic approach.* MIT Press, Mass Sec. 9.5-9.6.

[18] Torney, D. C. 1999. Sets pooling designs. *Ann. Combin.* 3:95–101.

[19] Yakir, A. 1993. Inclusion matrix of k vs 1 affine subspaces and a permutation module of the general affine group. *J. Combin. Theory, Ser. A* 63:301–317.