J Comb Optim (2008) 15: 77–94
DOI 10.1007/s10878-007-9085-1

New combinatorial structures with applications to efficient group testing with inhibitors

Annalisa De Bonis

Published online: 20 July 2007

Abstract *Group testing with inhibitors* (GTI) is a variant of classical group testing where in addition to positive items and negative items, there is a third class of items called *inhibitors*. In this model the response to a test is YES if and only if the tested group of items contains at least one positive item and no inhibitor. This model of group testing has been introduced by Farach et al. (Proceedings of compression and complexity of sequences, pp 357–367, 1997) for applications in the field of molecular biology. In this paper we investigate the GTI problem both in the case when the exact number of positive items is given, and in the case when the number of positives is not given but we are provided with an upper bound on it. For the latter case, we present a lower bound on the number of tests required to determine the positive items in a completely nonadaptive fashion. Also under the same hypothesis, we derive an improved lower bound on the number of tests required by *any* algorithm (using any number of stages) for the GTI problem.

As far as it concerns the case when the exact number of positives is known, we give an efficient trivial two-stage algorithm. Instrumental to our results are new combinatorial structures introduced in this paper. In particular we introduce generalized versions of the well known superimposed codes (Du, D.Z., Hwang, F.K. in Pooling designs and nonadaptive group testing, 2006; Dyachkov, A.G., Rykov, V.V. in Probl. Control Inf. Theory 12:7–13, 1983; Dyachkov, A.G., et al. in J. Comb. Theory Ser. A 99:195–218, 2002; Kautz, W.H., Singleton, R.R. in IEEE Trans. Inf. Theory 10:363–377, 1964) and selectors (Clementi, A.E.F, et al. in Proceedings of symposium on discrete algorithms, pp. 709–718, 2001; De Bonis, A., et al. in SIAM J Comput. 34(5):1253–1270, 2005; Indyk, P. in Proceedings of symposium on discrete algorithms, pp. 697–704, 2002) that we believe to be of independent interest.

A. De Bonis (✉)
Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84084 Fisciano (SA), Italy
e-mail: debonis@dia.unisa.it

Keywords Group testing algorithms · Superimposed codes · Selectors · Pooling designs · Computational molecular biology

1 Introduction

The group testing problem asks to determine the *positive* members of a set of objects $\mathcal{O}$ by performing tests on subsets (*pools*) of the given set $\mathcal{O}$. In classical group testing, a test yields a YES response if the tested subset contains one or more positive objects, and a NO response otherwise. The goal is to identify all positive items by using as few tests as possible. Group testing originated in the area of chemical analysis as a possible technique for mass blood testing (Dorfman 1943). Group testing like problems occur in various contexts, ranging from data compression (Hong and Ladner 2002), efficient access to storage systems (Kautz and Singleton 1964), conflict resolution algorithms for multiple-access systems (Berger et al. 1984; Wolf 1985), quality control in product testing (Sobel and Groll 1959), data gathering in sensor networks (Hong and Scaglione 2004), and sequential screening of experimental variables (Li 1962). Recently group testing has proved to be extremely useful in the area of Computational Molecular Biology where its primary application is for screening library of clones with hybridization probes (Barillot et al. 1991; Bruno et al. 1995), and sequencing by hybridization (Margaritis and Skiena 1995; Pevzner and Lipshutz 1994). We refer readers interested in this issues to Balding et al. (1996); Du and Hwang (2000, 2006); Farach et al. (1997); Ngo and Du (2000). The variety of situations in which group testing is applied motivates many extensions of the classical paradigm. Motivated by molecular biology applications, Farach et al. (1997) introduced an interesting variation of the classical group testing problem where, in addition to the category of the positive samples and the one of the negative samples, there is a third class of samples called *inhibitors*. In this model a pool tests positive if and only if it contains one or more positive items and no inhibitor. In the biological setting inhibitors correspond to spoiled samples which make the outcomes of the tests meaningless. Following De Bonis and Vaccaro (1998), we will refer to this search model as *Group Testing with Inhibitors* (GTI). Biological applications not only call for more complex querying models but also require search strategies to fulfill different criterions of performance. While the number of tests is typically adopted as a measure of efficiency, there are many biological experiments in which the time spent in preparing the pools during the testing procedure is also a crucial issue. In this setting, screening one pool at the time is far more expensive than screening many pools in parallel. For this reason the most preferable search strategies for screening are nonadaptive strategies. Nonadaptive algorithms, to which we will refer also as *one-stage algorithms*, are group testing procedures in which all tests are specified in advance without knowing the outcomes of other tests. Typically, nonadaptive strategies are much more costly than adaptive group testing strategies, i.e. algorithms in which the tests are performed one by one, and the pool for the current test is constructed by looking at the outcomes of previous tests. As an instance, it is known that nonadaptive algorithms for classical group testing are essentially equivalent to *superimposed codes* (Dyachkov and Rykov 1983; Erdös et al. 1985; Kautz and Singleton 1964) and as a consequence must use a number of tests $\Omega((p^2/\log p)\log n)$, where p is the maximum number of positives and

$n = |\mathcal{O}|$, whereas the best known nonadaptive algorithms for classical group testing use $O(p^2 \log n)$ tests. These asymptotic bounds are very far from the information theoretic lower bound of $\log \binom{n}{p} = \Omega(p \log(n/p))$ which is attained by adaptive strategies. Due to the high cost of nonadaptive strategies, often people consider using nearly nonadaptive algorithms. Of considerable interest for screening problems is the so called *trivial two-stage algorithm* (Knill 1995). Such an algorithm consists of two completely nonadaptive stages: in the first stage a "small" subset of $\mathcal{O}$ containing all positive items is determined; in the second stage the items in this set are individually tested so as to find those that are positive. In De Bonis et al. (2005) it has been proved that in classical group testing, trivial two-stage algorithms are asymptotically as efficient as the best *fully adaptive* group testing algorithms, that is, algorithms with arbitrarily many stages.

1.1 Previous results and a summary of our contributions

In this paper we address the problem of designing efficient one-stage and trivial two-stage procedures for the GTI problem. Previous work on this model has revealed that the information theoretic lower bound $\Omega(p \log(n/p))$ is not very informative of the number of tests really needed to solve the GTI problem. This is very much different from what happens in the model of classical group testing for which there are procedures that attains the information theoretic lower bound. Indeed, the authors of (De Bonis and Vaccaro 1998) uncovered a relation between group testing procedures for the GTI problem and a certain generalization of superimposed codes (Dyachkov and Rykov 1983), thus showing that the number of tests used by *any algorithm* (using any number of stages) that finds p positives in the presence of r inhibitors is lower bounded by the minimum length of such a code of size n. To give an idea of how costly procedures for the GTI problem are, we mention that the minimum length of such a code of size n is $\Omega(\frac{r^2}{p \log r} \log n)$. The authors of (De Bonis et al. 2005) gave an asymptotically optimal algorithm for the GTI problem that works under the hypothesis that one is given the exact number p of positive items and an upper bound r on the number of inhibitors. An interesting feature of this algorithm is that it consists of four completely nonadaptive stages.

In this paper we will always assume that a search strategy is provided with an upper bound r on the number of inhibitors. As for the number of positives, we will investigate both the case when the exact number of positive items is given, and the case when the number of positives is not given but we are provided with an upper bound on it. Under the latter hypothesis the GTI problem reveals another significant feature that differentiates it from classical group testing. Indeed, it is known that there exist procedures for classical group testing that work under this more difficult hypothesis and that are asymptotically as costly as the best procedures that work under the hypothesis of the exact number of positive items being known. In this paper, we show that the same does not hold for group testing with inhibitors. Indeed, we show that procedures for the GTI problem that work under this more difficult hypothesis require a number of tests which is significantly higher than the lower bound stated in De Bonis and Vaccaro (1998). An algorithm for the GTI problem that works under this more difficult hypothesis was described in De Bonis and Vaccaro (1998). The number of

tests used by this algorithm asymptotically differs by a $\log r$ factor from our lower bound. We remark that the algorithm of De Bonis and Vaccaro (1998) works also in the case when it is not known whether the set $\mathcal{O}$ contains at least one positive item. We show that in this case it is asymptotically optimal.

As far as it concerns the case when we know only an upper bound p on the number of positives, we pose also the question of how costly it would be to solve the GTI problem in a completely nonadaptive fashion. For this case we give a lower bound on the number of tests required by any one-stage procedure. This lower bound asymptotically differs by a $1/\log(p+r)$ factor from the best known upper bound (Dyachkov et al. 2001) on the minimum number of tests required by a one-stage algorithm that solves the GTI problem under the hypothesis that the exact number of positives is not known but one is provided with an upper bound p on it.

In the second part of the paper we turn our attention to the design of procedures that work under the hypothesis of the exact number of positives being given. We present an efficient trivial two-stage algorithm using an asymptotic number of tests that differs from the lower bound in De Bonis and Vaccaro (1998) by a factor of $O(\log r)$, where r is the known maximum number of inhibitors. When the number of positive items is larger than half the number of inhibitors, our two-stage algorithm is asymptotically optimal. Instrumental to these results are new combinatorial structures introduced in this paper. These combinatorial structures are a generalized version of the well known superimposed codes (Dyachkov and Rykov 1983; Dyachkov et al. 2002; Kautz and Singleton 1964). We present upper and lower bounds on the length of our generalized codes. Our upper bound is obtained by a greedy construction of a related combinatorial object that we also introduce here for the first time. This new combinatorial structure, which includes as a particular case our generalized superimposed codes, are a generalization of the (k, m, n)-selectors of De Bonis et al. (2005) and the k-selectors of Chrobak et al. (2000). Superimposed codes and selectors have been applied in a wide range of different fields including among the others, computational molecular biology (Balding et al. 1996; De Bonis et al. 2005; De Bonis and Vaccaro 1998; Du and Hwang 2000, 2006; Ngo and Du 2000), cryptography (Kumar et al. 1999; Stinson et al. 2000b), databases (Kautz and Singleton 1964), pattern matching (Indyk 1997), circuit complexity (Chaudhuri and Radhakrishnan 1996), and broadcasting in radio networks (Chrobak et al. 2000; Clementi et al. 2001). For that reason we believe that the significance of our generalized superimposed codes and selectors, and of the related combinatorial results, goes far beyond the particular issue of this paper.

1.2 Structure of the paper

In Sect. 2 we consider the case when the exact number of positives in not known but we are provided with an upper bound on the number of positives and an upper bound on the number of inhibitors. For this case, we provide an improved lower bound on the number of tests needed to solve the GTI problem in a completely nonadaptive fashion. In Sect. 2 we derive also a lower bound on the number of tests required by any algorithm (using any number of stages) for the GTI problem that improves on the lower bound stated in De Bonis and Vaccaro (1998).

Section 3 contains our main combinatorial results. In this section we define our generalized superimposed codes and selectors and give constructions for both combinatorial objects.

In Sect. 4 we turn our attention to two-stage procedures that solve the GTI problem under the hypothesis that one is provided with the exact number of positive items and an upper bound on the number of inhibitors. We present a trivial two-stage algorithm based on the combinatorial structures introduced in Sect. 3.

2 Dealing with the case when the exact number of positives is not known

In this section we deal with the case when we do not know the exact number of positives but we are only given an upper bound p on it. We recall that we assume that an upper bound r on the number of inhibitors is also provided.

In the next section we investigate one-stage algorithms for the GTI problem that work under the above hypothesis.

2.1 One-stage algorithms

We recall that a one-stage algorithm is a completely nonadaptive algorithm, that is, an algorithm in which all tests must be decided beforehand so as to be performed in parallel. In order to define one-stage algorithms for our problem, we need to introduce some notations and definitions.

A set $C = \{c_1, \ldots, c_n\}$ of n binary vectors of length N is called a *binary code* of size n and length N. Each c_j is called *codeword* and for any i, $1 \leq i \leq N$, $c_j(i)$ denotes the i-th entry of c_j. A binary code C can be represented by an $N \times n$ binary matrix $M = \|c_j(i)\|$, $i = 1, \ldots, N$ and $j = 1, \ldots, n$, having as columns the codewords of C. A binary code will be represented both by the set of its codewords and by the corresponding binary matrix.

In the following we denote by $[n]$ the set of the first n positive integers $\{1, \ldots, n\}$ and assume $\mathcal{O} = [n]$.

An $N \times n$ matrix $M = \|c_j(i)\|$, or equivalently a code $C = \{c_1, \ldots, c_n\}$, defines a one-stage group testing algorithm for a set $\mathcal{O} = [n]$ in the following way. For each $j \in [n]$, let us associate item j to the j-th column c_j of M. For each row i of M, we define the subset $T_i = \{j \in \{1, \ldots, n\} : c_j(i) = 1\}$. The one-stage algorithm that tests the sets $T_1, \ldots, T_N$ in parallel is the one-stage algorithm defined by M, or equivalently by C. For the sake of convenience, we represent the responses to the N tests by an N-entry binary vector whose i-th entry is equal to 1 if the answer to the i-th test is YES, and is 0 otherwise. We will refer to the vector z as the *response vector*.

In the classical model of group testing the response vector is the bitwise OR of the columns associated with the positive items. Hence, in that model a matrix M defines a one-stage algorithm that determines up to p positive items if and only if the OR's of up to p columns are all distinct.

In the GTI model, things become more complicated and a matrix that defines a one-stage group testing algorithm under this model should satisfy a condition stronger than the one working for classical group testing. In the following section

we will establish the combinatorial property that an n-column binary matrix should to satisfy in order to define a one-stage algorithm that successfully determines up to p positives in the presence of up to r inhibitors.

Let C be a code of length N. Given $q > 1$ codewords (binary columns) $c_{\ell_1}, \ldots, c_{\ell_q}$, we denote by $(c_{\ell_1} \vee \cdots \vee c_{\ell_q})$ the boolean sum (OR) of $c_{\ell_1}, \ldots, c_{\ell_q}$. Given two binary columns c_h and c_j, we say that c_h is *covered* by c_j if $c_h(i) = 1$ implies $c_j(i) = 1$, for all i.

For any two sets $Q \subset C$ and $J \subset C$ such that $Q \cap J = \emptyset$, we denote by $Z(Q, J) = (Z(1), \ldots, Z(N))$ the column vector defined as follows:

$$Z(i) = \begin{cases} 1 & \text{if } \bigvee_{x \in Q} x(i) = 1 \text{ and } \bigvee_{x \in J} x(i) = 0, \\ 0 & \text{otherwise,} \end{cases}$$

for $i = 1, \ldots, N$. For $J = \emptyset$ we define $Z(Q, J) = \bigvee_{x \in Q} x(i)$.

If $Q = \{c\}$ for some $c \in C$, then we will denote the vector $Z(Q, J)$ simply by $Z(c, J)$ for any set of columns $J \subset C$ such that $c \notin J$.

Definition 1 (Dyachkov et al. 2001) Given integers p, r and n, with $p + r \leq n$, we say that a boolean matrix M with n columns and N rows is an *inhibitory* (p, r)-*design* if for any choice of four subsets P_1, P_2, I_1, and I_2 of columns of M such that $P_1 \neq P_2$, $P_1 \cap I_1 = \emptyset$, $P_2 \cap I_2 = \emptyset$, $|P_1|, |P_2| \leq p$, and $|I_1|, |I_2| \leq r$, one has $Z(P_1, I_1) \neq Z(P_2, I_2)$. The minimum length of an inhibitory (p, r)-design of size n will be denoted by $N_I(p, r, n)$.

Observe that a matrix M defines a one-stage group testing algorithm that successfully determines up to p positive items in the presence of up to r inhibitors if and only if M is an inhibitory (p, r)-design. Indeed, if P is the set of columns associated with the set of the positive items $\mathcal{P}$ and I is the set of columns associated with the set of inhibitory items $\mathcal{I}$, then the response vector z is equal to $Z(P, I)$. An inhibitory (p, r)-design guarantees that for every set P' of up to p columns of M and for every set I' of up to r columns of M with $P' \cap I' = \emptyset$, we have $Z(P', I') = Z(P, I)$ if and only if $P' = P$.

In the following we establish a lower bound on the minimum length $N_I(p, r, n)$ of an inhibitory (p, r)-design of size n so as to limit from below the minimum number of tests needed to solve the GTI problem by a nonadaptive strategy.

To present our bound we need to recall the definition of (d, q)-superimposed codes (Dyachkov and Rykov 1983). These codes have the property that the boolean sum of any d columns is not covered by the boolean sum of any other q columns. When $d = 1$, these codes correspond to classical superimposed codes (Kautz and Singleton 1964), or equivalently cover free families (Erdös et al. 1985). Here we will denote by $N(d, q, n)$ the minimum length of a (d, q)-superimposed code of size n. The following asymptotic lower and upper bounds (De Bonis et al. 2005; Dyachkov and Rykov 1983) on $N(d, q, n)$ hold:

$$N(d, q, n) = \begin{cases} \Omega\left(\frac{q^2}{d \log q} \log \frac{n}{q}\right) & \text{if } q \geq 2d, \\ \Omega\left(q \log \frac{n}{d}\right) & \text{if } q < 2d, \end{cases} \tag{1}$$

$$N(d,q,n) = O\left(\frac{q^2}{d}\log\frac{n}{q}\right). \tag{2}$$

The following results uncover a relation between one-stage procedures for group testing with inhibitors and superimposed codes.

Theorem 1 *For any positive integers p, r and n, with $n \geq p+r$, the minimum length of an inhibitory (p,r)-design of size n is $N_I(p,r,n) \geq N(1,p+r-1,n) = \Omega(\frac{(p+r)^2}{\log(p+r)}\log(\frac{n}{p+r}))$.*

Proof We will show that any inhibitory (p,r)-design M is a $(1,p+r-1)$-superimposed code. Let M be an inhibitory (p,r)-design and let us suppose by contradiction that M is not a $(1,p+r-1)$-superimposed code. Then, in M there exist $p+r$ distinct columns $c, c_1, \ldots, c_{p+r-1}$ such that c is covered by $c_1 \vee \cdots \vee c_{p+r-1}$. Let $P = \{c_1, \ldots, c_{p-1}\}$ and $I = \{c_p, \ldots, c_{p+r-1}\}$. Notice that if $c(i) = 0$ then it is $Z(\{c\} \cup P, I)(i) = Z(P,I)(i)$. Moreover, since c is covered by the columns in $P \cup I$, one has that if $c(i) = 1$ then it must be $\bigvee_{c\in P} c(i) = 1$ or $\bigvee_{c\in I} c(i) = 1$ (or both). As a consequence, it is $Z(\{c\} \cup P, I)(i) = Z(P,I)(i)$. Hence, it follows that $Z(\{c\} \cup P, I) = Z(P,I)$ thus contradicting the hypothesis that M is an inhibitory (p,r)-design. By using lower bound (1), we obtain the asymptotic lower bound in the statement of the theorem. □

The following result is an immediate consequence of Theorem 1.

Theorem 2 *Let $\mathcal{O}$ be a set of n items which is known to contain at most $p \geq 1$ positive items and at most $r \geq 1$ inhibitors. Any one-stage algorithm that successfully identifies all positive items in $\mathcal{O}$ uses at least $N(1,p+r-1,n) = \Omega(\frac{(p+r)^2}{\log(p+r)}\log(\frac{n}{p+r}))$ tests.*

We remark that the above lower bound holds also if r is known to be the exact number of inhibitors.

Superimposed codes can be exploited not only to bound from below the minimum length of inhibitory (p,r)-designs, but also to provide existential results. Indeed the authors of (Dyachkov et al. 2001) proved the following result.

Theorem 3 (Dyachkov et al. 2001) *Let p, r and n be integers with $p+r \leq n$. There exists an inhibitory (p,r)-design of size n and length $N = N(1,p+r,n) = O((p+r)^2\log(\frac{n}{p+r}))$.*

The following theorem is an immediate consequence of Theorem 3.

Theorem 4 *Let $\mathcal{O}$ be a set of n items which is known to contain at most $p \geq 1$ positive items and at most $r \geq 1$ inhibitors. There exists a one-stage algorithm that successfully identifies all positive items in $\mathcal{O}$ by at most $N(1,p+r,n) = O((p+r)^2\log(\frac{n}{p+r}))$ tests.*

Notice that the one-stage algorithm of Theorem 4 uses an asymptotic number of tests that exceeds by a factor of $\log(r+p)$ the asymptotic lower bound of Theorem 2. In order to get rid of this difference between these upper and lower bounds, one should be able to give a tight asymptotic estimate of the minimum length of $(1,q)$-superimposed codes. This is a major open problem in extremal combinatorics.

In the following we derive a lower bound on the number of tests used by *any* algorithm that solves the GTI problem under the above hypothesis. This result holds independently of the number of stages used by the algorithm.

2.2 A lower bound on the number of tests required to solve the GTI problem

In De Bonis and Vaccaro (1998) it has been proved that *any algorithm* that finds p positives in the presence of r inhibitors requires

$$\Omega\left(N(p,r,n-p-1)+\ln\binom{n}{p}\right) \tag{3}$$

tests, no matter the number of stages it uses. We remark that in the computation of the above lower bound, it has been assumed that the exact number of positives is known. In De Bonis et al. (2005) this lower bound has been shown to be $\Omega(N(p,r,n)+r\log\frac{n}{r}+p\log\frac{n}{p})$. By using lower bound (1), we have that lower bound (3) is

$$\Omega\left(\frac{r^2}{p\log r}\log\frac{n}{r}+r\log\frac{n}{r}+p\log\frac{n}{p}\right) \quad \text{if } r\geq 2p, \tag{4}$$

and

$$\Omega\left(r\log\frac{n}{p}+p\log\frac{n}{p}\right), \quad \text{if } r<2p. \tag{5}$$

In De Bonis et al. (2005) it has been provided an algorithm for the case when the exact number of positives is known, that asymptotically attains lower bound (3). Therefore, to improve on this lower bound one has to exploit the hypothesis that the exact number of positives is not known.

Theorem 5 *Let $\mathcal{O}$ be a set of n items which is known to contain at most $p\geq 1$ positive items and at most $r\geq 1$ inhibitors. Any algorithm that successfully identifies all positive items in $\mathcal{O}$ uses at least $N(2,r,n)$ tests. If it is not known whether $\mathcal{O}$ contains at least one positive item then the algorithm uses $N(1,r,n)$ tests.*

Proof In this proof we will use an argument similar to the one exploited in Theorem 5 of De Bonis and Vaccaro (1998). Let us fix any group testing algorithm that finds up to p positive items in the presence of r inhibitors. Let $T_1,\ldots,T_m$ denote the sets tested by the first m tests of the algorithm. For each item $j\in[n]$ we define the m-th indicator set R_j^m of j as $R_j^m=\{i\in[m]: j\in T_i\}$. Let $M=\|c_j(i)\|$ be the $m\times n$ matrix whose columns are the characteristic vectors of sets $R_1^m,\ldots,R_n^m$, that is $c_j(i)=1$ if and only if $j\in T_i$. Notice that the rows of M are the characteristic vectors of $T_1,\ldots,T_m$.

First we deal with the case when it is not known whether the set P of the positive items is empty. We will show that in order for the algorithm to identify the set of positive items by m tests, M must be a $(1, r)$-superimposed code. Suppose by contradiction that there exist $r + 1$ columns $c_{j_1}, \ldots, c_{j_r}, c_{j_{r+1}}$ such that c_{j_1} is covered by the boolean sum $c_{j_2} \vee \cdots \vee c_{j_r}$. It follows that $R^m_{j_1}$ is contained in the union $R^m_{j_2} \cup \cdots \cup R^m_{j_{r+1}}$ thus implying that j_1 occurs in some test set T only if at least one of $j_2, \ldots, j_{r+1}$ belongs to T. Then a malicious adversary could make j_1 the unique positive item and $j_2, \ldots, j_{r+1}$ the inhibitory items so that the responses to the first m tests are NO. Unfortunately, after performing these m tests, the algorithm cannot distinguish between the case when $\mathcal{O}$ contains a positive item, namely item j_1, and the case when $\mathcal{O}$ does not contain any positive item. It follows that the algorithm is able to identify the positive items after m tests only if the matrix M is a $(1, r)$-superimposed code and consequently $m \geq N(1, r, n)$.

If $\mathcal{O}$ is known to contain at least one defective item then one can use a similar argument to prove that the matrix M associated with the first m tests must be a $(2, r)$-superimposed codes. Suppose by contradiction that M is not a $(2, r)$-superimposed code. Then there exist $r + 2$ items $j_1, \ldots, j_{r+2}$ such that $R^m_{j_1} \cup R^m_{j_2}$ is contained in the union $R^m_{j_3} \cup \cdots \cup R^m_{j_{r+2}}$. A malicious adversary could make $\{j_1, j_2\}$ the set of the positive items and $\{j_3, \ldots, j_{r+2}\}$ the set of inhibitors so that the responses to the first m test are all NO. In this case the algorithm cannot decide whether j_1 and j_2 are both positive or whether only one of the two items is positive. Therefore, the matrix M must be a $(2, r)$-superimposed code and as a consequence contains at least $N(2, r, n)$ rows. □

By Theorem 5 and by taking into account the information theoretic lower bound we get the following lower bound.

Corollary 1 *Let $\mathcal{O}$ be a set of n items which is known to contain at most $p \geq 1$ positive items and at most $r \geq 1$ inhibitors. Any algorithm that successfully identifies all positive items in $\mathcal{O}$ uses at least $N(2, r, n) + \Omega(p \log \frac{n}{p})$ tests. If it is not known whether $\mathcal{O}$ contains at least one positive item then the algorithm uses $N(1, r, n) + \Omega(p \log \frac{n}{p})$ tests.*

Corollary 1 along with lower bound (1) implies that any algorithm that successfully identifies up to $p \geq 1$ positive items in the presence of up to $r \geq 1$ inhibitors, uses

$$\Omega\left(\frac{r^2}{\log r} \log n + p \log \frac{n}{p}\right) \tag{6}$$

tests.

Notice that the above bounds hold even if it is known that $\mathcal{O}$ contains exactly r inhibitors.

The authors of (De Bonis and Vaccaro 1998) described an $N(1, r, n) + O(r \log(n/r) + p \log(n/p))$ worst case algorithm for the GTI problem. It is possible to show that their algorithm works also under the hypothesis considered in this section that the exact number of positives is not given. In view of upper bound (2)

on $N(1,r,n)$, the above upper bound is $O(r^2\log(n/r)+p\log(n/p))$ and consequently differs by a $\log r$ factor from lower bound (6). We remark that the algorithm of De Bonis and Vaccaro (1998) can be proved to work even in the case when it is not known whether the set $\mathcal{O}$ contains at least one positive item. Corollary 1 implies that in such case the algorithm is asymptotically optimal. Indeed, lower bound (1) implies $N(1,r,n)=\Omega(r\log(n/r))$, and consequently the second lower bound in Corollary 1 is $\Omega(N(1,r,n)+r\log(n/r)+p\log(n/p))$, which matches the upper bound on the number of tests used by the algorithm.

If $p=O(r)$ then lower bound (6) implies that the asymptotic number of tests used by any algorithm that works under the hypothesis considered in this section, differs by a factor not smaller than $1/\log r$ from the number of tests used by the one-stage algorithm of Theorem 4. Again this gap is a consequence of the $\log r$ gap between the known upper and lower bounds on the length of superimposed codes.

We mention that the authors of (Hwang and Liu 2003) presented an error-tolerant two-stage algorithm that works under the hypothesis considered in this section. We remark that the algorithm in Hwang and Liu (2003) is not a trivial two-stage algorithm in view of the fact that its second stage tests non-singleton subsets of items and consequently is not a trivial confirmatory stage. In the case when no erroneous response is allowed, this algorithm uses $N(1,p,n)+N(1,r,n)$ tests. For $p=O(r)$ this upper bound is $O(N(1,r,n))$ and consequently, by Theorem 5, is asymptotical optimal.

3 Generalized superimposed codes and selectors

In this section we introduce two new combinatorial structures that generalize very well known combinatorial objects. Though these combinatorial structures are instrumental to the results of Sect. 4, we chose to present them in a separate section since we believe them to be of independent interest and to have other important applications in other contexts.

Definition 2 Given two positive integers n and q, and a multiset of positive integers $\mathcal{P}=\{p_1,p_2,\ldots,p_s\}$, with $s=|\mathcal{P}|\geq 1$ and $q+\sum_{i=1}^{s}p_i\leq n$, we say that a boolean matrix M with n columns and N rows is a $(\mathcal{P},q)$-superimposed code if for any choice of $s+1$ pairwise disjoint subsets $A_1,\ldots,A_s,B$ of columns of M, with $|A_i|=p_i$ for $i=1,\ldots,s$, and $|B|=q$, there exists a row of M in which all entries corresponding to the columns in B are equal to zero, and for each $i\in\{1,\ldots,s\}$, the entries corresponding to columns in A_i contain at least one nonzero value. The integers n and N are the size and the length of the $(\mathcal{P},q)$-superimposed code, respectively. The minimum length of a $(\mathcal{P},q)$-superimposed code of size n will be denoted by $N(\mathcal{P},q,n)$.

The above definition generalizes the already recalled notion of (p,q)-superimposed code of Dyachkov and Rykov (1983). One can easily verify that for $\mathcal{P}=\{p\}$ our definition of $(\mathcal{P},q)$-superimposed codes corresponds to that of (p,q)-superimposed codes. Further, r-cover free families (Erdös et al. 1985) (equivalently, superimposed codes (Kautz and Singleton 1964), disjunct codes (Du and Hwang 2000, 2006), and

strongly selective families (Clementi et al. 2001; Chrobak et al. 2000)) correspond to our $(\{1\}, r)$-superimposed codes. Finally, families of sets such that the intersection of any s members of the family is not contained in the union of any other q members (Dyachkov et al. 2002; Stinson et al. 2000a), are equivalent to $(\mathcal{P}, q)$-superimposed codes in which $\mathcal{P}$ consists of s integers all being equal to 1.

In the following we provide an upper bound on the minimum length $N(\mathcal{P}, q, n)$ of a $(\mathcal{P}, q)$-superimposed code of size n. In order to derive our upper bound, we find convenient to introduce a new and generalized version of (k, m, n)-selectors (De Bonis et al. 2005).

Definition 3 Given integers k, m, w and n, with $1 \leq w < k \leq n$ and $1 \leq m \leq \binom{k}{w}$, we say that a boolean matrix M with t rows and n columns is a (k, m, w, n)-*selector* if *any* submatrix of M obtained by choosing k out of n arbitrary columns of M contains at least m distinct rows with exactly w entries equal to 1. The integer t is the *size* of the (k, m, w, n)-selector. The minimum size of a (k, m, w, n)-selector will be denoted by $t(k, m, w, n)$.

The (k, m, n)-selectors introduced in De Bonis et al. (2005) are equivalent to our $(k, m, 1, n)$-selectors. Our definition includes as a particular case the strongly selective families of Chrobak et al. (2000); Clementi et al. (2001) that coincide with our definition of $(k+1, k+1, 1, n)$-selectors, and the k-selectors of Chrobak et al. (2000) that correspond to our $(2k, 3k/2+1, 1, n)$-selectors.

It is possible to see that a (k, m, q, n)-selector with parameters $k = q + \sum_{i=1}^{s} p_i$, $m = \binom{k}{w} - \prod_{i=1}^{s} p_i + 1$, and $w = s$ is a $(\{p_1, \ldots, p_s\}, q)$-superimposed code. An upper bound on the minimum length of $(\{p_1, \ldots, p_s\}, q)$-superimposed codes of size n will be obtained by deriving an upper bound on the size of our generalized selectors of length n. The upper bound will be proved by showing that a (k, m, w, n)-selector is indeed the cover of a properly defined hypergraph.

Theorem 6 *For any integers k, m, w and n, with $1 \leq w < k$, $1 \leq m \leq \binom{k}{w}$ and $n \geq k^2 w$, there exists a (k, m, w, n)-selector of size t, with*

$$t < \frac{ek^{w+1}}{z} \ln\lceil n/k \rceil - \frac{ewk^w}{z} \ln w + \frac{ek^w}{z}(w + m + k + z - 1),$$

where $z = \binom{k}{w} - m + 1$ and $e = 2.7182\ldots$ is the base of the natural logarithm.

Proof Let X be a finite set and $\mathcal{F}$ be a family of subsets of X. We denote by $\mathcal{H} = (X, \mathcal{F})$ the hypergraph having X as the set of vertices and $\mathcal{F}$ as the set of hyperedges. A subset $T \subseteq X$ such that $T \cap E \neq \emptyset$, for any hyperedge $E \in \mathcal{F}$, is called a *cover* of $\mathcal{H}$. In the following we will think of a cover T as a matrix with $|T|$ rows. The minimum size of a cover of an hypergraph $\mathcal{H}$ is denoted by $\tau(\mathcal{H})$. Lovász proved the following upper bound (Lovàsz 1975) on the minimum size $\tau(\mathcal{H})$ of a cover of $\mathcal{H}$.

$$\tau(\mathcal{H}) < \frac{|X|}{\min_{E \in \mathcal{F}} |E|}(1 + \ln \Delta), \tag{7}$$

where $\Delta = \max_{x \in X} |\{E\colon E \in \mathcal{F} \text{ and } x \in E\}|$.

We will prove that a (k, m, w, n)-selector is a cover of a properly defined hypergraph so as to exploit upper bound (7) to limit from above the minimum size $t(k, m, w, n)$.

Let X be the set of all binary vectors $x = (x(1), \ldots, x(n))$ of length n having $b \geq w$ entries equal to 1, and let U be the set of all binary vectors of length k having exactly w entries equal to 1. For any subset of indices $S = \{i_1, \ldots, i_k\}$, with $1 \leq i_1 \leq i_2 < \cdots < i_k \leq n$, and for any binary vector $u = (u(1), \ldots, u(k)) \in U$, we define the set of binary vectors $E_{u,S} = \{x = (x(1), \ldots, x(n)) \in X : x(i_1) = u(1), \ldots, x(i_k) = u(k)\}$. For any set $A \subseteq U$ of size r, $r = 1, \ldots, \binom{k}{w}$, and any set $S \subseteq \{1, \ldots, n\}$, with $|S| = k$, we define $E_{A,S} = \bigcup_{u \in A} E_{u,S}$ and $\mathcal{F}_r = \{E_{A,S} : A \subset U, |A| = r, \text{ and } S \subseteq \{1, \ldots, n\}, |S| = k\}$. For any $r = 1, \ldots, \binom{k}{w}$, let $\mathcal{H}_r$ denote the hypergraph $\mathcal{H}_r = (X, \mathcal{F}_r)$. Let $z = \binom{k}{w} - m + 1$. We will show that *any* cover T of $\mathcal{H}_z$ is a (k, m, w, n) selector. Indeed, let T be a cover of $\mathcal{H}_z$ and let us assume by contradiction that there exists a set of indices $S = \{i_1, \ldots, i_k\}$ such that the submatrix of T formed by the columns of T with indices $i_1, \ldots, i_k$ contains at most $m - 1$ distinct rows with w entries equal to 1. Let $u_{j_1}, \ldots, u_{j_q}$, $q \leq m - 1$, denote these rows. Hence, there exists a subset $A \subseteq U$ of cardinality $|A| = z = \binom{k}{w} - m + 1$ that contains none of the vectors $u_{j_1}, \ldots, u_{j_q}$. It follows that T does not contain any vertex of the corresponding edge $E_{A,S}$ of $\mathcal{H}_z$, thus contradicting the hypothesis that T is a cover for $\mathcal{H}_z$.

Now we use inequality (7) to determine an upper bound on t. To this aim we need to estimate the quantities

$$|X|, \quad \min\{|E| : E \in \mathcal{F}_z\} \quad \text{and} \quad \Delta,$$

for the hypergraph $\mathcal{H}_z$.

The set X has size $|X| = \binom{n}{b}$ since it contains all binary vectors of length n with b entries equal to 1.

In order to compute $\min\{|E| : E \in \mathcal{F}_z\}$, we observe that each hyperedge $E_{A,S} \in \mathcal{F}_z$ is the union of z disjoint sets $E_{u,S}$ and each set $E_{u,S}$ contains $\binom{n-k}{b-w}$ vectors. Therefore, each hyperedge $E_{A,S}$ has cardinality

$$|E_{A,S}| = z\binom{n-k}{b-w}.$$

Now we have to compute Δ. To this aim, observe that for each $\mathbf{x} \in X$ there are $\binom{b}{w}\binom{n-b}{k-w}$ distinct sets $E_{u,S}$ that contains x, and for each $u \in U$ there are $\binom{z+m-2}{z-1}$ distinct subsets $A \subseteq U$ of size z that contain u. Consequently, each element $x \in X$ belongs to

$$\Delta = \binom{b}{w}\binom{n-b}{k-w}\binom{z+m-2}{z-1}$$

hyperedges.

Therefore, we have

$$t = \tau(\mathcal{H}_z) < \frac{\binom{n}{b}}{z\binom{n-k}{b-w}}\left[1 + \ln\left(\binom{b}{w}\binom{n-b}{k-w}\binom{z+m-2}{z-1}\right)\right]. \tag{8}$$

For the sake of the simplicity, let us assume n to be a multiple of k and let us set $b = n/k$ in inequality (8).

First we compute an upper bound on

$$\frac{\binom{n}{n/k}}{\binom{n-k}{n/k-w}}.$$

For $k = 2$ it is

$$\frac{\binom{n}{n/k}}{\binom{n-k}{n/k-w}} = \frac{\binom{n}{n/2}}{\binom{n-2}{n/2-1}} < 2e,$$

whereas for $k \geq 3$ it is

$$\frac{\binom{n}{n/k}}{\binom{n-k}{n/k-w}} = k\left(\prod_{i=1}^{k-w}\frac{n-i}{n-n/k-i+1}\right)\left(\prod_{i=1}^{w-1}\frac{n-k+w-i}{n/k-i}\right).$$

In both products, factors can be limited from above by the factor with the largest index. Therefore, we have

$$\begin{aligned}
\frac{\binom{n}{n/k}}{\binom{n-k}{n/k-w}} &\leq k\left(\frac{n-k+w}{n-n/k-k+w+1}\right)^{k-w}\left(\frac{n-k+1}{n/k-w+1}\right)^{w-1} \\
&= k\left(\frac{n-k+w}{n-n/k-k+w+1}\right)^{k-1}\left(\frac{n-k+w}{n-n/k-k+w+1}\right)^{-(w-1)} \\
&\quad \times \left(\frac{n-k+1}{n/k-w+1}\right)^{w-1} \\
&= k\left(\frac{k(n-k+w)}{k(n-k+w)-(n-k)}\right)^{k-1} \\
&\quad \times \left(\frac{(n-n/k-k+w+1)(n-k+1)}{(n-k+w)(n/k-w+1)}\right)^{w-1}.
\end{aligned} \tag{9}$$

Let us upper bound expression (9).

We have

$$\begin{aligned}
\left(\frac{k(n-k+w)}{k(n-k+w)-(n-k)}\right)^{k-1} &= \left(1+\frac{n-k}{k(n-k+w)-(n-k)}\right)^{k-1} \\
&< \left(1+\frac{1}{k-1}\right)^{k-1} < e.
\end{aligned}$$

Moreover, it results

$$\begin{aligned}
&\left(\frac{(n-n/k-k+w+1)(n-k+1)}{(n-k+w)(n/k-w+1)}\right)^{w-1}\\
&\quad= k^{w-1}\left(\frac{(n-n/k-k+w+1)(n-k+1)}{(n-k+w)(n-kw+k)}\right)^{w-1}\\
&\quad\le k^{w-1}\left(\frac{n-n/k-k+w+1}{n-kw+k}\right)^{w-1}\\
&\quad\le k^{w-1}\left(\frac{n-kw-k+w+1}{n-kw+k}\right)^{w-1} \quad \text{since } n\ge k^2w\\
&\quad< k^{w-1}.
\end{aligned}$$

Hence, we have

$$\frac{\binom{n}{n/k}}{\binom{n-k}{n/k-w}} < ek^w, \tag{10}$$

for all $k \ge 2$ satisfying the hypotheses of the theorem.

The well known inequality $\binom{a}{b} \le (ea/b)^b$ implies

$$\begin{aligned}
\Delta &= \binom{n/k}{w}\binom{n-n/k}{k-w}\binom{z+m-2}{z-1}\\
&\le \left(\frac{n}{kw}\right)^w\left(\frac{n-n/k}{k-w}\right)^{k-w}\left(\frac{z+m-2}{z-1}\right)^{z-1}e^{k+z-1}\\
&= \left(\frac{n}{k}\right)^k\left(\frac{1}{w}\right)^w\left(\frac{k-1}{k-w}\right)^{k-w}\left(\frac{z+m-2}{z-1}\right)^{z-1}e^{k+z-1}\\
&< \left(\frac{n}{k}\right)^k\left(\frac{1}{w}\right)^w\left(\frac{k}{k-w}\right)^{k-w}\left(\frac{z+m-2}{z-1}\right)^{z-1}e^{k+z-1}\\
&= \left(\frac{n}{k}\right)^k\left(\frac{1}{w}\right)^w\left(1+\frac{w}{k-w}\right)^{k-w}\left(1+\frac{m-1}{z-1}\right)^{z-1}e^{k+z-1}.
\end{aligned}$$

Since it is $(1+\frac{w}{k-w})^{k-w} \le e^w$ and $(1+\frac{m-1}{z-1})^{z-1} \le e^{m-1}$, we have

$$\Delta \le \left(\frac{n}{k}\right)^k\left(\frac{1}{w}\right)^w e^w e^{m-1}e^{k+z-1}. \tag{11}$$

Inequality (8) along with inequalities (10) and (11) imply

$$t = \tau(\mathcal{H}_z) < \frac{ek^w}{z}\left(1+\ln\left(\left(\frac{n}{k}\right)^k\left(\frac{1}{w}\right)^w e^{w+m+k+z-2}\right)\right).$$

If n is not a multiple of k then we set $n' = k\lceil n/k \rceil$ and by the above argument we have that there exists a (k, m, w, n')-selector of size

$$t < \frac{ek^w}{z}\left(1 + \ln\left(\left\lceil \frac{n}{k} \right\rceil^k \left(\frac{1}{w}\right)^w e^{w+m+k+z-2}\right)\right),$$

from which the upper bound in the statement of the theorem follows. □

Theorem 6 imply the following upper bound on the length of $(\mathcal{P}, q)$-superimposed codes.

Theorem 7 *Given two positive integers n and q, and a multiset of integers $\mathcal{P} = \{p_1, p_2, \ldots, p_s\}$, with $s = |\mathcal{P}| \geq 1$ and $q + \sum_{i=1}^{s} p_i \leq n$, there exists a $(\mathcal{P}, q)$-superimposed code of size n and length*

$$N < \frac{e(q + \sum_{i=1}^{s} p_i)^{s+1}}{\prod_{i=1}^{s} p_i} \ln\left\lceil \frac{n}{(q + \sum_{i=1}^{s} p_i)} \right\rceil + \frac{e(q + \sum_{i=1}^{s} p_i)^{s}}{\prod_{i=1}^{s} p_i}\left(s(1 - \ln s) + q + \sum_{i=1}^{s} p_i + \binom{q + \sum_{i=1}^{s} p_i}{s}\right).$$

Proof Observe that a (k, m, w, m)-selector with parameters $k = q + \sum_{i=1}^{s} p_i$, $m = \binom{k}{w} - \prod_{i=1}^{s} p_i + 1$ and $w = s$ is a $(\mathcal{P}, q)$-superimposed code. Then the theorem follows from Theorem 6. □

If we set $s = 1$ and $p_1 = p$ in the upper bound of Theorem 7, we obtain that the minimum length $N(p, q, n)$ of a (p, q)-superimposed code of size n is less than $e(q + p)^2(\log\lceil n/(q + p)\rceil)/p + e(q + p)(1 + 2q + 2p)/p$. Moreover, for $p > q$, any (q, q)-superimposed code is also a (p, q)-superimposed code, and consequently $N(p, q, n) \leq N(q, q, n) < 4eq \log\lceil n/(2q)\rceil + 2e(1 + 4q)$. These upper bounds are asymptotically the same as upper bound (2).

4 An efficient trivial two-stage algorithm for the GTI problem

In this section we present a trivial two-stage group testing algorithm for the GTI problem that works under the hypothesis that the exact number of positive items is given. Our two-stage algorithm works as follows.

Stage 1 This stage determines a set of size at most $2r + 2p - 2$ that contains all inhibitors and all positive items. Let M_1 be a $(\{p, r\}, r)$-superimposed code and let M_2 be a $(p, 2r + p - 1)$-superimposed code. We denote by P the set of the columns associated with the set of the positive items $\mathcal{P}$, and by I the set of the columns associated with the set of the inhibitory items $\mathcal{I}$.

This stage performs in parallel all tests associated with the rows of M_1 and M_2. In other words this stage executes the one-stage algorithm represented by the matrix obtained by concatenating the rows of M_1 with those of M_2.

Now we will show that the responses to the tests associated with the rows of M_1 allow to determine a set containing all inhibitory items. Let w be the bitwise complement of the vector of responses to the tests associated with the rows of M_1, that is w is the binary vector whose i-th entry is equal to 1 if the response to the test associated to the i-th row of M_1 is NO, and is 0 otherwise. We denote by X the set of columns of M_1 which are covered by w. One can easily see that for each column x of M_1 associated with an item in $\mathcal{I}$, one has $x \in X$. Indeed, let x be a column of M_1 associated with an item in $\mathcal{I}$. If the i-th entry of x is 1 then this means that the pool tested by the i-th test contains at least one inhibitor and consequently the response to the i-th test is NO thus implying that $w(i) = 1$. Observe that there might be positive items that are associated to columns in X. Now we will show that the columns of X not associated with positive items are at most $2r - 1$, that is $|X \setminus P| \leq 2r - 1$. Suppose by contradiction that $|X \setminus P| \geq 2r$. Since $|\mathcal{I}| \leq r$, there exists a subset $A \subset X \setminus P$ of size r that does not contain columns associated with items of $\mathcal{I}$. Since M_1 is a $(\{p, r\}, r)$-superimposed code, there exists a row index i such that all columns in I have the i-th entry set to 0, whereas at least one column of P and one column of A have the i-th entry equal to 1. This implies that the test subset associated with the i-th row contains at least one positive item and no inhibitory item so that the response to the i-th test is YES. Consequently the i-th entry of vector w is equal to 0. Since at least one column of A has the i-th entry equal to 1, we get a contradiction to the hypothesis that there are at least $2r$ columns not contained in P that are covered by w.

We need to determine a set of size smaller than $p + |P \setminus X|$ that contains all positive items that are not associated with columns in X. Let z be the vector of responses to the tests associated with the rows of M_2. Let Y be the set of columns $Y = \{c : c \text{ is a column of } M_2 \text{ such that } c \notin X \text{ and } Z(c, X) \text{ is covered by } z\}$. Notice that for any column $c \notin X$ associated with a positive item one has that $Z(c, I)$ is covered by z. Moreover, since we have just proved that $I \subseteq X$, we have that $Z(c, X)$ is covered by $Z(c, I)$ for any column $c \notin X$. Consequently, if $c \notin X$ is a column associated with a positive item then the vector $Z(c, X)$ is covered by z. This implies that $P \setminus X \subseteq Y$. Now we need to show that $|Y| < p + |P \setminus X|$. Suppose by contradiction that $|Y| \geq p + |P \setminus X|$ and let us consider a subset $L \subseteq Y \setminus P$ of size p. Such a set L exists since $|Y \cap P| \leq |P \setminus X|$. Since M_2 is a $(p, p + 2r - 1)$-superimposed code and $|X \cup P| \leq p + 2r - 1$, there exists a row index i such that at least one column of L has the i-th entry equal to 1, whereas all columns of $X \cup P$ have the i-th entry set to 0. It follows that the test subset associated with the i-th row contains no positive item so that $z(i) = Z(P, I)(i) = 0$. On the other hand, it follows also that at least one column $c' \in L \subset Y$ has the i-th entry set to 1 whereas all columns in X have the i-th entry set to 0 so that $Z(c', X)(i) = 1$. This is obviously a contradiction since, by definition of Y, one has that for each $y \in Y$, $Z(y, X)$ is covered by z.

We remark that the tests associated to rows of M_2 can be performed in parallel with those associated to rows of M_1 since we do not need to know the answers to these tests in order to construct M_2. On the other hand, in order to decode the responses to the tests defined by M_2, we need first to decode the responses to all tests associated with the rows of M_1.

Stage 2 This stage individually probes the items associated with the columns of $X \cup Y$ and returns those which test positive.

The above algorithm provides the following result.

Theorem 8 *Let $\mathcal{O}$ be a set of n items which is known to contain exactly $p \geq 1$ positive items and at most $r \geq 1$ inhibitors. There exists a trivial two-stage algorithm that successfully identifies all positive items in $\mathcal{O}$ by at most $N(\{p, r\}, r, n) + N(p, p + 2r - 1, n) + 2p + 2r - 2$ tests.*

We can estimate the number of tests performed by the two-stage algorithm by resorting to Theorem 7.

Corollary 2 *Let $\mathcal{O}$ be a set of n items which is known to contain exactly $p \geq 1$ positive items and at most $r \geq 1$ inhibitors. There exists a trivial two-stage algorithm that successfully identifies all positive items in $\mathcal{O}$, using a number of tests smaller than*

$$\frac{e(h+2r)^3}{hr} \ln\left\lceil \frac{n}{h+2r} \right\rceil + \frac{e(2p+2r-1)^2}{p} \ln\left\lceil \frac{n}{2p+2r-1} \right\rceil + O\left(\frac{(h+r)^4}{hr}\right) + O((p+r)^2/p),$$

where $h = \min\{p, r\}$.

Proof The stated upper bound is obtained by observing that $N(\{p, r\}, r, n) \leq N(\{h, r\}, r, n)$ and by applying the upper bound of Theorem 7 to $N(\{h, r\}, r, n)$ and $N(p, p + 2r - 1, n)$ in the upper bound of Theorem 8. □

Notice that if $p \leq r/2$ then the upper bound of Corollary 2 is $O(\frac{r^2}{p} \log(n/r))$, whereas if $p > r/2$ it is $O((r + p) \log(n/p))$. It follows that for $p > r/2$ our two-stage algorithm attains lower bound (5) and as a consequence is asymptotically optimal. For $p \leq r/2$ the two-stage algorithm uses a number of tests that exceeds lower bound (4) by a $\log r$ factor. Notice that to prove the optimality of our two-stage algorithm, one should be able to show that $N(p, r, n) = \Theta(N(\{p, r\}, r, n) + N(p, p + 2r - 1, n))$.

References

Barillot E, Lacroix B, Cohen D (1991) Theoretical analysis of library screening using an n-dimensional pooling strategy. In: Nucleic acids research, pp 6241–6247

Balding DJ, Bruno WJ, Knill E, Torney DC (1996) A comparative survey of non-adaptive pooling design. In: Speed TP, Waterman MS (eds) Genetic mapping and DNA sequencing. IMA volumes in mathematics and its applications. Springer, Berlin, pp 133–154

Berger T, Mehravari N, Towsley D, Wolf J (1984) Random multiple-access communication and group testing. IEEE Trans Commun 32(7):769–779

Bruno WJ, Balding DJ, Knill E, Bruce D, Whittaker C, Dogget N, Stalling R, Torney DC (1995) Design of efficient pooling experiments. Genomics 26:21–30

Chaudhuri S, Radhakrishnan J (1996) Deterministic restrictions in circuit complexity. In: Proceedings of the twenty-eighth annual ACM symposium on the theory of computing (STOC 96), pp 30–36

Chrobak M, Gasieniec L, Rytter W (2000) Fast broadcasting and gossiping in radio networks. In: Proceedings of 42nd IEEE annual symposium on foundation of computer science (FOCS 2000), pp 575–581

Clementi AEF, Monti A, Silvestri R (2001) Selective families, superimposed codes, and broadcasting on unknown radio networks. In: Proceedings of symposium on discrete algorithms (SODA'01), pp 709–718

De Bonis A, Vaccaro U (1998) Improved algorithms for group testing with inhibitors. Inf Process Lett 66:57–64

De Bonis A, Gasieniec L, Vaccaro U (2005) Optimal two-stage algorithms for group testing problems. SIAM J Comput 34(5):1253–1270

Dorfman R (1943) The detection of defective members of large populations. Ann Math Stat 14:436–440

Du DZ, Hwang FK (2000) Combinatorial group testing and its applications. World Scientific, Singapore

Du DZ, Hwang FK (2006) Pooling designs and nonadaptive group testing. World Scientific, Singapore

Dyachkov AG, Rykov VV (1983) A survey of superimposed code theory. Probl Control Inf Theory 12(4):1–13

Dyachkov AG, Macula AJ, Torney DC, Vilenkin PA (2001) Two models of nonadaptive group testing for designing screening experiments. In: Proceedings of the 6th international workshop on model-oriented designs and analysis, pp 63–75

Dyachkov AG, Vilenkin P, Macula A, Torney D (2002) Families of finite sets in which no intersections of ℓ sets is covered by the union of s others. J Comb Theory Ser A 99:195–218

Erdös P, Frankl P, Füredi Z (1985) Families of finite sets in which no set is covered by the union of r others. Israel J Math 51:75–89

Farach M, Kannan S, Knill EH, Muthukrishnan S (1997) Group testing with sequences in experimental molecular biology. In: Carpentieri B, De Santis A, Vaccaro U, Storer J (eds) Proceedings of compression and complexity of sequences. IEEE Computer Society, Los Alamitos, pp 357–367

Hong EH, Ladner RE (2002) Group testing for image compression. IEEE Trans Image Process 11:901–911

Hong YW, Scaglione A (2004) On multiple access for distributed dependent sensors: a content-based group testing approach. In: IEEE information theory workshop, pp 298–303

Hwang FK, Liu YC (2003) Error-tolerant pooling designs with inhibitors. J Comput Biol 10(2):231–236

Indyk P (1997) Deterministic superimposed coding with application to pattern matching. In: Proceedings of thirty-ninth IEEE annual symposium on foundations of computer science (FOCS 97), pp 127–136

Indyk P (2002) Explicit constructions of selectors and related combinatorial structures, with applications. In: Proceedings of symposium on discrete algorithms 2002 (SODA 2002), pp 697–704

Kautz WH, Singleton RR (1964) Nonrandom binary superimposed codes. IEEE Trans Inf Theory 10:363–377

Knill E (1995) Lower bounds for identifying subset members with subset queries. In: Proceedings of symposium on discrete algorithms 1995 (SODA 1995), pp 369–377

Kumar R, Rajagopalan S, Sahai A (1999) Coding constructions for blacklisting problems without computational assumptions. In: Proceedings of CRYPTO '99. Lecture notes in computer science, vol 1666. Springer, Berlin, pp 609–623

Li CH (1962) A sequential method for screening experimental variables. J Am Stat Assoc 57:455–477

Lovàsz L (1975) On the ratio of optimal integral and fractional covers. Discret Math 13:383–390

Margaritis D, Skiena S (1995) Reconstructing strings from substrings in rounds. In: Proceedings of thirty-seventh IEEE annual symposium on foundations of computer science (FOCS 95), pp 613–620

Ngo HQ, Du D-Z (2000) A survey on combinatorial group testing algorithms with applications to DNA library screening. In: Discrete mathematical problems with medical applications. DIMACS series in discrete mathematics and theoretical computer science, vol 55. American Mathematical Society, Providence, pp 171–182

Pevzner PA, Lipshutz R (1994) Towards DNA sequencing chips. In: 19th international conference on mathematical foundations of computer science. Lecture notes in computer science, vol 841. Springer, Berlin, pp 143–158

Sobel M, Groll PA (1959) Group testing to eliminate efficiently all defectives in a binomial sample. Bell Syst Tech J 38:1179–1252

Stinson DR, Wei R, Zhu L (2000a) Some new bounds for cover-free families. J Comb Theory Ser A 90:224–234

Stinson DR, van Trung T, Wei R (2000b) Secure frameproof codes, key distribution patterns, group testing algorithms and related structures. J Stat Plan Inference 86:595–617

Wolf J (1985) Born again group testing: multiaccess communications. IEEE Trans Inf Theory 31:185–191