Seventh International Workshop on Optimal Codes and Related Topics
September 6-12, 2013, Albena, Bulgaria *pp. 13-18*

Optimal strategies for a model of combinatorial two-sided search

H. Aydinian ayd@math.uni-bielefeld.de
University of Bielefeld, Department of Mathematics, GERMANY
F. Cicalese cicalese@dia.unisa.it
University of Salerno, ITALY
C. Deppe[1] cdeppe@math.uni-bielefeld.de
University of Bielefeld, Department of Mathematics, GERMANY
V. Lebedev[2] lebedev37@mail.ru
Institute for Information Transmission Problems of the Russian Academy of Sciences (Kharkevich Institute), RUSSIA

Dedicated to the memory of Professor Stefan Dodunekov

Abstract. We introduce a new combinatorial search problem in networks. This search model can be viewed as an adaptive group testing in a graph, where a searching object, or target, occupies one of the vertices. However, unlike standard group testing problems, the target in our model can move to an adjacent vertex once after each test. The problem is to find the location of the target, with a certain accuracy, using minimum number of binary tests applied on the subsets of vertices of the underlying graph. In this paper we consider cycles and paths as underlying graphs. We give optimal search strategies for isolation of the target within a subset of vertices of a given size. We also considered a restricted case of the problem, when the number of moves of the target is limited. Finally we present a coding analogue of the problem.

1 Introduction

Problems involving search arise in various areas of human activity. The first developments in search theory were made by Bernard Koopman and his colleagues during World War II. The purpose was to provide efficient ways to search for enemy submarines. The work done from 1942 to 1945 was published later (1946) in a book [9].

Basically, all search problems have two elements in common: a *hidden object*, in the broad sense of something being searched for, and a *searcher*. Search theory deals with the problem faced by a searcher: finding a hidden object,

[1]Supported by the DFG in the project "General Theory of Information Transfer"
[2]Supported in part by the Russian Foundation for Basic Research, project no 12-01-00905 and 13-07-00978.

in a given *search space*, in minimum time. In most of early developments it is assumed that a searching object is stationary and hidden according to a known distribution or it is moving and its motion is determined, by some known rules. This model of search is called one-sided search. In case the target cares about when he is found and reacts in any intelligent way to the searcher, the problem is called two-sided search.

A combinatorial search problem is considered in a discrete space and consists of finding a set of items in it satisfying specified requirements. Nowadays, combinatorial search involves an extensive number of challenging optimization problems which come directly from practical applications. The fundamentals of combinatorial search can be found in primary books [1], [3], [8].

During the Workshop "Search Methodologies II" (2012) Rudolf Ahlswede suggested to consider some combinatorial models of two-sided search for a moving object. This inspired us to introduce the following two-sided search model.

We define our search space $\mathcal{N} = \{1, 2, \ldots, N\}$ as the vertices of a graph $G = (\mathcal{N}, \mathcal{E})$. A searching object, also called a target, occupies one of those vertices unknown to the searcher. The searcher is able to detect the presence of the target at any subset of $\mathcal{N}$, i.e. for any subset $\mathcal{T} \subset \mathcal{N}$, called a test set, the searcher can learn whether the target is located at $\mathcal{T}$ or not. The goal is to find the location of the target, with a certain accuracy, in minimum time (number of tests). Note that in case of stationary target, the problem is equivalent to a classical group testing problem (see [7] for more awareness) of finding a single defective item.

We consider the following model of search for a moving target. After each test, the target can move once to an adjacent vertex or stay at the same place. For ease of description we assume that each vertex in our graph G has a loop. Thus, we may assume w.l.o.g. that in each time unit the target moves to an adjacent vertex. Furthermore, starting at vertex d_1, after n time units the target commits a walk $d^{n+1} := (d_1, \ldots, d_{n+1}) \subset \mathcal{N}^{n+1}$; $(d_i, d_{i+1}) \in \mathcal{E}$ $(i = 1, \ldots, n)$. Thus, vector d^{n+1} shows that the target occupies the vertex d_j at time j. Recall that in case of classical group testing we have $d_i = d_j$; $1 \leq i, j \leq n + 1$.

Next we give a formal description of our adaptive search model. Let $d_1 \in \mathcal{N}$ be the initial unknown position of the target and let $(\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n)$ be a sequence of test sets $\mathcal{T}_i \subset \mathcal{N}$ (tests for short) performed one after another at a time. Let also $(d_1, \ldots, d_{n+1})$ be the corresponding unknown walk performed by the target. For each test $\mathcal{T}_i$ we define the test function

$$f_{\mathcal{T}_i}(d_i) = \begin{cases} 0 & , \text{ if } d_i \notin \mathcal{T}_i \\ 1 & , \text{ if } d_i \in \mathcal{T}_i. \end{cases}$$

A sequence $(\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_n)$ is called a sequential or adaptive strategy of length n, if the result of each test $\mathcal{T}_i$ $(i = 1, \ldots, n-1)$, defined by $f_{\mathcal{T}_i}(d_i)$, can be used for the next test $\mathcal{T}_{i+1}$.

We denote by $\mathcal{D}_i$ the set of possible positions of the target after the ith test, thus $\mathcal{D}_0 = \mathcal{N}$ and for $i = 1, \ldots, n$ we have

$$\mathcal{D}_i = \begin{cases} \Gamma(\mathcal{T}_i) & , \text{ if } f_{\mathcal{T}_i}(d_i) = 1 \\ \Gamma(\mathcal{D}_{i-1} \backslash \mathcal{T}_i) & , \text{ if } f_{\mathcal{T}_i}(d_i) = 0, \end{cases}$$

where $\Gamma(\mathcal{A}) := \{j \in \mathcal{N} : \exists i \in \mathcal{A} \text{ with } (i,j) \in \mathcal{E}\}$ is the neighborhood of a subset $\mathcal{A} \subset \mathcal{N}$. Thus, we can define $\mathcal{D}_i$ as the search space after ith test.

Given a graph $G = (\mathcal{N}, \mathcal{E})$, a strategy of length n is called (G, s)–successful if $|\mathcal{D}_i| \leq s$ for some $i \leq n$. Let $s^*(G)$ be the minimal number s^* such that there exists a (G, s^*)–successful strategy. Given an integer $s \geq s^*(G)$, we denote by $n(G, s)$ the minimum number n such that there exists a (G, s)–successful strategy of length n. The corresponding strategy is called then an optimal (G, s) strategy. To the best of our knowledge, this problem was not studied earlier in the literature.

We also consider a more general problem when the mobility of the target is limited, namely the target can move at most t times, $t < |\mathcal{N}|$. In this case we use the corresponding notation (G, s, t)–successful and $n(G, s, t)$. Recall that we consider the worst case analysis, i.e. the goal of the target is to maximize the length of the strategy. In this paper we consider undirected cycle and paths as underlying graphs. We denote by $C_N = (\mathcal{N}, \mathcal{E})$ the cycle graph on N vertices, where $\mathcal{E} = \{(i, i+1) : i = 1, \ldots, N-1\} \cup \{N, 1\} \cup \{(i,i) : i = 1, \ldots, N\}$ and by $L_N = (\mathcal{N}, \mathcal{E})$ the path (also called linear) graph on N vertices with $\mathcal{E} = \{(i, i+1) : i = 1, \ldots, N-1\} \cup \{(i,i) : i = 1, \ldots, N\}$.

The paper is organized as follows: In Section 2 we give an optimal (C_N, s)–strategy for $s \geq 5$ and any cycle C_N. We also show that $s^*(C_N) = 5$ for $N \geq 5$. For a path L_N we give an optimal $(L_N, 4)$ strategy, which is linear in N. For $s \geq 5$ we give an optimal (L_N, s) strategy, which is logarithmic in N. In Section 3 we consider the problem for cycles and paths, when a target can move at most t times. We give optimal $(C_N, 3, t)$–strategies for $t = 1, 2$ and for $t \geq 3$ we give a general strategy. Finally, we give an optimal $(L_N, 3, 1)$-strategy and a general strategy for $s \geq 5$. In Section 4 we present a new coding problem, which is a twin to our search problem. We conclude the paper with some directions for future research. Due to lack of space we skip some of proofs.

2 Optimal strategies for cycles and paths

We consider only two classes of graphs: cycles and paths on N vertices. We introduce two new notation. Given integers $n, s \geq 1$, we denote by $N_c(n, s)$ resp. $N_l(n, s)$ the maximal number N, such that there exists a (C_N, s)–successful resp. (C_L, s)–successful strategy. Afterwards, for ease of exposition, we present our results in terms of $N_c(n, s)$ resp. $N_l(n, s)$ rather than $n(C_N, s)$ resp. $n(L_N, s)$. We start with a simple observation for cycles.

Proposition 1. *For $N \geq 5$ we have $s^*(C_N) = 5$.*

Theorem 1. *For $n \geq 0$ we have*

$$N_c(n, 5) = 2^n + 4.$$

Proof. We proceed by induction on n. For $n = 0$ the statement is obvious.
$n-1 \to n$: Suppose $N_c(n, 5) > 2^n + 4$ and $(\mathcal{T}_1, \ldots, \mathcal{T}_n)$ is an optimal strategy. Observe that in the worst case $|\mathcal{D}_1| \geq \lceil N_c(n, 5)/2 \rceil + 2$, with the equality iff $\mathcal{T}_1$ is a path in $\mathcal{C}_N$. This implies $|\mathcal{D}_1| > (2^n + 4)/2 + 2 = 2^{n-1} + 4$, a contradiction with the induction hypothesis $|\mathcal{D}_1| \leq N_c(n-1, 5) = 2^{n-1} + 4$. Hence we have $N_c(n, 5) \leq 2^n + 4$. On the other hand, by the induction hypothesis, taking a path $\mathcal{T}_1$ with $|\mathcal{T}_1| = \frac{N}{2}$, is sufficient (and necessary) to get an optimal strategy. □

Using the same approach we can extend the result to arbitrary $s \geq 5$.

Theorem 2. *For $s \geq 5$ and $n \geq 0$ we have*

$$N_c(n, s) = (s - 4)2^n + 4.$$

Next we consider a path as an underlying graph.

Proposition 2. *For $N \geq 4$ we have $s^*(C_L) = 4$.*

Lemma 1. *For $n \geq 0$ we have*

$$N_l(n, 4) = 2n + 4.$$

Theorem 3. *For integers $n \geq 0$ and $s \geq 4$ we have*

$$N_l(n, s) = (s - 4)2^n + 2n + 4.$$

Proof. In view of Lemma 1 we consider now the case $s \geq 5$. We denote by a_i the maximal number such that $\mathcal{D}_{n-i} = \{1, 2, \ldots, a_i\}$ in a (L_N, n)-successful strategy. It is not hard to show that $a_i = (a_{i-1} - 1) + (N_c(i-1, s) - 2)$. Our goal now is to show that $a_i = (s-4)2^i + i + 4$, using induction on i. It is clear that $a_0 = s$ and for $i - 1 \to i$ we have
$a_i = ((s-4)2^{i-1} + (i-1) + 4 - 1) + ((s-4)2^{i-1} + 4 - 2) = (s-4)2^i + i + 4.$
Since $N_l(n, s) = 2(a_{n-1} - 1)$ the statement follows. □

3 Optimal strategies for the restricted case

In this section we consider the case when the target can move at most t times. Thus, there are most $t + 1$ different vertices in any walk of the target. We consider the problem first for cycles C_N.

Theorem 4. *For integers $s \geq 5$, $1 \leq t < n$ we have*

$$N_c(n, s, t) \geq (s - 4)2^n + 4 + 4(2^{n-t} - 1).$$

In view of Theorem 2 we get the following

Corollary 1. *For integers $s \geq 5$, $1 \leq t < n$ we have*

$$N_c(n, s, t) \geq N_c(n, s) + 4(2^{n-t} - 1).$$

We consider now the cases $t = 1, 2$.

Theorem 5.

(i) For $n \geq 1$ and $s \geq 3$ we have $N_c(n, s, 1) = (s - 2)2^n$.

(ii) For $n \geq 4$ we have $N_c(n, 3, 2) = 2^{n-2}$.

Next we consider the restricted case for paths.

Theorem 6. *For integers $s \geq 5$, $1 \leq t < n$ we have*

$$N_l(n, s, t) \geq (s - 4)2^n + 2t + 2^{n-t+2}.$$

Theorem 7. *For $s \geq 3$ we have*

$$N_l(n, s, 1) = (s - 2)2^n + 2.$$

4 A new coding model and concluding remarks

Many adaptive search problems can be formulated as a coding problem. Already in 1964 Berlekamp [4] showed that the problem of searching for one element with at most e wrong answers is equivalent to construction of an e–error correcting code with feedback. More examples can be found in [2], [5]. [6].

We describe now a coding problem which is equivalent to our two-sided search model. Let $\mathcal{N} = \{1, 2, \ldots, N\}$ be a set of messages, which we identify with the vertices of an undirected graph $G = (\mathcal{N}, \mathcal{E})$. A source chooses a message $d_1 \in \mathcal{N}$ which the transmitter should transmit by sending at most $n(G)$ binary symbols (bits) step by step (adaptively) over a noiseless binary channel. However, after every transmission of one bit, the source may change the message into a neighboring message. The sequence of vertices $d_1, \ldots, d_j$ describes an alteration, after j transmissions with the actual message d_j. Let $(c_1, \ldots, c_{j-1}) \in \{0, 1\}^{j-1}$ be the submitted sequence of the sender. Then the jth bit c_j depends on the actual message d_j and the $j - 1$ submitted bits, so that $c_j = c_j(c_1, \ldots, c_{j-1}, d_j)$.

The goal is to describe an efficient scheme of transmission such that for every walk $d_1, \dots, d_{n+1}$ the receiver is able to find a set $S \subset \mathcal{N}$, of a given size s, which includes a message d_{j+1}, after $j \leq n$ transmissions.

It can be seen that this setting of the problem is equivalent to our search problem. On the other hand we note that from the coding point of view it seems more natural to consider the following problem: the goal is to find a set of size s containing the message d_{n+1}.
We emphasize that for cycles and paths the answers for both problems are the same.

We have considered a two-sided combinatorial search problem for two classes of underlying graphs, cycles and paths, with the most simple topologies. In fact, the problem essentially depends on the topology of the underlying graph. It is natural to consider the problem for other popular topologies like grids, trees, n-cubes etc. Another direction for future research is to consider probabilistic models of the problem.

References

[1] R. Ahlswede, I. Wegener, Suchprobleme, Teubner, 1979, (English translation, Search problems, Wiley, 1987).

[2] R. Ahlswede, C. Deppe, V. Lebedev, Nonbinary error correcting codes with noiseless feedback, localized errors or both, *Annals of European Academy of Sciences*, No. 1, 285 - 309, 2005.

[3] M. Aigner, *Combinatorial Search*, John Wiley and Sons, 1988.

[4] E. Berlekamp, *Block Coding with Noiseless Feedback*, PhD thesis, MIT EE Dept., 1964.

[5] F. Cicalese, *Fault-Tolerant Search Algorithms*, Springer-Verlag, 2013.

[6] C. Deppe, Searching with lies and coding with feedback, *Entropy, Search, Complexity, Bolyai Society Mathematical Studies*, Katona (Ed.), Vol. 16, 27–70, 2007.

[7] D. Du, F. Hwang, Combinatorial group testing and its applications, Series on Applied Mathematics, 1993.

[8] D. E. Knuth, *The Art of Computer Programming*, vol. 3, Addison-Wesley Publishing, 1973.

[9] B. Koopman, *Search and Screening*, Persimmon Press, 1th edition, 1946.