Discrete Applied Mathematics 205 (2016) 208–212

Contents lists available at ScienceDirect

Discrete Applied Mathematics

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

Note

Adaptive group testing with a constrained number of positive responses improved

Peter Damaschke *

Department of Computer Science and Engineering, Chalmers University, 41296 Göteborg, Sweden

ARTICLE INFO

Article history:
Received 23 April 2015
Received in revised form 28 November 2015
Accepted 14 January 2016
Available online 8 February 2016

Keywords:
Group testing
Positive test
Adaptive strategy
Randomized strategy

ABSTRACT

Group testing aims at identifying the defective elements of a set by testing selected subsets called pools. A test gives a positive response if the tested pool contains some defective elements. Adaptive strategies test the pools one by one. Assuming that only a tiny minority of elements are defective, the main objective of group testing strategies is to minimize the number of tests. De Bonis introduced in COCOA 2014 a problem variant where one also wants to limit the number of positive tests, as they have undesirable side effects in some applications. A strategy was given with asymptotically optimal test complexity, subject to a constant factor. In the present paper we reduce the test complexity, making also the constant factor optimal in the limit. This is accomplished by a routine that searches for a single defective element and uses pools of decreasing sizes even after negative responses. An additional observation is that randomization saves a further considerable fraction of tests compared to the deterministic worst case, if the number of permitted positive responses per defective element is small.

1. Introduction

Group testing aims at identifying the defective elements of a set by testing selected subsets called pools. That is, some unknown elements are defective, and a test gets a positive response if and only if the tested pool contains at least one defective element. Group testing is one of the most extensively studied combinatorial search problems. It has a history dating back to at least 1943, and has various modern applications including molecular biology [3,4], fault detection [5,8], conflict resolution [2], data compression [6], and computer security [10], to mention a few.

Usually the main goal is to find all defectives after a minimum number of tests. Countless variations of group testing differ in the assumptions on the number of defectives, the number of rounds of parallel tests, and restrictions on the choice of pools. We must refrain from even a cursory overview. Given the wealth of results it is amazing that only recently [1] a model has been introduced where one also limits the number of positive tests. The motivation is that positive tests can have undesired side effects. As an example, the defective elements could be radioactive sources or toxic substances, hence one wants to limit exposure to them. While, for trivial reasons, some positive tests are necessary to solve the search problem, their number should be kept small.

A test strategy is called adaptive if the tests are performed sequentially, hence the choice of the next test may depend on all earlier responses. Throughout the paper we use the following symbols for the parameters of interest.
n: number of elements
d: number of defective elements

* Tel.: +46 31 772 5405; fax: +46 31 772 3663.
E-mail address: ptr@chalmers.se.

http://dx.doi.org/10.1016/j.dam.2016.01.010

t: number of tests
y: number of positive ("yes") tests.

More precisely, n is the given number of elements, and d is a previously known upper bound on the number of defectives to expect (unless said otherwise). Whether t and y denote exact or expected numbers, upper bounds, or worst-case lower bounds will be clear from context or explicitly mentioned. Logarithms are base 2. To avoid heavy notation, rounding brackets and lower-order terms are omitted in expressions, as long as this does not affect their asymptotic behaviour. Symbol e denotes Euler's number $2.718\ldots$.

It is folklore in the field that $t > d\log(n/d)$ is the information-theoretic lower bound on the worst-case number of tests, and some simple group testing strategies need essentially this number of tests only, when a bound d is known beforehand. In the model with limited y we focus on the most relevant case where the allowed y is a small fraction of t. (Any limit $y > t/2$ is not a severe restriction, as then we are almost back to the ordinary group testing problem.) To have a predefined limit we may assume that y is smaller than half the trivial information-theoretic lower bound on t.

Due to a simple argument [1], any strategy for d defectives must admit $y \geq d$ positive tests. Next it is shown in [1] that any strategy needs

$$y > \frac{d\log(n/d)}{\log(et/y)}$$

where t is the actual number of tests performed. For any prescribed y we solve this inequality for t and obtain

$$t > \frac{y}{e}\left(\frac{n}{d}\right)^{\frac{d}{y}}.$$

Defining $f := y/d$ (note that $f \geq 1$) this becomes

$$t > \frac{fd}{e}\left(\frac{n}{d}\right)^{\frac{1}{f}}.$$

Moreover, a strategy is provided in [1] that needs

$$t < fd\left(\frac{n}{d}\right)^{\frac{1}{f}} + fd$$

tests, at least for integer-valued f. Notice that there remains, essentially, a multiplicative gap of e. The algorithm in [1] builds upon Li's classic algorithm [7]. It works in f stages of tests that can be performed in parallel. Apart from minor technicalities, every stage partitions the remaining elements (that might still be defective) into the same number of disjoint pools of equal sizes. Only the positive pools in a stage need to be further searched from the next stage on. But when a strategy is adaptive anyway, it does not seem optimal to use equally sized pools within a stage. Intuitively it would be better to let the pool sizes strictly decrease even after negative tests, because the earlier a positive pool is encountered in a sequence, the more tests are still available to find a defective element therein. We will derive pool sizes that make optimal use of this idea. Interestingly, this improvement suffices to close the multiplicative gap, as we will see. This also establishes asymptotic tightness, that is, subject to a factor $1 + o(1)$, of the lower bound from [1].

Our second contribution is a simple randomized version of the strategy. A thorough analysis turns out to be intricate, but we get a few partial results implying that the expected test number is considerably smaller than the worst-case test number in the case of small y/d. We end with a conjecture about the optimal randomized test number.

2. An asymptotically optimal adaptive strategy

Lemma 1. *Suppose that a given set of n elements is already known to contain some defective elements, and we want to identify one of them, permitting at most y positive responses. This can be accomplished with fewer than $n^{1/y}(y/e)V + y$ tests, where V is some term that goes to 1 as y grows.*

Proof. For $t \geq y$ we define $N(t, y)$ to be the largest n such that it is possible to find one defective out of n elements using at most t test at most y of which may be positive. In the case of $t < y$ we define $N(t, y) := N(t, t)$.

Note that any sequential strategy keeps on testing certain pools until the first positive response is seen. Without loss of generality these pools are pairwise disjoint, since elements in negative pools need not be tested again. Upon a positive test we know that the tested pool has some defective element, and it remains to solve the problem restricted to this pool.

If $y = 1$, we cannot afford testing pools with more than one element, since after a positive response we have already used up the positive answer but have not yet identified a defective element. It follows immediately $N(t, 1) = t + 1$. (If t tests have been negative, we know without testing that the last element is defective.)

Now consider any $y > 1$. The ith pool can have $N(t - i, y - 1)$ elements. Indeed, if the ith pool is positive, we have performed i tests one of which was positive, thus we can still solve the residual problem on the ith pool. The last pool, with