J. N. Srivastava et al., eds., *A Survey of Combinatorial Theory*
© North-Holland Publishing Company, 1973

CHAPTER 23

Combinatorial Search Problems

G. O. H. KATONA

Mathematical Institute, Hungarian Academy of Sciences, Budapest, Hungary

1. Introduction

The basic problem is the following: *We have a finite set* $X = \{x_1, \ldots, x_n\}$ *and we want to identify an unknown element* x_i *of* X *testing some subsets* A *of* X *whether* A *contains* x_i *or not.*

There are many practical problems of this type. The first one (known from mathematical problems) is the following (see Dorfmann [1943] and Sterrett [1953]):

1. "*Wasserman-type*" *blood test of a large population.* X is the set of some men. The test may be divided conveniently into two parts: (1) A sample of blood is drawn from every man. (2) The blood sample is subjected to a laboratory analysis which reveals the presence or absence of "syphilitic antigen". The presence of syphilitic antigen is a good indication of infection; for the second step, instead of carrying out the test individually we can pour together some samples. Carrying out the second step on the mixture we may determine whether the given subset of men contains an infected man or not.

2. *Diagnosis of a sick TV set.* X is the set of parts of the TV set. First we see that there is a good picture. The trouble must be in the "sound-channel", which is a subset of the set of parts of the TV set. Similarly, by different tests we can determine whether certain subsets contain the ill part or not.

3. *Chemical analysis.* Assume we have an unknown chemical element and we want to identify it. X is the set of chemical elements. We pour some chemical to the unknown one; if its colour turns red, we know that it belongs to a subset of the set of chemical elements; in the contrary case, it does not. After carrying out some such tests, we can identify the unknown element.

4. *Defective coin problem.* X consists of 27 coins, one of them is defective. The defective coin is heavier than the good ones. We have an equal arm balance, and we want to identify the defective coin by weighings. If we put on the balance two sets of coins of equal size, then we can see which one contains the defective coin, and if they are equally heavy then the remaining set must contain it. In the previous examples we divided the set X into two subsets (A and its complement $X-A$). However, in this case we divide X into three disjoint subsets, and after the weighing we know which one

contains the unknown defective coin. Thus, this problem is generalization of the original problem.

The above examples differ in many things.

(A) (α) In the 3rd and 4th (and probably in the 2nd) example there is exactly one unknown defective† element.

(β) In the 1st example the elements may be infected independently with equal probability. It may occur that all the persons are infected or that all of them are healthy.

(B) In these examples the next subset may be (α) dependent or (β) independent on the answers of the previous tests. If the person or the machine performing the tests has a sufficiently large memory, then it may depend on the answers; in the contrary case it may not.

(C) (α) In the 1st example we may choose any subsets for test. (β) However, in the cases of the 2nd, 3rd and 4th examples, the electrical construction, the chemical properties and the condition that two subsets of three parts are equally sized produce restrictions for choosing sets.

(D) (α) In the 1st, 2nd and 3rd example we test a subset of X; in other words, we divide X into two subsets (A and $X-A$). The answer says which one contains the (or an) unknown element.

(β) In the 4th example we divide into three parts. Practically, in the 3rd example we always divide into many parts; pouring the testing chemical we can get many different colours. From the colour we may determine to which subset the unknown element belongs. The number of subsets may change from step to step.

(E) Our aim (in all the cases) is to minimize either (α) the average number of tests, or (β) the maximal number of tests.

There are many other different questions. We do not want to list all of them in order not to frighten away the reader. We shall investigate some of them later. There is one more reassuring fact: We do not know the solutions of all the problems obtained by combination of the cases (A), (B), (C) and (D).

We shall not investigate three kinds of problems: (1) the method "element by element"; p_i is the probability of x_i being wrong, c_i is the cost of testing x_i, determine the optimal order of the tests; (2) the case in which X is infinite; (3) sequential decoding of information theory. Problem 1 has no combinatorial aspects, problem 2 has some, but its methods are rather analytical. Finally, problem 3 has some connections with problems treated in this survey paper; however, these problems are very involved and the connections are not clear yet.

Let us first examine (for warming up) a trivial case: (Aα), (Bβ), (Cα), (Dα), (Eα) = (Eβ). We have a finite set $X = \{x_1, \ldots, x_n\}$ and exactly one

† Sometimes we say briefly "unknown element" or "unknown".

unknown element x_i. We have to determine a family $A_1, \ldots, A_m$ of subsets in such a way, that

after knowing whether $A_1, \ldots, A_m$ contains x_i or not we can determine x_i. (1.1)

(Bβ) means that we test all A_j's independently of the answers; (Cα) means that we can use any subset of X for A_j's. The number of tests does not depend on the unknown element x_i; it is m. Thus, we have to minimize m under the condition (1.1), where the A_j's run over all the subsets of X.

Put $B_j^1 = A_j$ and $B_j^2 = X - A_j$ $(1 \leqslant j \leqslant m)$. If we know whether A_j $(1 \leqslant j \leqslant m)$ contains x_i or not, we also know whether $B_1^{i_1} B_2^{i_2} \cdots B_m^{i_m}$ $(i_1, \ldots, i_m = 1$ or 2) contains x_i or not. These sets are disjoint for different sequences $i_1, \ldots, i_m$. Conversely, if we know which $B_1^{i_1} \cdots B_m^{i_m}$ contains x_i, then we know whether A_j $(1 \leqslant j \leqslant m)$ contains x_i or not (depending on i_j). Thus, (1.1) is equivalent with the condition that

$$B_1^{i_1} B_2^{i_2} \cdots B_m^{i_m} \text{ contains at most 1 element for each } i_1, \ldots, i_m, \tag{1.2}$$

and if we write $i_j = 1$ if $x_i \in A_j$ and $i_j = 2$ if $x_i \notin A_j$ then $B_1^{i_1} \cdots B_m^{i_m}$ is the unknown element.

Moreover, (1.2) is equivalent to the following condition:

For each pair x_j, x_k $(j \neq k)$ there is an A_l such that

$$x_j \in A_l \quad \text{and} \quad x_k \notin A_l$$

or

$$x_j \notin A_l \quad \text{and} \quad x_k \in A_l. \tag{1.3}$$

Indeed, if (1.3) does not hold, then $x_j \in B_l^i$ and $x_k \in B_l^i$ are satisfied at the same time ($i = 1$ or 2). Choosing $i_1, \ldots, i_m$ in such a way that $x_j \in B_1^{i_1} \cdots B_m^{i_m}$, it has another element x_k, in contradiction with (1.2). Conversely, if (1.2) does not hold, then for some x_j, x_k $(j \neq k)$ and $i_1, \ldots, i_m$ we have $x_j, x_k \in B_1^{i_1} \cdots B_m^{i_m}$. In this case, $x_j \in B_l^{i_l}$ and $x_k \in B_l^{i_l}$, that is, $x_j \in A_l$ and $x_k \in A_l$ hold at the same time $(1 \leqslant l \leqslant m)$ in contradiction with (1.3).

We call a family of subsets $A_1, \ldots, A_m$ a *separating system* if they satisfy either (1.1) or (1.2) or (1.3).

There is a 4th characterization of separating systems. Define the 0, 1 matrix $M = (a_{ij})$ in the following way:

$$a_{ij} = 1 \quad \text{iff} \quad x_j \in A_i \quad (1 \leqslant i \leqslant m, \quad 1 \leqslant j \leqslant n).$$

Then (1.3) is equivalent to:

$$M \text{ has different columns.} \tag{1.4}$$

After these preliminary remarks, our first mathematical problem becomes very easy: *Given n, determine the minimal m such that there exists an $m \times n$ matrix with different columns.* The number of different columns is 2^m, thus $2^m \geqslant n$ necessarily holds. In other words $m \geqslant \log n$ (we shall always use logarithms with basis 2) or $m \geqslant \{\log n\}$, where $\{x\}$ denotes the least integer $\geqslant x$. This

estimation is best possible: choosing n columns arbitrarily from the different $2^{\{\log n\}}$ 0, 1 sequences, we obtain a good matrix M.

Theorem 1.1. *If X is a finite set of n elements, then the minimal separating system consists of $\{\log n\}$ elements.*

2. Connections with noiseless encoding

Let us restrict ourselves now to the case (Aα), (Bα), (Cα), (Dα), (Eα).

We have again a finite set $X = \{x_1, \ldots, x_n\}$; exactly one element x_i of X is defective (wrong, unknown) with probabilities $p_1, \ldots, p_n$. Further, there are subsets A_1, $A_j(e_1, \ldots, e_{j-1})$ where A_1 is the first test, and $A_j(e_1, \ldots, e_{j-1})$ $(1 < j \leqslant m;\ e_1, \ldots, e_{j-1} = 0 \text{ or } 1)$ is the j-th test when the answer of the previous tests were $e_1, \ldots, e_{j-1}$ ($e_k = 1$ means: it contains x_i; $e_k = 0$ means: it does not contain x_i).

$$\begin{aligned} &\text{If } A_{l+1}(e_1, \ldots, e_l) \text{ is not defined, but } A_l(e_1, \ldots, e_{l-1}) \text{ is} \\ &\text{then the answers } e_1, \ldots, e_{l-1}, e_l \text{ (together with the subsets} \\ &A_1, A_2(e_1), \ldots, A_k(e_1, \ldots, e_{l-1})) \text{ uniquely determine } x_i. \end{aligned} \tag{2.1}$$

We call such a family of subsets a *strategy*.

If we fix x_i for a moment, then the sequence $e_1(i), \ldots, e_{l_i}(i)$ of answers is uniquely determined ($(A_{l_i+1}(e_1(i), \ldots, e_{l_i}(i)))$ is not defined). The number of necessary tests is l_i. The average number of tests is

$$\sum_{i=1}^{n} p_i l_i. \tag{2.2}$$

We have to minimize (2.2) over the strategies, where the A's run over all the subsets of X.

Observe that in this way we corresponded a 0, 1 sequence $e_1(i), \ldots, e_{l_i}(i)$ with every x_i. This is a *code* in the language of information theory. The sequences are called codewords. This code has a simple property: There are no two different i and j such that $l_j \geqslant l_i$ and

$$e_1(i) = e_1(j), \ldots, e_{l_i}(i) = e_{l_i}(j).$$

In other words, no codeword is a *segment* of another one. We say that it is a *prefix code*.

This definition is adopted for the case when we use codewords formed from r different symbols $y_1, \ldots, y_r$ instead of 0 and 1.

Conversely, if we have a prefix code $x_i \to e_1(i), \ldots, e_{l_i}(i)$ formed from 0's and 1's, then we can define a strategy in the following way

$$\begin{aligned} A_1 &= \{x_i : e_1(i) = 1\} \\ A_j(e_1, \ldots, e_{j-1}) &= \{x_i : e_1(i) = e_1, \ldots, e_{j-1}(i) = e_{j-1}, e_j(i) = 1\} \\ &\qquad (j > 1) \end{aligned} \tag{2.3}$$

where $e_1, \ldots, e_{j-1}$ is a fixed sequence of 0's and 1's. If the set on the right hand side is empty, we do not define $A_i(e_1, \ldots, e_{i-1})$. For any fixed x_i we get the

results $e_1(i), \ldots, e_{l_i}(i)$, writing 1 if the testing subset contains x_i and 0 if not. (It is easy to see by induction.) After these l_i tests, x_i is uniquely defined by the prefix property of the code. Thus (2.3) is a strategy and we found a correspondence between the prefix codes and the strategies; moreover, this correspondence is length-preserving: the length of the codeword of x_i is equal to the number of tests necessary to identify x_i.

This correspondence was described by Sobel [1960] (cf. Section 12), by Picard [1965] and by Campbell [1968], and it may also have been known to earlier authors. However, the optimal prefix codes and optimal strategies do not coincide, as Sobel [1967] has noticed in his Appendix. He also investigated this connection in another paper (Sobel [1970]).

This correspondence allows us to use the following well known theorem of information theory:

Noiseless Coding Theorem. *If the symbols $x_1, \ldots, x_n$ are encoded by the symbols $y_1, \ldots, y_m$ in a prefix way, then*

$$L = \sum_{i=1}^{n} p_i l_i \geqslant \frac{-\sum_{i=1}^{n} p_i \log p_i}{\log m} = \frac{H(P)}{\log m} \tag{2.4}$$

where $P = (p_1, \ldots, p_n)$ $(p_i > 0, \sum p_i = 1)$ is the vector of probabilities of the symbols $x_1, \ldots, x_n$ and l_i is the length of the codeword of x_i.

On the other hand, we can always find a prefix code satisfying the inequality

$$L \leqslant \frac{H(P)}{\log m} + 1. \tag{2.5}$$

We do not prove it here. The reader can find it in any information-theoretical book (e.g. Feinstein [1958]).

Substituting $m = 2$, this theorem gives us good estimates for the minimum of the average test-number:

$$H(P) \leqslant L \leqslant H(P) + 1. \tag{2.6}$$

However, it remains an open question what is the exact minimum. To answer this question let us examine some simple properties of the (in average sense) shortest code. Assume $p_1 \geqslant \cdots \geqslant p_n$.

Lemma 2.1. *For the optimal prefix code, $l_1 \leqslant \cdots \leqslant l_n$.*

Proof. If there is a pair i, j such that $p_i > p_j$ and $l_i > l_j$, then changing the code words of x_i and x_j the average increases by $p_i l_j + p_j l_i - p_i l_i - p_j l_j = (p_i - p_j)(l_j - l_i)$ and this is negative. The lemma is proved.

Lemma 2.2. *If $l_i = l_n$, then $e_1(i), \ldots, e_{l_i-1}(i)$, $1 - e_{l_i}(i)$ is also a code word together with $e_1(i), \ldots, e_{l_i-1}(i), e_{l_i}(i)$.*

Proof. In the contrary case change the code word $e_1(i), \ldots, e_{l_i}(i)$ for $e_1(i), \ldots, e_{l_i-1}(i)$. The new word can not be a segment of another one (the only possibilities $e_1(i), \ldots, e_{l_i}(i)$ and $e_1(i), \ldots, 1 - e_{l_i}(i)$ are excluded). Conversely, any segment of the new code word is a segment of $e_1(i), \ldots, e_{l_i}(i)$

and this is impossible by the prefix property. Thus, the new code is prefix, too. The average code length is smaller; this is a contradiction. The proof is completed.

Denote by $L(p_1, \ldots, p_n)$ the average code length $\sum p_i l_i$ for a given code and by $L_{\min}(p_1, \ldots, p_n)$ its minimum for prefix codes. Let us consider a code with average code length $L_{\min}(p_1, \ldots, p_n)$. By Lemma 2.1, x_n has a code of maximal length: $e_1(n), \ldots, e_{l_n}(n)$. If we change its last element, then the new sequence is also a code word:

$$(e_1(n), \ldots, 1-e_{l_n}(n)) = (e_1(i), \ldots, e_{l_n}(i)) \qquad (i \neq n).$$

Here $l_i = l_n$, thus, again by Lemma 2.1, $l_i = l_{n-1} = l_n$. Changing the code words of x_i and x_{n-1}, the average code length does not change; we may assume $i = n-1$. Let us omit the code words of x_{n-1} and x_n and take a new one for both of them: $e_1(n), \ldots, e_{l_n-1}(n)$. This code is prefix again, and its average code length is smaller by $p_{n-1}+p_n$.

$$L_{\min}(p_1, \ldots, p_n) = L(p_1, \ldots, p_{n-1}+p_n)+p_{n-1}+p_n.$$

Hence

$$L_{\min}(p_1, \ldots, p_n) \geqslant L_{\min}(p_1, \ldots, p_{n-1}+p_n)+p_{n-1}+p_n \tag{2.7}$$

follows. On the other hand, given a code with average code length $L_{\min}(p_1, \ldots, p_{n-1}+p_n)$ then we can form a new prefix code writing 0 and 1 at the end of the code word with probability $p_{n-1}+p_n$. The average code length is enlarged by $p_{n-1}+p_n$:

$$L_{\min}(p_1, \ldots, p_n) \leqslant L_{\min}(p_1, \ldots, p_{n-1}+p_n)+p_{n-1}+p_n. \tag{2.8}$$

(2.7) and (2.8) result in

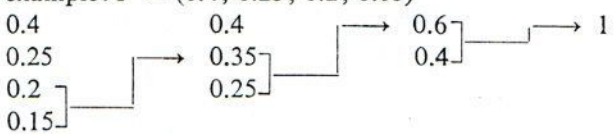
$$L_{\min}(p_1, \ldots, p_n) = L_{\min}(p_1, \ldots, p_{n-1}+p_n)+p_{n-1}+p_n. \tag{2.9}$$

We have the following important result:

Theorem 2.1. *We reach the optimal code with the following Huffman procedure: Assume that a code with average code length* $L_{\min}(p_1, \ldots, p_{n-1}+p_n)$ *is determined, where* p_{n-1} *and* p_n *are the two smallest probabilities. Write* 0 *and* 1 *at the end of the code word with probability* $p_{n-1}+p_n$. *This is the optimal code for* $P = (p_1, \ldots, p_n)$. *The optimal code for* $P = (1)$ *is the void sequence.*

This theorem was first proved by Huffman [1952], but it was independently found by Zimmerman [1959] in the language of search.

A simple example: $P = (0.4; 0.25; 0.2; 0.15)$



The code for (0.6; 0.4) is 0, 1.

The code for (0.4; 0.35; 0.25) is 1, 00, 1.

The code for (0.4; 0.25; 0.2; 0.15) is 1, 01, 000, 001.

Theorem 2.1 gives us the answer to our question. The next question arises: Is there any difference between search theory and noiseless code theory? The answer is clear: there are many differences.

1. Code theory does not give solutions for the problems of type (Aβ) or, in general, for the problems where there are two unknown elements with positive probability.

2. In case (Cβ) the possible restrictions for the testing subsets give restrictions for the corresponding codes. However, these restrictions are different from the usual restrictions of code theory.

3. Perhaps the most important difference is that at a noiseless channel we have many symbols to transmit. Thus, we consider the sequences of length N formed from $x_1, \ldots, x_n$ and we transmit these sequences as new symbols. By this method we may approximate the lower bound of (2.6) arbitrarily good. The Huffman procedure has less interest in this case. However, in the case of search we have usually only one set and one unknown. Here the Huffman procedure has a great interest.

In any case, if the code theorems do not give the exact solution of a search problem, they give (sometimes good) estimates.

We have to mention that in the case when we can divide the set by one test into m subsets, then we can also use the noiseless coding theorem and a modified form of the Huffman procedure.

3. Results

3.1. *Case* (Aα), (Bα), (Cα), (Dα), (Eα)

After these long preliminaries we start the real survey of results.

First consider the following problem: Just one of the elements $x_1, \ldots, x_n$ is defective with equal probability; what is the minimum of the average number of tests necessary to identify the defective element? This problem is obviously a particular case of the problem treated in the previous section. Theorem 2.1 gives an algorithm to determine $L_{\min}(1/n, \ldots, 1/n)$; however, in this special case we may determine the exact value.

Lemma 3.1. *The code words of the code having average length* $L_{\min}(1/n, \ldots, 1/n)$ *can have just two different lengths, which are consecutive integers.*

Proof. Assume $l_1 \leqslant \cdots \leqslant l_n$. If $l_1 \leqslant l_n - 2$ then consider the code word $e_1(n), \ldots, e_{l_n}(n)$. By Lemma 2.1 there exists a code word of the form $e_1(n), \ldots, 1 - e_{l_n}(n)$. Change the code words

$$\left.\begin{array}{l} e_1(n), \ldots, e_{l_n-1}(n), e_{l_n}(n) \\ e_1(n), \ldots, e_{l_n-1}(n), 1-e_{l_n}(n) \\ e_1(1), \ldots, e_{l_1}(1) \end{array}\right\} \quad \text{for} \quad \begin{cases} e_1(n), \ldots, e_{l_n-1}(n) \\ e_1(1), \ldots, e_{l_1}(1), 0 \\ e_1(1), \ldots, e_{l_1}(1), 1. \end{cases}$$

It is easy to see that the new code is prefix. However, the average code length is increased by $(2(l_1+1)+l_n-1)/n-(2l_n+l_1)/n = (l_1-l_n+1)/n$, which is negative by the assumption $l_1 = l_n-2$. The new code has a smaller average length. This is a contradiction. We proved $l_1 \geqslant l_n-1$. The proof is completed.

Choosing an arbitrary 0, 1 sequence $c_1, \ldots, c_{l_n-1}$ of length l_n-1, either it is a code word or one of the sequences

$$\begin{aligned} &e_1, \ldots, e_{l_n-1}, 0, \\ &e_1, \ldots, e_{l_n-1}, 1 \end{aligned} \tag{3.1}$$

is a code word. In the contrary case we would change a code word of length l_n for $c_1, \ldots, c_{l_n-1}$ preserving the prefix property and decreasing the average length. This is a contradiction. However, by Lemma 2.1 if one of the sequences (3.1) is a code word then the second one is also a code word. Thus either $e_1, \ldots, e_{l_n-1}$ or both of (3.2) are code words. Denoting by s the number of code words of length l_n-1 we have

$$s+\tfrac{1}{2}(n-s) = 2^{l_n-1}. \tag{3.2}$$

Here $0 \leqslant s < n$, and $\frac{1}{2}n \leqslant \frac{1}{2}(n+s)$. Using (3.2), we obtain the inequality

$$\tfrac{1}{2}n \leqslant 2^{l_n-1} < n,$$

or $\log n \leqslant l_n < \log n+1$. It results in $l_n = \{\log n\}$, where $\{x\}$ denotes the least integer $\geqslant x$. On the other hand, we may count s from (3.2):

$$s = 2^{\{\log n\}}-n.$$

The average is

$$\frac{s(\{\log n\}-1)+(n-s)\{\log n\}}{n} = \{\log n\}-\left(\frac{2^{\{\log n\}}}{n}-1\right).$$

Theorem 3.1.

$$L_{\min}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) = \{\log n\}-\left(\frac{2^{\{\log n\}}}{n}-1\right).$$

This theorem was first proved by Sandelius [1961]. Sobel [1968b] has it also as a by-product. The proof published here is different from both that of Sandelius and that of Sobel.

By this method it is easy to see the following generalizations (Katona and Lee):

Theorem 3.2. *Let $p_1 \geqslant \cdots \geqslant p_n$ be given probabilities, where $p_{n-1}+p_n \geqslant p_1$. Then*

$$L_{\min}(p_1, \ldots, p_n) = \{\log n\}-\sum_{i=1}^{s} p_i,$$

where $s = 2^{\{\log n\}}-n$.

Theorem 3.3. *If* $p_1 \geqslant \cdots \geqslant p_n$ *and* $p_n + kp_{n-1} > p_1$ *then*

$$l_n - l_1 \leqslant k;$$

that is, the number of different code lengths is at most $k+1$.

Sobel [1968b] determined another special case:

Theorem 3.4. *If*

$$p_i = \frac{i}{\frac{1}{2}n(n+1)} \quad (1 \leqslant i \leqslant n)$$

then† *for* $2^{[\log n]} \leqslant n < 3 \cdot 2^{[\log n]-1}$,

$$L_{\min}(p_1, \ldots, p_n) = ([\log n]+2) + \tfrac{1}{2}n(n+1)(3 \cdot 2^{2[\log n]-3} - 3 \cdot 2^{[\log n]-2}(2n+1))$$

and for $3 \cdot 2^{[\log n]-1} \leqslant n < 2^{[\log n]+1}$,

$$L_{\min}(p_1, \ldots, p_n) = ([\log n]+2) + \tfrac{1}{2}n(n+1)(3 \cdot 2^{2[\log n]-1} - 3 \cdot 2^{[\log n]-1}(2n+1)).$$

3.2. *Case* (Aα), (Bα), (Cα), (Dα), (Eβ)

In this case the probabilities do not play any role. We may choose them $p_1 = 1/n$ $(1 = i = n)$, and use Theorem 3.1:

$$L_{\min}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) = \{\log n\} - \left(\frac{2^{\{\log n\}}}{n} - 1\right).$$

However, the maximum $\geqslant$ the average. Denote by l the minimum of the maximal test number. Thus

$$l \geqslant \{\log n\} - \left(\frac{2^{\{\log n\}}}{n} - 1\right). \tag{3.3}$$

Here

$$1 \leqslant \frac{2^{\{\log n\}}}{n} < 2$$

and

$$\{\log n\} - \left(\frac{2^{\{\log n\}}}{n} - 1\right) > \{\log n\} - 1. \tag{3.4}$$

From (3.3) and (3.4) we obtain

$$l \geqslant \{\log n\}.$$

However, by Theorem 1.1 we can construct a strategy even by independent tests (case (Bβ)) with test length $\{\log n\}$.

Theorem 3.5. *Assume we have strategies for* $X = \{x_1, \ldots, x_n\}$. *The minimum (runs over strategies) of the maximal number of tests is*

$$l = \{\log n\}.$$

† $[x]$ denotes the largest integer $\leqslant x$.

It is quite interesting that in the case (Aα)(Cα)(Eβ) the optimality problems are equivalent for (Bα) and (Bβ); we do not obtain anything if we choose the next test depending on the answers of the previous tests.

3.3. *Case* (Aα), (Bβ), (Cα), (Dα), (Eα) = (Eβ).

Theorem 1.1 gives answer for this case.

3.4. *Case* (Aα), (Bα), (Cβ), (Dα), (Eα).

It is a very natural assumption that a linear order $x_1 < \cdots < x_n$ is given in X, and the admissible subsets are of type $\{x_1, \ldots, x_j\}$ or $\{x_j, \ldots, x_n\}$ $(1 \leqslant j \leqslant n)$.

For example, we want to classify apples according to their sizes; $x_1, \ldots, x_n$ are the classes, and the unknown x_i is the class of the given apple. We carry out the tests by holes. If the given apple falls through the hole then its class x_i belongs to $\{x_1, \ldots, x_h\}$ for some j depending on the hole. Conversely, if it does not fall through, then x_i belongs to $\{x_{j+1}, \ldots, x_n\}$.

What does this restriction mean on the language of codes? The tests $\{x_1, \ldots, x_j\}$ and $\{x_{j+1}, \ldots, x_n\}$ are equivalent; assume we use always the type $\{x_{j+1}, \ldots, x_n\}$. If $A_1 = \{x_{j+1}, \ldots, x_n\}$, then for the corresponding code we have

$$e_1(1) = \cdots = e_1(j) = 0, \qquad e_1(j+1) = \cdots = e_1(n) = 1.$$

Similarly, if $A_k(e_1, \ldots, e_{k-1}) = \{x_{l+1}, \ldots, x_n\}$ then considering the set $T = \{t : e_1(t) = e_1, \ldots, e_{k-1}(t) = e_{k-1}\}$ we have again

$$e_k(t) = 0 \qquad \text{if} \quad t \leqslant l, \quad t \in T$$

and

$$e_k(t) = 1 \qquad \text{if} \quad t > l, \quad t \in T.$$

Reformulating, it means that for any pair (t, u),

$$e_1(t) = e_1(u), \ldots, e_{k-1}(t) = e_{k-1}(u), e_k(t) = 0, e_k(u) = 1$$

for some k; that is, the code words are in *lexicographic* order. We say that the code is *alphabetical* if it possesses this property.

Our problem is to determine the prefix alphabetical code with minimal average length. Denote this average by $A_{\min}(p_1, \ldots, p_n)$.

Gilbert and Moore [1959] gave an efficient construction for alphabetical codes, which ensures the following estimation:

Theorem 3.6.

$$H(P) \leqslant A_{\min}(p_1, \ldots, p_n) \leqslant H(P)+2, \tag{3.5}$$

where $P = (p_1, \ldots, p_n)$.

Proof. The left hand side is a consequence of the left hand side of (2.6) and of the trivial inequality

$$L_{\min}(p_1, \ldots, p_n) \leqslant A_{\min}(p_1, \ldots, p_n). \tag{3.6}$$

We prove the right hand side of (3.5) by a construction. Define the numbers $q_1, \ldots, q_n$ and $l_1, \ldots, l_n$ as follows:

$$q_j = \sum_{i=1}^{j-1} p_i + \tfrac{1}{2}p_j, \qquad (3.7)$$
$$l_j = \{-\log p_j\} + 1.$$

Let the first l_j digits of the binary expansion of the number q_j be the code of x_j. If the prefix property does not hold, then the code of some x_i is a segment of the code of another x_j. It means that q_i and q_j have the same binary digits on the first l_i places. In other words,

$$|q_i - q_j| \leqslant \frac{1}{2^{l_i}} \leqslant \frac{1}{2^{-\log p_i + 1}} = \tfrac{1}{2}p_i,$$

and this contradicts (3.7), since

$$|q_i - q_j| \geqslant \tfrac{1}{2}p_i + \tfrac{1}{2}p_j > \tfrac{1}{2}p_i.$$

The constructed code is prefix, indeed. The alphabetical property is trivially satisfied. The average length is

$$\sum_{j=1}^{n} p_j l_j \leqslant \sum_{j=1}^{n} p_j(\{-\log p_j\} + 1) \leqslant \sum_{j=1}^{n} p_j(-\log p_j + 2) = H(P) + 2.$$

The proof is completed.

Knuth [1971] and further Hu and Tucker [1970] worked out algorithms to determine a good alphabetical code.

In the paper of Hu and Tucker the *tentative-connecting* algorithm is written down. This need not be directly associated with an alphabetical code, but it is proved that there exists an alphabetical code with the same code word lengths as the code generated by the tentative-connecting algorithm.

A code is equivalent to the following tree: The nodes are the different possible segments of the code words (including the void sequence, which is called *root*), and two nodes are connected if one of them is a segment of the other and their lengths differ one. The *terminal* nodes are the code words. The tentative-connecting algorithm determines the tree rather than the code.

We start the algorithm with the subtree consisting of the terminal nodes $c_1, \ldots, c_n$ with the given order (no edges). Every terminal node has weight p_j. We take the minimal sum of the form $p_j + p_{j+1}$ $(1 \leqslant j < n)$, we draw a new node d with weight $p_j + p_{j+1}$ and we connect d with c_j and c_{j+1}. We have a new subtree and a new *construction sequence*: $c_1, \ldots, c_{i-1}, d, c_{i+2}, \ldots, c_n$. In general, assume we have a subtree and its roots and terminal nodes form a construction sequence $d_1, \ldots, d_k$ (some of the d's are c's); they have weights $q_1, \ldots, q_k$. d_i and d_j $(i < j)$ are *tentative connected* if there is no d_k $(i < k < j)$ such that $d_k = c_l$ for some l. We form the minimal sum $q_i + q_j$ where d_i and d_j are tentative connected $(i < j)$. We connect the new code e with d_i and d_j.

The new construction sequence is $d_1, \ldots, d_{i-1}, e, d_{i+1}, \ldots, d_{j-1}, d_{j+1}, \ldots, d_k$. The corresponding weights are $q_1, \ldots, q_{i-1}, q_i+q_j, q_{i+1}, \ldots, q_{j-1}, q_{j+1}, \ldots, q_k$. We continue this procedure until the construction sequence consists of one element.

Observe that in this language, the Huffman algorithm means that we choose the minimal sum q_i+q_j without any restriction.

Notice that

$$A_{\min}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) = L_{\min}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right).$$

This is a consequence of the fact that in this case the average does not depend on the order.

By the methods of Lemma 3.1, the following theorem is easy to verify (Katona and Lee).

Theorem 3.7. *If $p_i+p_{i+1} > p_j$ $(1 \leqslant i < n,\ 1 \leqslant j \leqslant n)$ then the minimal alphabetic code can have only two different code word lengths, which are consecutive integers.*

It is an interesting question how the Huffman algorithm is modified if we have a prescribed bound b for the code lengths $(b > \{\log n\})$

$$l_i \leqslant b \qquad (1 \leqslant i \leqslant n);$$

but we are interested in the minimal average length. Cesari [1968] has a partial solution for the problem.

3.5. *Case* (Aα), (Bα), (Cβ), (Dα), (Eβ).

In this case the solutions of the problems of the preceding section are trivial and identical to Theorem 3.5.

However, there are other problems which are too difficult in the case of (Eα).

Suppose we have n coins $x_1, \ldots, x_n$, one of them being defective (say x_i). The weight w_i of a non-defective coin is

$$1 \leqslant w_i \leqslant 1+\delta$$

and $w_i = 1+\varepsilon$ $(\varepsilon > \delta)$. We can use scales (not equal arm balance), thus by one test we may determine the weight of a subset A of $\{x_1, \ldots, x_n\}$. If the number $|A|$ of elements of A is less than $[\varepsilon/\delta]$, then $x_i \in A$ if and only if the weight of the subset A is $\geqslant |A|+\varepsilon$, because in the contrary case its weight is less than $|A|+\delta[\varepsilon/\delta] \leqslant |A|+\varepsilon$.

This example raises the following problem. Suppose a finite set $X = \{x_1, \ldots, x_n\}$ is given. Determine the minimum of the maximal test number for strategies consisting of subsets of at most k elements (k is fixed $\leqslant n$).

Denote this minimum by $f_k(n)$. We know from Theorem 3.5 that

$$f_k(n) \geqslant \{\log n\}.$$

If $k \geqslant \frac{1}{2}n$, we do not have an essential restriction by $|A| \leqslant k$, for instead of

$|A| > k$ we can use the complement $X-A$, where $|X-A| < n-k \leqslant k$. Thus, by Theorem 3.5

$$f_k(n) = \{\log n\} \qquad \text{if} \quad [\tfrac{1}{2}n] \leqslant k. \tag{3.8}$$

Assume now $\frac{1}{2}n > k$. It is clear that $f_k(n)$ is a monotonically increasing function of n. Suppose for the optimal strategy $|A_1| = l$ $(1 \leqslant l \leqslant k)$ holds. If the subsets $A_j(e_1, \ldots, e_{j-1})$ form a strategy, then $A_1 \cap A_j(e_1, \ldots, e_{j-1})$ and $(X-A_1) \cap A_j(e_1, \ldots, e_{j-1})$ form a strategy on A_1 and $X-A_1$, respectively. Similarly, $|A_j(e_1, \ldots, e_{i-1})| \leqslant k$ results in $|A_1 \cap A_j(e_1, \ldots, e_{j-1})| \leqslant k$ and

$$|(X-A_1) \cap A_j(e_1, \ldots, e_{j-1})| \leqslant k.$$

For these strategies the maximal number of test is at least $f_k(l)$ and $f_k(n-l)$, respectively. We have the following inequality

$$f_k(n) \geqslant 1 + \max\,(f_k(l) + f_k(n-l)). \tag{3.9}$$

Here $l \leqslant n-l$ by $l \leqslant k$ and $k < \frac{1}{2}n$. Applying the monotonicity of $f_k(n)$ we have

$$\max\,(f_k(l), f_k(n-l)) = f_k(n-l) \tag{3.10}$$

and

$$f_k(n-l) \geqslant f_k(n-k). \tag{3.11}$$

Substitute (3.10) and (3.11) into (3.9):

$$f_k(n) \geqslant 1 + f_k(n-k). \tag{3.12}$$

Applying $v = \{n/k\} - 2$ times (3.12),

$$f_k(n) \geqslant v + f_k(n-kv)$$

follows. Here $n-kv \leqslant 2k$ is trivial; for the last term we can apply (3.8)

$$f_k(n) \geqslant v + \{\log\,(n-kv)\}.$$

However, it is easy to construct a strategy with this maximal test number; $A_1 = \{x_1, \ldots, x_k\}$, $A_2 = \{x_{k+1}, \ldots, x_{2k}\}, \ldots, A_v = \{x_{(v-k)k-1}, \ldots, x_{vk}\}$ are the first v tests. They are independent from the previous answers. After these tests we know that either $x_i \in A_j$ for some j $(1 \leqslant j \leqslant v)$ or $x_i \in \{x_{dk+1}, \ldots, x_n\}$ In the first case we have a strategy with maximal length $\{\log k\}$ to identify x_i by Theorem 3.5. In the second case we have a strategy with $\{\log\,(n-kv)\}$. Here $n-kv > k$, and the maximal length is $v + \{\log\,(n-kv)\}$. The conjecture of Vigassy is proved:

Theorem 3.8. *The minimum of the maximal test number of a strategy given to identify one of the n elements is* $v + \{\log\,(n-kv)\}$ *if the subsets used on the strategy can have at most k elements* $(k < n)$.

The next problem is a typical problem of computers: *Suppose there are given n numbers* $y_1, \ldots, y_n$ *whose values are unknown and pairwise unequal. We wish to order them using only binary comparisons.*

In other words we have an unknown permutation x_i from all the permutations $x_1, \ldots, x_{n!}$ of $y_1, \ldots, y_n$. The subsets we can use for tests consist of the permutations where y_i precedes y_j (for some fixed i and j $(i \neq j)$. There are $n!$ permutations, thus by Theorem 3.5 the minimum of the maximal test number is

$$l \geqslant \log(n!). \tag{3.13}$$

Steinhaus [1950] proposed the following algorithm: Assume we have already ordered $y_1, \ldots, y_t$. We compare y_{t+1} first with $y_{\{t/2\}}$, secondly with $y_{\{t/4\}}$ or $y_{\{3t/4\}}$ depending on the answer of the first test, and so on . . . . The number of tests is maximally

$$l \leqslant \{\log 2\} + \{\log 3\} + \cdots + \{\log(n-1)\} < \log((n-1)!) + n - 3. \tag{3.14}$$

Steinhaus conjectured in [1950] this procedure to be optimal, however, in [1958] he disproved the conjecture. Asymptotically, the lower (3.13) and the upper (3.14) bounds are equivalent, but we do not know the best algorithm up to now. Ford and Johnson [1959] determined an algorithm better than Steinhaus' one. (See also Wells [1965], and Cesari [1968].)

A generalization of the above problem is to find and order the t largest y's. This generalization does not belong to the general search problem treated here. But we can generalize it toward this direction: *The n objects* $x_1, \ldots, x_n$ *are divided into disjoint classes. We wish to determine just the class to which the unknown* x_i *belongs.*

In our case: $x_1, \ldots, x_n$ are the permutations of $y_1, \ldots, y_n$. The classes consist of the permutations where the last t elements are fixed. The number of classes is $n(n-1) \cdots (n-t+1)$.

If $t = 1$, it is easy to see that

$$l = n - 1.$$

The case $t = 2$ has been solved by Schreier [1932], Slupecki [1949–51] and Sobel [1968a]. The case of general t is obviously unsolved. For estimations see Hadian and Sobel [1970]. A further considered but unsolved problem is to determine the minimax of binary comparisons sufficient to identify the tth largest element from $y_1, \ldots, y_n$. Kislicyn [1964], Hadian and Sobel [1969], and Hadian [1969] worked out algorithms.

R. C. Bose and Nelson [1961] modified Steinhaus' problem: We wish to determine the natural order of the given (pairwise different) numbers $y_1, \ldots, y_n$ by binary changes instead of binary comparisons. That is, if $y_i < y_j$ $(i \neq j)$ there is no change, if $y_i > y_j$, we use the order $y_1, \ldots, y_{i-1}$, y_j, $y_{i+1}, \ldots, y_{j-1}, y_i, y_{j+1}, \ldots, y_n$. What is the minimum of the maximal number of steps needed to determine the natural order?

The minimum is not known, but a good algorithm is given by R. C. Bose and Nelson [1961]. About the ordering problems see also David [1959] and Moon [1968].

3.6. *Case* (Aα), (Bβ), (Cβ), (Dα), (Eα) = (Eβ)

We have to determine the minimal m for which there exist subsets $A_1, \ldots, A_m$ of $X = \{x_1, \ldots, x_n\}$ constituting a strategy and satisfying $|A_i| \leqslant k$ ($k < \frac{1}{2}n$). If the subsets of a strategy do not depend on the previous answers, then they form simply a separating system (see the Introduction). It is proved by Katona [1966] that this minimal m is equal to the minimal m such that there exist non-negative integers $s_0, \ldots, s_m$ satisfying

$$\begin{aligned} mk &= \sum_{j=0}^{m} j s_j, \\ n &= \sum_{j=0}^{m} s_j, \\ s_j &\leqslant \binom{m}{j} \qquad (0 \leqslant j \leqslant m). \end{aligned} \tag{3.15}$$

By this fact, the next theorem was proved.

Theorem 3.9. *Suppose that* $A_1, \ldots, A_m \subset X = \{x_1, \ldots, x_n\}$ *satisfy the condition* $|A_j| \leqslant k$ $(1 \leqslant j \leqslant m)$ (k *is given* $< \frac{1}{2}n$) *and constitute a separating system. Under this condition, for the minimum of* m *the inequalities*

$$\frac{\log n}{\log (en/k)} \frac{n}{k} \leqslant \min m \leqslant \left\{ \frac{\log 2n}{\log (n/k)} \right\} \frac{n}{k}$$

hold.

Dickson [1969] introduced the concept of the completely separating system. (It does not have, probably, a nice interpretation in search theory, but it is interesting in itself): $A_1, \ldots, A_m$ is a *completely separating system* if for any pair x_i, x_j $(i \neq j)$ there is a k such that $x_i \in A_k$, $x_j \notin A_k$.

What is the minimum of m such that there exists a completely separating system $A_1, \ldots, A_m$ for $\{x_1, \ldots, x_n\}$? This is solved asymptotically by Dickson [1969] and Spencer [1970] proved

Theorem 3.10. *The minimal* m *for which a completely separating system* $A_1, \ldots, A_m$ *exists is*

$$\min \left\{ m : \binom{m}{[\frac{1}{2}m]} \geqslant n \right\}.$$

Two subsets A_1 and A_2 of X are said to be *qualitative independent* if none of the sets A_1, A_2, $A_1\bar{A}_2$, $\bar{A}_1A_2$, $\bar{A}_1\bar{A}_2$ is empty, where $\bar{A}$ denotes the complement $X-A$. In other words, testing first by A_1, we obtain some information by testing A_2, independently of the answer of the first test. For instance if $A_1A_2 = \emptyset$ then after the answer $x_i \in A_1$ the test A_2 does not give any information. Rényi [1971] asked what is the maximum of the pairwise qualitative independent sets. He solved the problem for even n in the following way: If $A_1, \ldots, A_m$ are qualitative independent, then it is easy to see that $A_1, \bar{A}_1, \ldots,$

A_m, $\bar{A}_m$ form such a system that none of them is contained in another one. By Sperner's theorem [1928] we obtain

$$2m \leqslant \binom{n}{[\frac{1}{2}n]} \quad \text{and} \quad m \leqslant \tfrac{1}{2}\binom{n}{\frac{1}{2}n}.$$

This estimation is the best possible because we can choose $\frac{1}{2}\binom{n}{\frac{1}{2}n}$ pairwise disjoint subsets of $\frac{1}{2}n$ elements. For odd n this estimation is not the best possible. The right value is

$$\binom{n-1}{\frac{1}{2}(n-1)}$$

(see Rényi [1971]). The maximal number of r-wise qualitative independent sets is not yet determined. An estimation is given in Rényi's book [1971].

3.7. *Case* (Aβ), (Bα), (Cα), (Dα), (Eα)

Each of the elements $x_1, \ldots, x_n$ can be defective independently with probability p. We can not use the results of encoding type for this model, but it can be done for a transformed variant: Let $x'_1, \ldots, x'_{2^n}$ be the subsets of $X = \{x_1, \ldots, x_n\}$. Exactly one x'_i of the elements of $X' = \{x'_1, \ldots, x'_{2^n}\}$ is "defective" (it is the subset of all defective elements). However, the testing subsets $A \subset X$ are also transformed. A has a defective element if and only if $A = x'_j$ has a common element with the set x'_i of defective elements. It is equivalent to $x'_i \in A'$ where A' is the set of x_k's non-disjoint to x'_j. However, since such subsets A' are very special, we reduced our problem to a problem of type (Aα), (Bα), (Cβ), (Dα), (Eα).

The restrictions for the testing subsets are very particular. We can not solve the problem, but an easy lower bound for the average number of tests follows from (2.6):

$$L_{\min}(P) \geqslant H(P),$$

where $P = (p^n, p^{n-1}q, p^{n-1}q, \ldots, q^n)$. It is well known (see e.g. Feinstein [1958]) that $H(P) = n(-p \log p - (1-p) \log (1-p))$ holds in this case. We obtain for the average test number

$$L \geqslant n(p \log p - (1-p) \log (1-p)). \tag{3.16}$$

However, it is not the best possible lower bound. For example, Ungar [1960] proved the following

Theorem 3.11. *If* $p \geqslant \frac{1}{3}(3-\sqrt{5})$ *then*

$$L \geqslant n$$

and for $0 \leqslant p < \frac{1}{2}(3-\sqrt{5})$ *there is a strategy with*

$$L < n.$$

For large P, $L \geqslant n$ is obviously a better estimate than (3.16), and it is exact in this case, since for the strategy "element by element" $L = n$. In this

case the combinatorial search fails. However, Ungar's theorem ensures that for small P it is a good method. Sobel [1960] and Sobel and Groll [1959] worked out good strategies for searching. These procedures give upper estimates for the optimal average test number $L_{\min}(n)$. For example, Sobel [1960] (partly personal communication) has proved

$$\lim_{n\to\infty} \frac{L_{\min}(n)}{n} \leqslant -p \log p - q \log q + \frac{p}{1-q^x},$$

where x is the smallest integer such that $1-q^x-q^{x+1} \geqslant 0$. The right hand side is $\leqslant p \log p - q \log q + 1$, or, if p is small, then it is $\leqslant -p \log p - q \log q + 2p$.

The next problem does not belong formally to this section, but it is a very closed generalization of the problem treated here. We have three types of elements in X: good, mediocre and defective ones. Testing a subset A of X it shows the "minimum" of its elements: The test says "good" if all the elements of A are good; it says "mediocre" if there is at least one mediocre element in A, but none of them is defective; it says "defective" if there is at least one defective element in A. The elements of X are good, mediocre and defective independently with probabilities q_1, q_2 and q_3 ($q_1+q_2+q_3 = 1$), respectively. Kumar [1970] has a result analogous to Ungar's theorem: If $q_1 \geqslant \frac{1}{2}(q_2-1+(5q_2^2-6q_2+5)^{\frac{1}{2}})$ then $L \geqslant n$; that is, the test "element by element" is the best possible. On the other hand, if $q_1 < \frac{1}{2}(q_2-1+(5q_2^2-6q_2+5)^{\frac{1}{2}})$ then there is a better strategy satisfying $L < n$. Similarly, Kumar [1970] gives a good strategy, which is a generalization of Sobel [1960] and Sobel-Groll [1959].

3.8. *Cases* (Aα), (Bα), (Cα), (Dα), (Eβ) *and* (Aβ), (Bβ), (Cα), (Dα), (Eα) = (Eβ)

These cases are uninteresting, because for $p = \frac{1}{2}$, (3.16) gives $L \geqslant n$, and this is a lower bound for these cases. The strategy "element by element" is the best one.

3.9. *Case* (Aβ), . . ., (Cβ), (Dα), . . .

These problems are not considered in the literature. Sobel [1960] is the only author that points out that his strategy is *alphabetical* in the sense that the testing subsets are "intervals" in the ordered set $\{x_1, \ldots, x_n\}$.

3.10. *Case* (Aγ)

We did not introduce this case in Section 1. The common generalization of the cases (Aα) and (Aβ) is the case when the probabilities $p(A)$ of A ($\subset X$) being the set of defective elements are given for all A.

In this generality the problem is too hard to solve. A very particular case is when $p(A) = 1/\binom{n}{2}$ for $|A| = 2$ and $p(A) = 0$ otherwise. (Assume (Bα), (Cα), (Dα), (Eα)). It is easy to transform it into a problem of type (Aα), (Bα),

(Cβ), (Dα), (Eα) considering the set of unordered pairs (x_i, x_j) $(i \neq j)$. For this modified problem we may apply Theorem 3. More exactly, the formula of Theorem 3 gives a lower bound for the minimum of the average test number. Moreover, Sobel (1968b) proved that we can reach this lower bound for infinitely many n's.

Theorem 3.12. *There are exactly two defective elements in the set* $\{x_1, \ldots, x_n\}$, *all possibilities with equal probabilities. For the optimal strategy the average test length is denoted by* $L_2(n)$. *Then*

$$L_2(n) = \{\log \binom{n}{2}\} - (2^{\{\log \binom{n}{2}\}} / \binom{n}{2} - 1),$$

if

$$n = 2^{\frac{1}{2}(m+1)} + [\tfrac{1}{3}(2^{\frac{1}{2}(m-1)} - 4)] \text{ for some odd } m \geqslant 1,$$

or

$$n = 2^{\frac{1}{2}m} + [\tfrac{1}{3}(2^{\frac{1}{2}(m+2)} - 4)] \text{ for some even } m \geqslant 0.$$

For the remaining n's there is a small difference between the lower bound and the average of the strategy worked out by Sobel [1968].

Sobel and Groll [1966] investigated the problem (Aβ) in the case when we do not know the exact value of p and we use an *a priori* distribution by the test as well as tests to get a Bayes solution of the problem. This problem is more statistical than combinatorial.

3.11. *Case* (Dβ)

In this case at each test we divide X into disjoint subsets and the result of the test shows us which one (or which ones) includes the unknown element(s). If the number of disjoint subsets is at each test a constant (say r), then many problems can be solved (and they are) in the same way as for (Dα). We do not want to repeat them.

It may occur that the number r of subsets depends on the situation, that is, on the previous tests and previous results. Picard [1965] generalized Theorem 2.1 (Huffman algorithm) toward this case.

There is one classical problem which belongs typically to this case: the so called "defective coin problem". The basic situation is the following: *We have n coins, and one of them is defective, with probability* 1. *The good coins weigh* 1 *and the defective one weighs* $1+\varepsilon$ $(0 < \varepsilon < 1)$. *We wish to find the defective coin using an equal arm balance.* Let X be the set of coins. Taking a subset A and a subset B $(A \cap B = \emptyset)$ on the right and left hand side of the balance, respectively, we may obtain three results: balance and unbalance in two ways. In the first case we know that the defective coin x_i is in $X-A-B$ and in the second case we know which one of A and B includes x_i. One test divides X into three parts, and says which one includes x_i. However, there is a restriction: $|A| = |B|$. If we try to weigh subsets with different cardinalities,

no information is obtained. (The problem belongs to case (Aα), (Bα), (Cβ), (Dβ), (Eα) or (Eβ).) Let us generalize our Theorem 3.1:

$$L^3_{\min}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) = \{\log_3 n\} - [\tfrac{1}{2} \cdot 3^{\{\log_3 n\}} - \tfrac{1}{2}n]/n. \tag{3.17}$$

It gives a lower bound for the average test number. This lower bound is attainable even under the condition $|A| = |B|$ (see Cairns [1963] and Baranyai (a)) except for $n = 6$, when the minimum of the average is 2. On the other hand, the last term in (3.17) is less than 1 (if $\{\log_3 n\} = \log_3 n$ then 0), thus $\{L^3_{\min}(1/n, \ldots, 1/n)\} = \{\log_3 n\}$ gives a lower bound for the maximal number of tests sufficient to identify the defective coin.

A different problem is if in a test we can use only the elements of the subset containing the defective one according to the last test (that is we cannot weigh the coins proved to be good). Equation (3.17) is again a lower bound.

However, because of the restrictions we can not reach (3.17) for every n (for we can not reach $3^{\{\log_3 n\}} - n \equiv 3 \pmod 4$). Cairns [1963] (for a new simpler proof see Baranyai (a)) determined the optimal strategy which is optimal for both cases (Eα) and (Eβ):

Theorem 3.13. *The optimal strategy is the following: If we know that the defective coin is an element of an n' element subset, then let us weigh m coins against m other ones from this subset, where m is the odd one of the numbers $[\frac{1}{3}(n'+1)]$ and $[\frac{1}{3}(n'+4)]$. The maximum test number is $\{\log_3 n\}$ for this strategy, and the average test number is also optimal.*†

The case when there are more (but fixed number h) defective coins is more complicated, if we assume that we are not able to determine the number of defectives in a subset by one test because the weights w_i of the defectives are different (but $1 < w_i < 1+1/h$). For particular results see Cairns ($h = 2$) [1963], Bellmann and Gluss [1961] and Smith [1947].

A different problem is proposed by Shapiro [1960] and Fine [1960]. Again, we have n coins, some of them being defective. The weights of good and defective coins are a and b, respectively. We may use for tests scales (not equal arm balance). Thus, by one test we are able to determine the number of defective coins in the tested subset. Determine $l(n)$, the minimum of the maximal test number needed to determine all the defectives. Many authors (Cantor [1964], Shapiro and Söderberg [1963], Erdős and Rényi [1963]) have asymptotical results for $l(n)$. Finally, Lindström [1964, 1965, 1966] proved

$$\lim_{n\to\infty} \frac{l(n) \log n}{n} = 2.$$

† Baranyai noticed that this is not true if $n' = 3^a+2$, and the right $m = 3^{a-1}$.

4. Random search

Let us go back to the simplest case: exactly one of the elements $x_1, \ldots, x_n$ is defective, x_i is defective with probability p_i. Again, subsets are used to test (any subset). Rényi proposed to choose the subsets randomly, with probability $1/2^n$. Is the number of tests much larger than in the traditional case? The answer is definitely "no" (Rényi [1962a, 1961a]):

Theorem 4.1. *If the subsets $A_1, \ldots, A_m$ of $X = \{x_1, \ldots, x_n\}$ are chosen independently with probability $1/2^n$, and $P(n, m)$ denotes the probability of the event that $A_1, \ldots, A_m$ constitute a separating system then*

$$\lim_{n\to\infty} P(n, 2\log n + c) = \begin{cases} 1 & \text{if } c = \infty \\ e^{-1/2^{c+1}} & \text{if } c \text{ is finite} \\ 0 & \text{if } c = -\infty. \end{cases}$$

It means that if we choose e.g. $m = 2\log n + 6$, then for sufficiently large n the system $A_1, \ldots, A_m$ is a separating system with probability $e^{-1/2^7} \sim 0.99$. Comparing with Theorem 1, choosing randomly the subsets, we have to test roughly twice as many as the minimal number of systematically selected subsets which determine uniquely the unknown element. If the costs of the tests are small and the costs of working out a systematical plan are large, then it is better to use the random search.

However, we do not need a separating system to identify the unknown x_i. It is sufficient if $A_1, \ldots, A_m$ *separates* x_i from the other elements, that is, if they satisfy (1.3) for x_i and for an arbitrary x_k ($1 \leqslant k \leqslant m$, $i \neq k$). Denote by $S(n, m)$ the probability of the event that $A_1, \ldots, A_m$ separates x_i (it does not depend on i).

Theorem 4.2. (Rényi [1962b]). *If $c > 0$, then*

$$S(n, \log n + c) \geqslant e^{-1/2^c}.$$

It means that if we use seven more questions than at the systematical search, then we find the unknown element with probability $e^{-1/2^7} \sim 0.99$. This is very surprising.

The random choice of the subsets with probability means that we choose subsets with sizes about $n/2$. If the probability of choice of a subset $|A|$ is $p^{|A|}q^{n-|A|}$ then the chosen subsets have about p^n elements. A generalization (Rényi [1961b]) of Theorem 4.2 says that in this case we need about $m = \log n/[H(p, 1-p)]$ tests. (Compare with Theorem 3.9.)

Again, the next problem was proposed and solved by Rényi [1961b]: It may occur that our tests are not reliable. The result of a test is right and false with probabilities β and $1-\beta$, respectively (obviously these cases are independent from the results of the other tests). In this case we need about $\log n/[1-H(\beta, 1-\beta)]$ tests to identify the unknown element. (There are strong connections with Gallager's random coding [1968].)

For further generalizations see Rényi [1961b] and [1965].

Finally, we wish to mention a result of Rényi [1970] which appeared after his tragic death. A *q-regular strategy* is a strategy which divides into exactly q parts the subset which is known to include the unknown element (case (Aα), (Bα), (Cα), (Dα)). It is easy to see that in this case $n = 1 \pmod{q-1}$. Rényi determined the number $Cq(n)$ of different (they are not different if they differ only in the permutations of the elements $x_1, \ldots, x_n$) q-regular strategies:

$$C_q(n) = \frac{(kq)!}{k!n!} \quad \text{where} \quad n = k(q-1)+1.$$

Similarly, the total number $D(n)$ of different strategies for $X = \{x_1, \ldots, x_n\}$ is

$$D(n) = \frac{1}{n}\sum_{k=1}^{n-1}\binom{n-2}{k-1}\binom{n+k-1}{k} \sim \frac{\sqrt{3-2\sqrt{2}}}{4\sqrt{\pi}}\frac{(3+2\sqrt{2})^n}{n^{\frac{3}{2}}}$$

(see also Rényi [1969]). Recently, Chorneyko and Mohanty have a generalization of these results.

5. Open problems

Comparing the several sections and combining their conditions it is easy to obtain a large number of open problems. We want to emphasize some of them (it does not mean that they are the most important ones, they are the most interesting only to the author):

1. Generalize the Huffman algorithm for the case (Aγ). More exactly: Probabilities $p(A)$ are given for any $A \subset X = \{x_1, \ldots, x_n\}$ of the event that A is the set of defective elements.

A *general strategy* is a strategy which is able to determine all the defective elements. Find an algorithm which determines the general strategy with the minimal average number. (See Theorem 2.1, case (Aγ) and the beginning of (Aβ).)

2. Find the conditions under which it is possible to determine the optimal average length (see Theorems 3.1, 3.2, 3.3 and 3.4).

3. Generalize the results for alphabetical codes (see Theorems 3.7, 3.1, 3.2, 3.3 and 3.4).

4. Generalize the Huffman algorithm for the case if we can use only subsets with size $\leqslant k$ ($k < \frac{1}{2}n$) (see Theorems 3.8 and 3.9).

5. Determine the best strategy for Steinhaus' problem, or at least give a better lower bound (see (3.13)).

6. Determine the minimal number m for which there exists a separating system $A_1, \ldots, A_m$ satisfying $|A_i| \leqslant k$ ($i = 1, \ldots, m$, k fixed $< \frac{1}{2}n$). Theorem 3.9 gives good estimates for this minimum. Generalize (3.15) for the case (Dβ) when $A_1, \ldots, A_m$ are partitions into r parts and the sizes of the first $r-1$ parts are bounded.†

† Very recently it is solved by Zs. Baranyai (b). (Added in December, 1971.)

7. Determine the minimal number m for which there exists a completely separating system $A_1, \ldots, A_m$ satisfying $|A_i| \leqslant k$ $(i = 1, \ldots, m,\ k$ fixed $< \frac{1}{2}n)$. (See Theorem 3.10.)

8. Determine the minimal number m for which there exists a system $A_1, \ldots, A_m$ $(\subset X)$ such that for any x_i, x_j $(i \neq j)$ there are disjoint A_k and A_l $(A_k \cap A_l = \emptyset)$ with $x_i \in A_k$, $x_j \in A_l$.

9. Determine the maximal m for which there are subsets $A_1, \ldots, A_m$ such that any r different of them are qualitative independent (none of the sets of type $A_{i_1}\bar{A}_{i_2} \cdots A_{i_r}$ are empty). (See the end of section 3.6.)

10. Find a better estimate than (3.16) (see also problem 1).

11. Generalize Theorem 3.13 for a "*three-arm balance*" which has three equally sized arms (with angles $\frac{2}{3}\pi$), and which is balanced only if three equal weights are weighed.

12. $X = \{x_1, \ldots, x_n\}$. There is exactly one defective element. It is x_i with probability p_i. We can test any subset $A \subset X$ whether $x_i \in A$ or not. The next test may depend on the results of the previous tests (case (Aα),(Bα),(Cα),(Dα)). However, the tests are noisy, that is, we received the contrary results with probability q. Find an algorithm which determines the strategy which has the minimal average length, but discovers the defective element with given probability $1-\varepsilon$. (In the language of codes: variable length (not black) code with minimal code length with error probability ε.)

13. There are exactly one defective element and one mediocre element in the set X, with probabilities $p_1, \ldots, p_n$ and $q_1, \ldots, q_n$. Which strategy minimize the maximal number of tests needed to identify both elements (see the end of section 3.1 and Theorem 3.12).

References

Zs. Baranyai, (a), to be published later.

Zs. Baranyai, (b), to be published later.

R. Bellman and B. Gluss, 1961, On various versions of the defective coin problem, *Inf. Control* **4**, 118–131.

R. C. Bose and R. I. Nelson, 1961, A sorting problem, Case Inst. Technol., Computing Center, Rept. No. 1043, pp. 1–22.

S. S. Cairns, 1963, Balance scale sorting, *Am. Math. Monthly* **70**, 136–148.

L. L. Campbell, 1968, Note on the connection between search theory and coding theory, *Proc. Colloq. on Information Theory* (A. Rényi, ed.; János Bolyai Math. Soc., Budapest).

D. G. Cantor, 1964, Determining a set from the cardinalities of its intersections with other sets, *Canad. J. Math.* **16**, 94–97.

Y. Cesari, 1968, Questionnaire, codage et tris, Institute Blaise Pascal, Paris.

Y. Cesari, 1970, Optimisation des questionnaires avec contrainte de rang.

I. Z. Chorneyko and S. G. Mohanti, On the enumeration of pseudo-search codes, submitted to *Studia Sci. Math. Hungar.*

H. A. David, 1959, *The Method of Paired Comparisons* (Hafner Publ. Co., New York).

T. I. Dickson, 1969, On a problem concerning separating systems of a finite set, *J. Combin. Theory* **7**, 191–196.

R. Dorfman, 1943, The detection of defective members of large populations, *Ann. Math. Statist.* **14**, 436–440.

P. Erdös and A. Rényi, 1963, On two problems of information theory, *Publ. Math. Inst. Hungar. Acad. Sci.* **8**, 241–254.

A. Feinstein, 1958, *Foundations of Information Theory* (McGraw-Hill, New York).

N. J. Fine, 1960, Solution EI 399, *Am. Math. Monthly* **67**, 697.

L. R. Ford and S. M. Johnson, 1959, A tournament problem, *Am. Math. Monthly* **66**, 387–389.

R. G. Gallager, 1968, *Information Theory and Reliable Communication* (Wiley, New York).

E. N. Gilbert and E. F. Moore, 1959, Variable-length binary encodings, *Bell Syst. Tech. J.* **38**, 933–967.

A. Hadrian, 1969, Optimality properties of various procedures for ranking n different numbers using only binary comparisons, Tech. Rept. No. 117, Dept. of Statistics, Univ. of Minnesota.

A. Hadrian and M. Sobel, 1969, Selecting the tth largest of n items using binary comparisons, Tech. Rept. No. 121, Dept. of Statistics, Univ. of Minnesota.

A. Hadrian and M. Sobel, 1970, Ordering the t largest items using binary comparisons, *Combinatorial Math. and its Appl.*, Univ. of North Carolina, Chapel Hill, N.C.

T. C. Hu and A. C. Tucker, 1970, Optimum binary search trees, *Combinatorial Mathematics and its Applications* (Univ. of North Carolina, Chapel Hill, N. Car.).

D. A. Huffman, 1952, A method for the construction of minimum redundancy codes, *Proc. I.R.E.* **40**, 1098.

G. Katona, 1966, On separating systems of a finite set, *J. Combin. Theory* **1**, 174–194.

G. Katona and M. A. Lee, Some remarks on the construction of optimal codes, submitted to *Acta Math. Acad. Sci. Hungar.*

S. S. Kislicyn, 1964, On the selection of the kth element of an ordered set of pairwise comparison, *Sibirsk. Mat. Zh.* **5**, 557–564 (in Russian).

D. E. Knuth, 1971, Optimum binary search trees, *Acta Inform.* **1**, 14–25.

S. Kumar, 1970, Group-testing to classify all units in a trinomial sample, *Studia Sci. Math. Hungar.* **5**, 229–247.

B. Lindström, 1964, On a combinatorial detection problem I, *Publ. Math. Inst. Hungar. Acad. Sci.* **9**, 195–206.

B. Lindström, 1965, On a combinatorial problem in number theory, *Canad. Math. Bull.* **8**, 477–490.

B. Lindström, 1966, On a combinatorial detection problem II, *Studia Sci. Math. Hungar.* **1**, 353–361.

J. W. Moon, 1968, *Topics on Tournaments* (Holt, Rinehart and Winston, New York), p. 48.

C. Picard, 1965, *Théorie des Questionnaires* (Gauthier-Villars, Paris).

A. Rényi, 1961a, On random generating elements of a finite boolean algebra, *Acta. Sci. Math. (Szeged)* **22**, 75–81.

A. Rényi, 1961b, On a problem of information theory, *Publ. Math. Inst. Hungar. Acad. Sci.* **6**, 505–516.

A. Rényi, 1962a, Statistical laws of accumulation of information, *Bull. Inst. Intern. Statist.* **39**(2), 311–316.

A. Rényi, 1962b, Az információ-akkumuláció statisztikus törvényszerüségéröl, *Magyar Tud. Akad. III Oszt. Közl.* **12**, 15–33.

A. Rényi, 1965, On the theory of random search, *Bull. Am. Math. Soc.* **71**, 809–828.

A. Rényi, 1969, *Lectures on the Theory of Search*, Mimeo Series No. 600.7, Dept. of Statistics, Univ. of North Carolina, Chapel Hill, N. Car.

A. Rényi, 1970, On the enumeration of search-codes, *Acta Math. Acad. Sci. Hungar.* **21,** 27–33.

A. Rényi, 1971, *Foundations of Probability* (Holden-Day, San Francisco).

M. Sandelius, 1961, On an optimal search procedure, *Am. Math. Monthly* **68,** 138–154.

J. Schreier, 1932, On a tournament elimination system, *Mathesis Polska,* 7 154–160 (in Polish).

H. S. Shapiro, 1960, Problem E 1399, *Am. Math. Monthly* **67,** 82.

J. Slupecki, 1949–51, On the system S of tournaments, *Colloq. Math.* **2,** 286–290.

C. A. B. Smith, 1947, The counterfeit coin problem, *Math. Gaz.* **31,** 31–39.

M. Sobel, 1960, Group testing to classify efficiently all defectives in a binomial sample, *Information and Decision Processes* (R. E. Machol, ed.; McGraw-Hill, New York), pp. 127–161.

M. Sobel, 1967, Optimal group testing, *Proc. Colloq. on Information Theory,* Bolyai Math. Society, Debrecen, Hungary.

M. Sobel, 1968a, On the ordering of the t best of n items using binary comparisons, Tech. Rept. No. 113, Dept. of Statistics, Univ. of Minnesota (submitted for publication).

M. Sobel, 1968a, Binomial and hypergeometric group-testing, *Studia Sci. Math. Hungar.* **3,** 19–42.

M. Sobel, 1970, A characterization of binary codes that correspond to a class of group-testing procedures, Tech. Rept. No. 148, Dept. of Statistics, Univ. of Minnesota.

M. Sobel and P. A. Groll, 1959, Group testing to eliminate efficiently all defectives in a binomial sample, *Bell System Tech. J.* **38,** 1179–1252.

M. Sobel and P. A. Groll, 1966, Binomial group-testing with an unknown proportion of defectives, *Technometrics* **8,** 631–656.

M. Sobel and S. Kumar, 1971a, Finding a single defective in binomial group-testing, *J. Am. Statist. Assoc.* (accepted for publication).

M. Sobel and S. Kumar, 1971b, Group-testing with at most c tests for finite c and c, Tech. Rept. No. 146, Dept. of Statistics, Univ. of Minnesota.

E. Sperner, 1928, Ein Satz über Untermengen einer endlichen Menge, *Math. Z.* **27,** 544–548.

J. Spencer, 1970, Minimal completely separating systems, *J. Combin. Theory* **8,** 446–447.

H. Steinhaus, 1950, *Mathematical Snapskost* (Oxford Univ. Press, New York).

H. Steinhaus, 1958, *One Hundred Problems in Elementary Mathematics* (Pergamon Press, London), Problems 52, 85.

A. Sterrett, 1957, On the detection of defective members of large populations, *Ann. Math. Statist.* **28,** 1033.

P. Ungar, 1960, The cut-off point for group testing, *Commun. Pure Appl. Math.* **13,** 49–54.

J. Vigassy, personal communication.

M. B. Wells, 1965, Application of a language for computing in combinatorics, IFIP Congress.

S. Zimmerman, 1959, An optimal search procedure, *Am. Math. Monthly* **66,** 690–693.