ELSEVIER

Theoretical Computer Science 270 (2002) 71–109

Theoretical Computer Science

www.elsevier.com/locate/tcs

Fundamental Study

Searching games with errors—fifty years of coping with liars

Andrzej Pelc [1]

Département d'Informatique, Université du Québec à Hull, C.P. 1250, Succ. B, Hull, Qué., Canada J8X 3X7

Received November 2000; accepted July 2001
Communicated by A.S. Fraenkel

Abstract

This is a survey on searching with errors, considered in the framework of two-person games. The *Responder* thinks of an object in the search space, and the *Questioner* has to find it by asking questions to which the Responder provides answers, some of which are erroneous. We give a taxonomy of such games, depending on the type of questions allowed, on the degree of interactivity between the players, and on the imposed limitations on errors. We survey the existing results concerning such games, concentrating on the issue of optimizing the Questioner's querying strategy, and pointing out open problems. We show the relations between searching games with errors and problems concerning communication through a noisy channel and error-correcting codes. Finally, we discuss other search and computation problems with faulty feedback which are related to searching with errors. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Searching; Error; Lie; Game; Strategy

Contents

[1] Supported in part by NSERC grant OGP 0008136.
E-mail address: pelc@uqah.uquebec.ca (A. Pelc).

PII: S0304-3975(01)00303-6

1. Codes, noisy channels, and the Rényi–Ulam game

One of the most important applications of fault-tolerance concerns two-party communication through a noisy channel. The two communicating agents, the *sender* and the *receiver* agree a priori on a set of M messages (binary sequences). The sender sends one of these messages through a noisy channel. During the transmission some of the bits of the message may get distorted, and the receiver (sometimes) gets a corrupted message. His task is to recover the message sent by the sender. This task is clearly impossible to achieve if the message can be distorted arbitrarily. However, in practice, only a small part of the message is corrupted due to the "noise" in the communication channel. Hence it is reasonable to assume some limitation on the possible distortion, such as an upper bound on the number of corrupted bits, or a probability that a given bit is changed. This yields the problem of finding a coding of the M messages, such that in spite of any distortion within the supposed limits, the receiver is always able to recover the original message. A classic limitation on the amount of distortion is assuming that at most e bits can be changed during the transmission but these errors can occur in arbitrary places.

A set C of M binary sequences of length n is called an *e-error-correcting code* of *size M* and *length n*, if, whenever the sender sends a sequence $s \in C$ through the channel, and at most e of its bits are changed, resulting in a sequence s', the receiver can always compute correctly the sequence s from the obtained sequence s'. Elements of C are called *words* of the code.

Moreover, the length n of a code should be as small as possible, because the length of a code increases the cost of its use. Hence the fundamental problem of the theory of error-correcting codes is

Given positive integers M and e, find the shortest e-error-correcting code of size M.

It is well known that the error-correcting capacity of a code depends on the minimum *Hamming distance* between its words, where the Hamming distance between two sequences is the number of positions where they differ. More precisely, a code is e-error-correcting if and only if this minimum distance is $2e + 1$. For such codes, the decoding strategy of the receiver is simply to compute the word of the code closest (in the sense of Hamming distance) to the received sequence s'.

In view of this characterization, the fundamental problem of coding theory has the following equivalent purely combinatorial, and more widely used formulation:

Given positive integers n and d, find the largest set C of binary sequences of length n with minimum Hamming distance d.

The above way of modeling noisy communication goes back to Shannon [110] and Hamming [45]. It assumes that the sender does not get any feedback from the receiver, and consequently the sender does not get any advantage from delaying the computation of consecutive bits of the message: all bits of the message may be computed a priori, before the first bit is sent. Berlekamp [17], Dobrushin [39] and Shannon [111] considered a different communication setting, where the noisy channel through which the sender sends messages to the receiver, has a companion *noiseless delayless feedback channel.* The receiver can send some feedback information to the sender through this feedback channel. It is assumed that the sender gets feedback on a given bit before sending the next bit, and that feedback is never distorted. Such asymmetry in the characteristics of the two channels can be explained if the transmission powers of the sender and the receiver are not equal. For example, if the sender is located in a space-ship and the receiver at the base on Earth, the transmitter of the sender is usually much weaker than that of the receiver, and consequently the likelihood of distortion of messages sent by the receiver may be negligible, while that of messages sent by the sender may be quite significant.

The existence of the noiseless delayless feedback channel permits to organize communication in an interactive way: the sender may send a bit, wait for the feedback, compute the next bit of the message on the basis of this feedback, and so on. Again, imposing some limitations on the amount of distortion caused by the noisy channel, it is important to find efficient (interactive) coding strategies which permit the receiver to recover the original message if the limits of distortion are not exceeded. Now the length (number of bits) of the messages sent by the sender may differ, depending on the feedback he/she receives. The *length* of a coding strategy is the worst-case number of bits used by the sender. In particular, assuming that at most e bits are distorted per message, we define an *adaptive e-error correcting code* as a strategy which permits the receiver to recover the original message, provided that at most e bits are corrupted. Similarly as before, we may now ask the fundamental question of what could be called *adaptive e-error-correcting code* theory:

> Given positive integers M and e, find the shortest adaptive e-error-correcting code of size M.

These two fault-tolerant communication problems have a close game-theoretic counterpart, first proposed (to the best of our knowledge) by Rényi [105]:

> Two players are playing the game, let us call them A and B. A thinks of something and B must guess it. B can ask questions which can be answered by 'yes' or 'no' and he must find out of what A had thought from the answers. (...) it is better to suppose that a given percentage of the answers are wrong (because A misunderstands the question or does not know certain facts).

A similar game was later proposed by Ulam [117]:

> Someone thinks of a number between one and one million (which is just less than 2^{20}). Another person is allowed to ask up to twenty questions, to each of which the first person is supposed to answer only yes or no. Obviously the number can be guessed by asking first: is the number in the first half-million? and then again reduce the reservoir of numbers in the next question by one-half, and so on. Finally the number is obtained in less than $\log_2 1\,000\,000$. Now suppose one were allowed to lie once or twice, then how many questions would one need to get the right answer?

Both above games are variations of binary search with lies (errors). In the first one a probability of erroneous answer is assumed, and in the second—the number of lies is upper bounded by 1 or 2. To see how these games are connected to noisy transmissions, consider a searching strategy in a search space of size M, with some limitation on errors in answers (e.g., given probability of error, or given upper bound on the number of errors). Suppose that this strategy guarantees that the searching player (called *Questioner*) always finds the unknown object chosen by the responding player (called *Responder*) in at most n queries, provided that the limitation on errors is respected. Identify the Questioner with the receiver and the Responder with the sender in Berlekamp's communication setting involving a noisy channel with noiseless delayless feedback. Let the unknown element of the search space be the message that the sender (Responder) wants to send. Suppose that the questions in a particular run of the game are asked (according to the strategy) through the noiseless feedback channel, while answers yes or no, corresponding to bits 1 or 0, are sent through the noisy channel. The properties of the strategy guarantee that the receiver identifies the unknown object (i.e., the message sent by the sender) after getting at most n bits from him/her. Hence a searching strategy always succeeding in at most n queries yields an adaptive code of length n with the same error-correcting capability. Conversely, an adaptive e-error-correcting code of size M and length n yields a searching strategy that guarantees finding an unknown object in a search space of size M, using at most n questions, provided that the number of erroneous answers does not exceed e. An analogous observation is true for other limitations on errors, e.g., random ones.

It should be noted that it is not necessary to send (through the feedback channel) the actual questions determined by the searching strategy. Since both the sender and the receiver know the searching strategy, it is enough for the receiver to send back in each round the received (possibly corrupted) bit. Since the feedback channel is noiseless, the sender will learn what answer to the previous question the receiver got, and hence he/she will be able to compute what question the receiver *should ask* at this point, according to the strategy. The sender's next bit is the answer to this question, and so on.

Note that in the particular case when the limitation on errors is a fixed upper bound e on the number of corrupted bits, Berlekamp's model involving a noisy channel with noiseless feedback is equivalent to a weaker model in which the feedback channel is also noisy. Indeed, any strategy for the stronger model (allowing up to e corrupted bits one way, and assuming that all feedback transmissions are perfect) works also in the weaker model allowing a total of up to e corrupted bits in transmissions in both directions. However, for other types of limitations on errors, such as imposing a given probability of error for every transmitted bit, this equivalence does not hold anymore.

We have seen that the fundamental problem concerning adaptive e-error-correcting codes is equivalent to finding optimal searching strategies in the (adaptive) searching game with at most e errors (lies) in answers. Similarly, the fundamental problem of the theory of (nonadaptive) e-error-correcting codes is equivalent to finding optimal strategies in the nonadaptive version of this game, where all questions should be asked in one batch, without waiting for answers, then all answers collected, and the unknown object identified.

In the remainder of this paper we will study the game-theoretic formulation of the above problems, the described equivalences with fault-tolerant communication serving as one of the motivations for this study. Searching with errors has other practical applications, the following of which were among those given by Rényi [105].

- The search space is the set of components of a complex mechanism, one of which is faulty. Questions are tests of subsets of the components, revealing if the faulty one is among them. However, due to imperfections of the tests setup or of the measuring instruments, some test results are erroneous.

- The search space is the set of illnesses that a patient may have. In order to make a diagnosis, the doctor examines the patient in different ways, each examination or test answering the question if the patient's illness is in a given subset of possibilities. Due to possible errors in laboratory tests or to ambiguity of symptoms, some examinations are misleading.

We conclude this introductory section with some historic remarks. The first one concerns the name of searching games with errors, which are the main subject of our survey. As said above, such a game seems to have been defined for the first time by Rényi in the paper [105] published in 1961. In the formulation from [105] errors were assumed random. In a later paper [106] published in 1976 in Hungarian (i.e., at the same time as Ulam's biography [117]) but translated to English only in 1984, Rényi reformulated and generalized the definition saying that "the one who answers is allowed to lie a certain number of times". As mentioned above, Ulam's formulation in [117] referred to lying "once or twice". Thus it seems obvious that searching games with errors were introduced independently by Rényi and Ulam, Rényi being first, chronologically and alphabetically. However, in many papers published in the late eighties and in the nineties, e.g., [32, 44, 61, 66, 71, 73, 84, 85, 91] the game was called "Ulam's game" and the related problem "Ulam's problem". In most of the literature on searching games with errors, Rényi's papers [105, 106] are not even cited: the authors were apparently unaware of their existence and cited only [117] as the source of the game-theoretic formulation. The author of this survey must regretfully admit his ignorance of Rényi's contribution in this matter at the time of writing [29–32, 89–92, 94–97]. The first to acknowledge Rényi's definition were Cicalese and Vaccaro [26] and Cicalese [19]. They quoted [106] but not [105], and proposed the name "Rényi–Ulam game". In later papers [22–25, 27] the phrase "Ulam–Rényi game" was used. We adopt the term "Rényi–Ulam games", originally proposed in [26, 19], to denote all searching games with errors, regardless of constraints on lies and other variations. We feel that this term reflects both the apparent independence of both formulations and Rényi's chronological and alphabetic precedence.

The second issue concerns the terms "errors" and "lies". In [105] Rényi seems to hint at the first, citing practical examples where distortions are unintentional. In [106] he uses the term "lies" similarly as Ulam in [117]. In most of the literature these terms are used as synonyms, and we will follow this tradition. Nevertheless we note that, since the word "error" suggests lack of intention and the word "lie" its presence, in practical applications the first term is usually more appropriate. It is hard to give a practical example of the game where lies are intentional, apart from war-time situations when an enemy tries to maliciously interfere with communication.

The final historic remark concerns the second part of the title of the present survey: we announce "fifty years of coping with liars". This is justified by the fact that the first papers of Shannon [110] and Hamming [45] on error-correcting codes appeared in 1948 and 1950, respectively. Von Neumann's [118] pioneering research on relations between probabilistic logic and systems reliability should also be quoted in this context. Most of the papers on related issues, published in the fifties, sixties and seventies, deal with

e-error-correcting codes (see the large bibliography in [67]). We will not survey this part of the literature, first because there are excellent and comprehensive books on this field, such as [67], and second because, from the game-theoretic point of view, error-correcting codes are of only marginal interest. While formally they are equivalent to a nonadaptive version of Rényi–Ulam game with at most e errors, this version can be hardly even called a game as it has only three moves: the Questioner asks all questions, the Responder gives all answers, and the Questioner finds the unknown number. As for the adaptive version of the game, apart from the early papers [17, 105, 106], and Ulam's autobiography [117], most of the literature on this subject appeared in the eighties and nineties, and mostly these papers will be discussed in the present survey.

2. The taxonomy of Rényi–Ulam games

The general definition of a Rényi–Ulam game is that of a game between two players, the Questioner and the Responder, in which, before starting to play, both players agree on:
1. a set of objects, called the *search space*, and the number n of questions,
2. some limitation on the way the Responder is allowed to lie,
3. the format of questions,
4. the degree of interactivity between players.

The Responder thinks of an object in the search space, unknown to the Questioner. Then the Questioner asks questions and the Responder gives answers, according to the agreed rules. The Questioner wins the game if he/she identifies the unknown object, after n questions and answers. Otherwise the Responder wins. The Questioner has a *winning strategy* of length n if he/she can always win the game in at most n questions, regardless of the Responder's behavior (as long as it does not violate the agreed rules).

Since the game depends only on the size of the search space, and not on the nature of its elements, we will always denote this size by M, and assume that the search space is the set $\{0,\dots,M-1\}$.

Various ways of specifying items 2–4 in the above definition provide numerous possible sets of rules giving descriptions of particular Rényi–Ulam games. Many combinations of these rules have been proposed and investigated in the literature, and will be surveyed in the sequel. Below we describe the possible variations of each of these rules. The respective papers using a particular set of rules will be cited later, when the results under various scenarios are discussed.

The first rule to be fixed is the type of limitation on the way the Responder may lie. Clearly, some such limitation has to be imposed because answers in which the Responder may lie arbitrarily do not provide the Questioner with any information and are therefore useless. The limitations used in the literature are the following:

- a fixed upper bound on the number of lies: the Responder can lie at most e times during the entire game, where e is a fixed positive integer.

- an upper bound on the fraction of lies: if the entire game lasts n questions, the Responder can lie at most pn times, where $p<1$ is fixed.
- an upper bound on the fraction of lies in any initial segment: in any initial segment of first m answers the Responder can lie at most pm times, where $p<1$ is fixed.
- an upper bound on the number of lies in any segment of given fixed length: in any segment of m consecutive answers, where $m>1$ is fixed, the Responder can lie at most e times, where e is a fixed positive integer.
- random lies: before each answer the Responder tosses a coin with heads probability $p<1/2$ and lies, if and only if the result is heads.
- arbitrary sets of lie patterns: in a game of n questions, a lie pattern is a binary sequence of length n. The Responder lies according to a lie pattern s if he/she lies in answer i, if and only if $s(i)=1$. Before the game, both players agree on a set S of lie patterns, then the Responder chooses a lie pattern $s\in S$, unknown to the Questioner, and lies according to this pattern.
- half-lies: the number of lies is at most e during the entire game but only answers "no" can be erroneous, all answers "yes" are guaranteed to be correct.

The second rule to be specified concerns the format of admissible questions. The basic format is arbitrary yes–no questions, i.e., questions of the type $x\in A$?, where A is an arbitrary subset of the search space. This format was originally proposed by Rényi and Ulam. However, this was both generalized and restricted in various papers:

- arbitrary q-ary questions, for fixed $q>1$: the questions are of the form "To which of the sets $A_1,\dots,A_q$ the unknown object belongs?", where $A_1,\dots,A_q$ is any partition of the search space. For $q=2$ these are arbitrary yes–no questions.
- comparison questions: questions of the type "$x<a$?", where a is any element of the search space $\{1,\dots,M\}$.
- interval (resp. bi-interval) questions: questions of the form "Is x in the interval $[a,b]$?" (resp. "Is x in the set $[a,b]\cup[c,d]$?"), where $a<b$ and $c<d$.
- questions of restricted size: questions of the type "$x\in A$?", where the size of A cannot exceed a given upper bound k.
- prefix questions: questions of the type "Does the binary representation of x have prefix s?", where s is a binary sequence.
- variable-cost questions: the Questioner is charged depending on the question, and has a restricted budget.
- nonrepetitive questions: arbitrary yes–no questions which cannot be repeated.

The last thing the players have to agree upon before starting to play is the amount of interactivity between them.

- fully adaptive game: the Questioner learns the answer to each question before asking the next question.
- k-batch game: the Questioner asks questions in k batches (series), where k is a fixed positive integer. After each batch he/she gets the answers to all questions in the batch and then prepares the next batch of questions.
- nonadaptive game: this is a 1-batch game.

- delays and time-outs: the players agree on two integers $c, d \geqslant 0$. Up to c answers may be lost (the Responder may not answer up to c questions), and the Questioner has to ask one question at each time unit but receives an answer to a question d time units after asking it.

Many combinations of these rules were adopted by various authors (although not all 196 possibilities were considered). In the rest of this paper we will review the existing results concerning thus obtained variants of Rényi–Ulam games, and see how changes of rules influence differences in solutions of the main underlying problem, which is to find a shortest searching strategy for a given game.

3. Binary search with a fixed number of errors

In this section we consider the class of Rényi–Ulam games to which most of the papers in the literature of this domain were devoted: games using only yes–no queries (arbitrary or restricted) and imposing a fixed upper bound e on the number of allowed lies in all answers. The main tools for the analysis of these games, subsequently used by many authors, are due to Berlekamp [17]. Below we describe these tools and sketch the way they are used.

At any stage of the game the *state* of the game is a sequence $(x_0, x_1, \ldots, x_e)$ of integers, such that x_i, for $i = 0, 1, \ldots, e$ is the number of elements of the search space that falsify exactly i previous answers. The kth *volume* of a state $(x_0, x_1, \ldots, x_e)$ is the number

$$V_k(x_0, x_1, \ldots, x_e) = \sum_{i=0}^{e} x_i \sum_{j=0}^{e-i} \binom{k}{j}.$$

Every question "Does the unknown number belong to the subset A of the search space?" asked in a state $(x_0, x_1, \ldots, x_e)$ of the game can be coded as a sequence $(u_0, u_1, \ldots, u_e)$, where u_i is the number of elements of A that falsify exactly i previous answers. (Notice that, since the entire run of the game is invariant under permutations of the search space, we can describe states and questions in terms of sequences of integers, rather than sequences of subsets of the search space.) Consider a question $(u_0, u_1, \ldots, u_e)$ asked in the state $(x_0, x_1, \ldots, x_e)$. Let $v_i = x_i - u_i$. The answer "yes" (resp. "no") to this question yields state $(u_0, u_1 + v_0, \ldots, u_e + v_{e-1})$, called the yes-state (resp. state $(v_0, v_1 + u_0, \ldots, v_e + u_{e-1})$, called the no-state). Berlekamp [17] proved the following *volume conservation* law: if $\bar{y}$ and $\bar{z}$ are the yes-state and the no-state resulting from a state $\bar{x}$ after any question then $V_k(\bar{x}) = V_{k-1}(\bar{y}) + V_{k-1}(\bar{z})$. Since the initial state of the game is $(M, 0, \ldots, 0)$, and the Questioner wins the game exactly in states $(x_0, x_1, \ldots, x_e)$ for which $\sum_{i=0}^{e} x_i = 1$, the volume conservation law implies the following *volume bound* [17]: if the Questioner has a winning strategy of length n then $M \sum_{i=0}^{e} \binom{n}{i} \leqslant 2^n$. Notice that the above formula is the same as in the sphere-packing bound due to Hamming [45] but Berlekamp's result is a strengthening of the sphere-packing bound, as it gives the same lower bound on the length of e-error-correcting codes in the more general adaptive situation.

The volume conservation law indicates what should be the method to produce a shortest possible Questioner's strategy. In every state of the game the Questioner should ask a question resulting in splitting the volume of the state as evenly as possible. It is easy to check that if all terms of the state $(x_0, x_1, \ldots, x_e)$ are even then the question $(x_0/2, x_1/2, \ldots, x_e/2)$ splits the n-th volume of $(x_0, x_1, \ldots, x_e)$ into exact halves, regardless of n. The difficulty of designing good questioning strategies comes from the fact that eventually some terms in the resulting states are odd. (Even for $M = 2^r$, after r halving questions we get a state with the first term 1.) Since terms in the beginning of the sequence representing a state are "heavy" (each unit contributes a large amount to the volume), even a difference of one unit between respective terms in the yes-state and the no-state creates a large difference in volumes. This is the situation when a state with odd initial terms has to be split. The above mentioned difference of volumes usually has to be compensated by unbalanced splitting of "less significant" terms of the state (those with larger indices). Computing the best question in every state becomes therefore a difficult task, particularly for larger values of e. This task is even harder when additional restrictions are imposed on allowed questions.

3.1. Arbitrary questions

We first consider Rényi–Ulam games in which arbitrary yes–no questions are permitted, i.e., games in which the Questioner can ask "Does the unknown number belong to the subset A of the search space?", for any $A \subseteq \{0, \ldots, M-1\}$. Using the terms introduced above this means that in state $(x_0, x_1, \ldots, x_e)$ every question $(u_0, u_1, \ldots, u_e)$, where $u_i \leqslant x_i$, is allowed.

3.1.1. Fully adaptive search

As explained in Section 1, fully adaptive binary search with arbitrary questions and a fixed upper bound on the number of lies corresponds to communication through a noisy channel with noiseless delayless feedback, where we assume that at most e errors can be made during the entire transmission. This application, together with the fact that this was the precise setting proposed by Ulam [117], are probably the reasons why so many papers were devoted to this variation of Rényi–Ulam game.

We start with the exact results concerning the minimum length of the Questioner's strategy in the case when e is a small positive integer. The first such result was obtained by Pelc [91]. He proved that the minimum length of the Questioner's strategy, for $e = 1$, is $\min\{n\colon M(n+1) \leqslant 2^n\}$ if M is even, and $\min\{n\colon M(n+1)+(n-1) \leqslant 2^n\}$, if M is odd. He also gave an optimal questioning algorithm for arbitrary size M of the search space. In particular, this implies that whenever M is even, Berlekamp's volume bound can be achieved for $e = 1$. For the value $M = 1\,000\,000$, specifically mentioned by Ulam [117], this gives 25 questions in the worst case. The result from [91] also showed for the first time the difference in efficiency between adaptive and nonadaptive searching with lies. For 1 lie, the smallest search space for which this difference appears is $M = 21$: adaptive search can be done using 8 questions but the shortest 1-error-correcting code of size 21 is known to have length 9 (see [67]), and consequently this is the minimum length of the Questioner's nonadaptive strategy for $M = 21$.

The case $e = 2$ is more complicated, and was gradually solved in three consecutive papers. Czyzowicz, Mundici and Pelc [31] proved that the minimum length of the Questioner's strategy, for $M = 1\,000\,000$ or for $M = 2^{20}$, is 29. In a subsequent paper [32] the same authors solved the problem for all values of M which are powers of 2. It turns out that if $M = 2^m$, for any $m \neq 2$, then the volume bound can be achieved, i.e., the minimum length of the Questioner's strategy is $\min\{n\colon\ n^2 + n + 1 \leqslant 2^{m-n+1}\}$. For the particular case $m = 2$, the volume bound cannot be achieved: for a 4-element search space the corresponding lower bound is 7 but the minimum number of questions is 8. The problem of searching with at most two lies was completely solved by Guzicki [44]. He proved that, for any M, the minimum length of the Questioner's strategy is either equal to the volume bound or exceeds it by 1. He also characterized values of M for which the volume bound is achieved. The characterization depends on the remainder in division by 4 of M and of the value given by the volume bound.

The case $e = 3$ is the largest number of lies for which the Rényi–Ulam problem is solved for an arbitrary size M of the search space. Again the problem was solved gradually in four consecutive papers. Hill and Karim [48] and Negro and Sereno [84] showed independently that if $M = 1\,000\,000$ or $M = 2^{20}$, the minimum length of the Questioner's strategy corresponds to the volume bound for these numbers, i.e., it is 33. In [85] Negro and Sereno solved the problem for all values of M which are powers of 2. It turns out that if $M = 2^m$, for any $m \neq 2, 3, 5$, then the volume bound can be achieved, and for the exceptional values $2, 3, 5$, the minimum length of the Questioner's strategy exceeds the volume bound by 1. The problem of searching with at most three lies was completely solved by Deppe [35]. He showed that, for any M, the minimum length of the Questioner's strategy is either equal to the volume bound, or exceeds it by 1, or by 2. He also characterized values of M corresponding to each situation. (In fact, the length of the strategy exceeds the volume bound by at most 1, for all M except 3 and 5.) The fact that the volume bound is exceeded by at most one, apart from exceptional cases, has been previously showed by Auletta et al. [11], although they missed the exception $M = 5$.

For larger values of e, the exact results concerning the minimum length of the Questioner's strategy are valid only for special values of M. For $e = 4$, Auletta et al. [12] and Hill and Karim [48] showed independently that, if $M = 2^{20}$, the minimum length of the Questioner's strategy is equal to the volume bound, which is 37. Hill et al. [49] solved the Rényi–Ulam problem for all e, when the size of the search space is $M = 1\,000\,000$ or $M = 2^{20}$. They showed that for $e \geqslant 9$ the minimum length of the Questioner's strategy is $3e + 26$ in both cases, and found the missing values of this length for smaller e. For $e = 5, 6, 7, 8$, the number of questions is, respectively, 40, 43, 46, 50, if $M = 2^{20}$. For $e = 4, 5, 6, 7, 8$, the number of questions is, respectively, 36, 40, 43, 46, 49, if $M = 1\,000\,000$. The value 49 for $e = 8$ and $M = 1\,000\,000$ is due to

DesJardins [37]. Hill [47] reported results of Hill and Karim on the minimum length of the Questioner's strategy for $M = 2^i$, where $i < 20$, and for arbitrary e. He determined the exact length of this strategy, except for the cases $i = 14$ and $i = 19$, where a gap of 1 remained. The exact length for $i = 14$ was later found by Cicalese and Vaccaro [27].

Spencer [113] proved a general result showing that the possibility of winning the Rényi–Ulam game by the Questioner in a given number of questions (moves) depends on his ability of making e first "good" moves. Spencer [113] considered games with exactly n questions. He introduced the following definition: the Questioner can *survive* e moves in an n-move game, if after e questions the state $(x_0, x_1, \ldots, x_e)$ of the game has the property $V_{n-e}(x_0, x_1, \ldots, x_e) \leqslant 2^{n-e}$, regardless of the answers. He proved that for n sufficiently large, and for every size M of the search space, the Questioner wins the n-move game, if and only if, he/she can survive e moves. In particular, if $M = 2^m$, for $m \geqslant e$, then the Questioner can survive e moves in an n-move game, where n corresponds to the volume bound: this follows from the fact that in the first e questions all terms of the current state can be split exactly in halves, in this case. Consequently, Spencer's result implies that for all but finitely many m, the minimum length of the Questioner's strategy in the game with search space of size $M = 2^m$ corresponds to the volume bound.

On the experimental side, Lawler and Sarkissian [61] gave a simple suboptimal questioning strategy for the Rényi–Ulam game with arbitrary e and M, and conducted tests showing how this heuristic performs. An improved heuristic was proposed by Cicalese and Vaccaro [27] who also conducted experimental tests showing that their strategy is optimal for any $M = 2^m$, where $m \leqslant 16$, and any $e \leqslant 9$. It is interesting to note in this context the following asymptotic result due to Berlekamp [17]: fix the size M of the search space and denote by $n(M, e)$ the minimum length of the Questioner's strategy. Then $n(M, e)/e \to 3$, as $e \to \infty$. Thus, for a fixed M and large e, the Questioner must ask approximately $3e$ questions (cf. the number $3e + 26$, when $M = 1\,000\,000$).

3.1.2. Restrictions on adaptability

Restricting adaptability of the Rényi–Ulam game may be important in many practical applications. If the communicating agents are far apart in space, waiting for feedback after sending every bit considerably increasing time of communication. Likewise, if, e.g., a series of medical tests are conducted, it may be desirable to perform many tests in parallel without waiting for the results, in order to accelerate the diagnostic process. In such cases we would like to ask questions in the corresponding Rényi–Ulam game in as few batches as possible, and seek results only after each batch, in order to prepare the subsequent one.

The most extreme restriction on adaptability of the game is the case, already mentioned in Section 1, when the game is nonadaptive, i.e., when all questions have to be asked in a single batch, without waiting for answers, then all answers collected, and the unknown number revealed. As observed above, this scenario is equivalent to that of error-correcting codes, and thus we do not survey it, referring the reader to [67].

We only mention two important differences between the fully adaptive and the fully nonadaptive games. First, already for the case of one lie, the minimum lengths of the Questioner's strategy differ under these scenarios: as mentioned above, the smallest size of the search space for which this happens, is $M=21$. (On the other hand, Niven [86] observed that, in the case of one lie, this difference is at most 1, for any M.) Second, the knowledge about this minimum length is much less complete in the nonadaptive than in the adaptive case, already for small values of e. This is so even for sizes of the search space which are powers of 2 (which, as we have seen, are almost completely settled for the adaptive case). While in the case $e=1$, well known Hamming codes provide nonadaptive strategies of length corresponding to the volume bound, already for $e=2$ the situation is far from being completely understood. As opposed to the fully adaptive game, in which an optimal strategy is known (and in the case $M=2^m$, $m\neq 2$ has length corresponding to the volume bound [32]), shortest 2-error correcting codes of size 2^m are not known for many values of m (cf. [67]).

This lack of knowledge concerning fully nonadaptive games raises the question of how the situation changes if some small amount of adaptability is allowed. The least addition of adaptability is allowing two batches of questions. This assumption was considered by Cicalese and Mundici in [21, 23], for $e=2$, and for search spaces whose size is of the form 2^m. (For $e=1$ one batch is enough to match the volume bound, for such spaces.) They proved that the minimum length of the Questioner's strategy in a two-batch game is the same as in the fully adaptive game (and hence, in view of [32] it corresponds to the volume bound for all $m\neq 2$). Moreover, for all $m\neq 2,4$ this optimal strategy is of particularly simple *canonical* type: the first batch consists of m questions "Is the ith bit in the binary representation of the unknown number equal 0?", and all the remaining questions are in the second batch. The authors also proved that, for $m=4$, the shortest canonical two-batch strategy is by one longer than the optimal one. These results were subsequently generalized by Cicalese et al. in [24]. The main result of this paper states that for any positive integer e and for sufficiently large m, the minimum length of the Questioner's strategy in a two-batch game with e lies, played on a search space of size 2^m, corresponds to the volume bound. Moreover, this optimal strategy is the canonical one, described above. This result shows that, while nonadaptive searching with lies is often less efficient than the adaptive one, a minimum level of adaptability is enough to overcome this difference, for M which is a power of 2.

3.2. Restrictions on the types of questions

We now consider several variants of the fully adaptive Rényi–Ulam game in which the types of questions allowed are various restrictions of the general yes–no queries, i.e., the format of the questions is "Does the unknown number belong to the subset A of the search space?", for a restricted class of subsets of the search space. Such restrictions make it more difficult for the Questioner to evenly split the volume of the current state, which may increase the length of his/her strategy.

3.2.1. Comparison questions

The first restriction studied in the literature was that of *comparison questions* in the search space $\{0,\dots,M-1\}$, i.e., questions of the type "Is the unknown number smaller than a?", where $a \in \{0,\dots,M-1\}$. This was the setting in the paper [107], by Rivest et al., where the authors proved that the minimum length of the Questioner's strategy in the game with at most e lies is $\log M + e \log\log M + O(e \log e)$. Spencer [112] studied the Rényi–Ulam game with comparison questions and 1 lie. He proved that the minimum length of the Questioner's strategy is at most $\min\{k\colon\ M \leqslant (5/8)2^k/(k+1)\}$, for $M \geqslant 3$. This estimate exceeds the volume bound by at most 1, for any M. However, for the value $M = 10^6$ from the original question of Ulam, it leaves two possibilities for the minimum length of the Questioner's strategy: 25 or 26. Aigner [3] and Innes [50] proved independently that this minimum length is indeed 25. Auletta et al. [11] gave a comparison search algorithm assuming at most 2 lies, with the number of questions exceeding the volume bound by at most 2.

Negro et al. [83] considered comparison search in k batches of questions. This problem was studied for the first time by Pelc [93] in the case where all answers to questions are reliable. (Unlike for general yes–no questions, when adaptive and non-adaptive search without lies has the same length $\lceil \log M \rceil$, in the case when only comparison questions are allowed, restricting adaptability significantly influences efficiency of searching, even for reliable answers, cf. [93]). In [83] the authors constructed a k-batch comparison search algorithm assuming at most e lies, which uses the minimum number of questions, for arbitrary positive e and k.

Ambainis et al. [8] considered comparison searching with delays instead of lies. Questions have to be asked in consecutive time units but the answer to each question is given d time units after it was asked. In [8] the minimum length of the Questioner's strategy for this variation of the game was given, up to an additive constant. Cicalese and Vaccaro [28] considered the following generalization of the setting from [8]: comparison searching has to be performed by asking questions in consecutive time units, assuming that the answer to each question is given d time units after it was asked, and up to c questions may not be answered at all. (However, all obtained answers are reliable.) In [28] the minimum length of the Questioner's strategy was computed for arbitrary positive d, and for $c = 1$.

3.2.2. Interval and bi-interval questions

An interval (resp. bi-interval) question is a question of the form "Is x in the interval $[a,b]$?" (resp. "Is x in the set $[a,b] \cup [c,d]$?"), where $a < b$ and $c < d$. Hence this is a type of question slightly more general than comparison questions but much more restrictive than arbitrary yes–no questions. (An arbitrary yes–no question can be equivalently formulated as "Is x in $I_1 \cup \cdots \cup I_k$?", where $I_1,\dots,I_k$ are intervals included in the domain, and there is no restriction on the number k of intervals.)

The Rényi–Ulam game with bi-interval questions and 2 lies was considered by Mundici and Trombetta [81]. They proved that the minimum length of the Questioner's strategy in this game corresponds to the volume bound, for any size M of the search space which is a power of 2 and which is not equal to 4. They also showed that for interval questions this result does not hold.

3.2.3. Prefix questions

A prefix question is a question of the form: “Is the sequence t a prefix of the binary representation of the unknown number?”, where t is an arbitrary binary sequence. Pelc [92] considered searching with at most 1 lie using prefix questions, for search spaces whose size M is a power of 2. (For such spaces the game is equivalent to searching for an unknown leaf in a complete binary tree, by asking questions of whether the unknown leaf is in a given subtree.) In [92] the minimum length of the Questioner’s strategy was established for this variation of the Rényi–Ulam game. This length was compared to the minimum length of the Questioner’s strategy in the game with arbitrary yes–no questions (and at most 1 lie). It was shown that while the difference between these two lengths diverges to infinity as M grows, their ratio converges to 1.

3.2.4. “Small” questions

Macula [66] considered the nonadaptive Rényi–Ulam game with one lie, under the restriction that all questions must be of the form “Does the unknown number belong to the subset A of the search space?”, where A has at most k elements, for a fixed integer k. Macula constructed a Questioner’s strategy for this game, and gave upper bounds on the minimum number of questions.

A generalization of this problem to the case of at most e lies, for $e \geqslant 1$, was considered by Katona [52]. He determined the minimum number of questions sufficient to complete the search in an M-element search space, when $k \in \mathrm{O}(M^{\alpha})$, where $\alpha < 1$.

3.2.5. Variable cost questions

Sereno [109] considered a variation of the adaptive Rényi–Ulam game with arbitrary yes–no questions and e lies, in which at most B questions can get answer “yes”. The name *variable cost questions* comes from the assumption that the Questioner is charged only for ‘yes” answers, and his/her budget is restricted. Sereno derived the following lower bound on the length of the Questioner’s strategy for this variant of the game (on a search space of size M):

$$\min\left\{q\colon \sum_{i=0}^{B}\binom{q}{i} \geqslant M\cdot\sum_{i=0}^{e}\binom{q}{i}\right\}.$$

3.2.6. Nonrepetitive questions

Pelc [97] considered a variation of the Rényi–Ulam game with e lies (both adaptive and nonadaptive) in which questions cannot be repeated. This variant of the game comes from the assumption that errors are not caused by intermittent failures of the communication channel but are due to permanent faults of hardware units responsible for the execution of a test. (Thus repetition of a question to which an erroneous answer was given once, always results in an erroneous answer.) More precisely, the restriction imposed in [97] on possible questions is the following: for any pair of questions "$x \in A$?" and "$x \in B$?", B can be neither equal to A nor to the complement of A. Such questions are called *nonrepetitive*.

As opposed to the situation when questions can be repeated, in the nonrepetitive variation of the game search is not always feasible. In fact it was proved in [97] that search with e lies and nonrepetitive questions is feasible for a search space of size M, if and only if, $e < 2^{M-3}$. (As far as feasibility is concerned, the adaptive and nonadaptive versions of the game are equivalent.) As for the minimum length of the Questioner's strategy, it was investigated in [97] both for the adaptive and nonadaptive versions of the game, in the case of one lie. It was shown that, for the adaptive game on any search space of size $M > 3$, this length is the same as in the variant allowing repetitions of questions. (For $M \leqslant 3$ nonrepetitive search is not feasible, by the above mentioned characterization.) For the nonadaptive game with $M = 2^m$, the minimum length of the Questioner's strategy is the same as in the game with repetitions allowed, provided that $m \geqslant 3$. (In this case it is the length of the respective Hamming 1-error-correcting code.) If $m = 2$, one more question is needed in the nonrepetitive variation, as compared to the classic one: 6 questions instead of 5. Finally, for $m = 1$, nonrepetitive search is not feasible.

4. q-ary search with a fixed number of errors

q-ary search is a generalization of binary search in which the questions are of the form: "To which of the sets $A_1, A_2, \ldots, A_q$ the unknown number belongs?", where $A_1, A_2, \ldots, A_q$ is a partition of the search space. Thus binary search is the same as 2-ary search. As in the binary case, both the adaptive and nonadaptive q-ary search can be considered. q-ary nonadaptive search with at most e lies corresponds to e-error-correcting codes over an alphabet consisting of q symbols.

q-ary search with lies, for $q > 2$, was first investigated by Pelc [94] under the formulation of unreliable coin weighing. More precisely, the problem studied in [94] was the following. There is one counterfeit coin among M coins, which is heavier than the other coins. Find the counterfeit coin using the least possible number of weighings on a beam balance, assuming that at most one weighing is unreliable. This formulation of the problem is equivalent to 3-ary search with one lie, under the additional assumption that in each test $|A_1| = |A_2|$. Indeed, the sets A_1, A_2 and A_3 in a question correspond to the sets of coins put on both scales and the set of coins left. The condition $|A_1| = |A_2|$ corresponds to the requirement that the number of coins put on each scale must be the same (as we do not know the difference between the weight of a good and a counterfeit coin). In [94] the above problem was solved exactly, for arbitrary M. Since subsequent weighings can be decided on the basis of previous weighings' results, the problem corresponds to adaptive 3-ary search.

q-ary adaptive search with one lie in an M-element space, for arbitrary q and M, was investigated first by Malinowski [68] and then by Aigner [4]. In both these papers the minimum length of the Questioner's strategy was established. Aigner [4] also considered q-ary nonadaptive search with one lie and proved the following result. Let $f(M)$ and $g(M)$ denote the minimum lengths of the Questioner's strategy in the q-ary adaptive (resp. nonadaptive) search with one lie, and assume that q is a prime power. Then $g(M) = f(M)$ if $M \leqslant q^{q-1}$, and $f(M) \leqslant g(M) \leqslant f(M) + 1$ otherwise.

q-ary adaptive search with two lies was investigated by Cicalese [19] and Cicalese and Vaccaro [26]. The main result of these papers was establishing the minimum length of the Questioner's strategy for arbitrary $q \geqslant 2$, and for M being a power of q. Generalizing Berlekamp's [17] notion, the kth *volume* of a state $(x_0, x_1, \ldots, x_e)$ corresponding to q-ary search with e lies is the number

$$V_k(x_0, x_1, \ldots, x_e) = \sum_{i=0}^{e} x_i \sum_{j=0}^{e-i} (q-1)^j \binom{k}{j}.$$

The result from [19, 26] says that, with the only exception of $q = m = 2$, the minimum length of the Questioner's strategy in the q-ary search with two lies, for size $M = q^m$ of the search space, is the least k satisfying inequalities $V_k(M, 0, 0) \leqslant q^k$ and $V_{k-2}(0, M, 0) \leqslant q^{k-2}$. (Recall that the case $q = m = 2$ was settled by Czyzowicz et al. [32].)

Cicalese and Vaccaro [26] also studied the minimum length of the Questioner's strategy in q-ary adaptive search with two lies, for an arbitrary size M of the search space. They proved that, if $\lceil \log_q M \rceil < q$ then this length is $\lceil \log_q M \rceil + 4$, and if $\lceil \log_q M \rceil \geqslant q - 1$ then this length is either k_0 or $k_0 + 1$, where k_0 is the least k satisfying inequalities $V_k(M, 0, 0) \leqslant q^k$ and $V_{k-2}(0, M, 0) \leqslant q^{k-2}$. They also obtained partial results concerning the case of an arbitrary number of lies, disproving a conjecture of Aigner [4].

The above results concerning q-ary search with two lies were further extended by Cicalese and Mundici [23]. They studied the minimum length of the Questioner's strategy in such a search, using the minimum amount of adaptability exceeding the nonadaptive setting, i.e., in the situation when questions have to be asked in two batches. Extending their result on binary search (from the same paper) they proved that, if M is a power of q and q is arbitrary, then the minimum length of the Questioner's strategy in a 2-batch search corresponds to the volume bound, for sufficiently large M. Moreover, for all M which are powers of q, this minimum length is equal in the 2-batch search and in the fully adaptive search.

q-ary adaptive search with an arbitrary number of lies was studied by Muthukrishnan [82]. He developed a Questioner's strategy whose length differs from the minimum possible length by at most 1, for any q, e and sufficiently large M. In fact Muthukrishnan showed that this result holds not only for a fixed number of lies but also when the number of lies is a slowly growing function of the length of the game.

5. Other restrictions on error types and patterns

In this section we consider variants of the Rényi–Ulam game in which the limitation of lies is different from imposing a fixed upper bound on their number during the entire game. In fact, it may be more realistic to assume that the number of errors grows with the length of the game. This is the case, e.g., for random errors, the first scenario considered by Rényi [105].

5.1. Bounded error fraction

We begin by discussing a type of limitation of lies to which a lot of attention was devoted in the literature: imposing a bound on the *fraction* of erroneous answers. Two main scenarios were considered in this context, depending on the precise way in which the restriction is formulated. The fraction of errors can be globally bounded or prefix-bounded.

5.1.1. Globally bounded error fraction

We say that the fraction of errors is *globally bounded* by a constant $p<1$, if the total number of erroneous answers must be at most pn, where n is the total number of questions. We first present the results concerning the Rényi–Ulam game in this error model and with arbitrary yes–no questions.

The study of searching with bounded error fraction was originated by Pelc [90] who considered nonadaptive search with globally bounded error fraction, formulated as a problem in coding theory. He proved that if $p<1/4$ then there exists a Questioner's strategy of length $O(\log M)$, where M is the size of the search space, and that no strategy can exist (search is impossible), if $p>1/4$ for sufficiently large M. (Due to a typographic error in [90], the latter assumption was mistakenly stated as $p\geqslant 1/4$.) The above result was later rediscovered by Spencer and Winkler [114] who also proved that for $p=1/4$ the minimum length of the Questioner's strategy is $\Theta(M)$. In the same paper Spencer and Winkler considered adaptive search with globally bounded error fraction. They proved that if $p<1/3$ then there exists a Questioner's strategy of length $O(\log M)$, and if $p\geqslant 1/3$ then search is impossible for $M\geqslant 5$.

Later, Dhagat et al. [38] considered the same problems but with more restrictive questions: bit questions (of the format "Is the ith bit of the binary representation of the unknown number equal 0?"), and comparison questions. They showed that nonadaptive search with these types of questions is impossible for any positive p, when M is sufficiently large. As for adaptive search, they also showed that bit questions make the Questioner's task impossible for any positive p and sufficiently large M. On the other hand, comparison questions turn out to be sufficiently powerful to perform adaptive search using $O(\log M)$ questions, provided that $p<1/3$. (Recall that for $p\geqslant 1/3$ search is impossible even with arbitrary yes–no questions.) Recently Pedrotti [87, 88] improved the multiplicative constant in the $O(\log M)$ bound for comparison questions. This constant was further improved, for small values of p, by Albers and Damaschke [9]. They showed a simple strategy using $c(p)\log M$ comparison questions, where the constant coefficient $c(p)$ (depending on p) converges to 1 as $p\to 0$. The impossibility of search for $p\geqslant 1/3$ should be contrasted with the following result of Borgstrom and Kosaraju [18]: for any $p<1/2$ the Questioner can confine the unknown number to a set of size O(1), using $O(\log M)$ comparison questions.

5.1.2. Prefix-bounded error fraction

We say that the fraction of errors is *prefix-bounded* by a constant $p < 1$, if for any positive integer i, there are at most pi errors in the sequence of i initial answers. The study of searching in this error model was originated by Pelc [90] in the nonadaptive version with arbitrary yes–no questions, formulated as a problem in coding theory. He proved the following result: if $p < 1/4$ then there exists a Questioner's strategy of length $O(\log M)$, if $1/4 \leqslant p < 1/2$ then the minimum length of the Questioner's strategy is polynomial in M, and if $p \geqslant 1/2$ then search is impossible for $M > 2$.

Adaptive searching with prefix-bounded error fraction and comparison questions was first studied by Pelc [96]. He proved that search is possible if and only if $p < 1/2$, and proposed a Questioner's strategy of length $O(\log M)$, whenever $p < 1/3$. For $1/3 \leqslant p < 1/2$ the Questioner's strategy given in [96] has length $O(n^{\log(1/(1-2p))})$. Pelc stated the problem of whether there exists a Questioner's strategy of length $O(\log M)$ for $1/3 \leqslant p < 1/2$. This problem was further studied by Aslam and Dhagat [9]. They improved the length of the Questioner's strategy to $O(n^{\log(1/(1-p))})$ for comparison questions, and showed a Questioner's strategy of length $O(\log M)$ for arbitrary yes–no questions. The latter result was obtained independently by Spencer and Winkler [114]. The length $O(n^{\log(1/(1-p))})$ of the Questioner's strategy, for arbitrary $p < 1/2$ and for bit questions, was obtained by Dhagat et al. [38]. The final solution of the problem for comparison questions is due to Borgstrom and Kosaraju [18] who showed a Questioner's strategy of length $O(\log M)$ for arbitrary $p < 1/2$.

5.1.3. Segment-bounded errors

An error model related to the assumption of bounded error fraction was studied by Czyzowicz, Lakshmanan and Pelc [30]. They investigated the adaptive game with arbitrary yes–no questions, and assumed that at most one lie is allowed in every sequence of r consecutive answers, for a fixed $r \geqslant 3$. Under this scenario, a Questioner's strategy of asymptotically optimal length was proposed in [30]. More precisely, the ratio of the length of the proposed strategy to the lower bound established in [30] converges to 1, as the size of the search space grows.

5.2. Random errors

The earliest studied way of limiting lies in Rényi–Ulam games is the scenario of random errors originally proposed by Rényi [105]. It is assumed that the Responder lies with fixed probability $p < 1/2$, independently for every answer. Rényi [105] studied the problem of *random search* in which questions are of the form "$x \in A$?", where subsets A of the search space of size M are chosen randomly and independently with the same probability 2^{-M}. (Obviously, for random search, the adaptive and nonadaptive versions of the search are equivalent.) Rényi [105] stated the problem of what is the minimum number $k(M)$ of random tests (questions) sufficient to perform correct search with probability at least α, for a fixed parameter $0<\alpha<1$. He answered this question by showing that

$$k(M)=\frac{\log M+\mathrm{o}(\log M)}{1-I(p)},$$

where $I(p)=p\log(1/p)+(1-p)\log(1/(1-p))$ is the entropy of the probability distribution $(p,1-p)$. Notice that the order of magnitude of $k(M)$ does not depend on α.

Deterministic adaptive search with random errors and comparison questions was first investigated by Schalkwijk [108], for some particular values of error probability p. The general case was studied by Pelc [96]. The assumption on errors is as in [105], and the problem is to determine the minimum length of the Questioner's strategy which guarantees correct search with probability at least α, for a fixed parameter $0<\alpha<1$. Pelc [96] showed a Questioner's strategy of length $\mathrm{O}(\log M)$ for arbitrary $p<1/3$, and observed that a simple strategy of repeating each question of the binary search $\mathrm{O}(\log M)$ times (with an appropriate multiplicative constant), and taking the majority answer in each case, gives a strategy of length $\mathrm{O}(\log^2 M)$, for arbitrary $p<1/2$. He asked if a strategy of logarithmic length exists for $1/3\leqslant p<1/2$. However, his method of obtaining a logarithmic strategy for $p<1/3$ is derived from an analogous result for the prefix-bounded error model. Applying the same argument, and using the above quoted result of Borgstrom and Kosaraju [18] for the prefix-bounded error model, gives a positive answer to this problem.

5.3. Arbitrary error patterns

A very general way of restricting lies of the Responder is the following. Fix a priori the length of the entire game, i.e., the number n of questions to be asked. Moreover, fix a set $\mathscr{S}$ of binary sequences of length n, called the set of *lie patterns*. The size M of the search space, the length n of the game, and the set of lie patterns are known to both players. The Responder chooses a lie pattern $s\in\mathscr{S}$ unknown to the Questioner, along with the unknown element of the search space. Then the Responder gives an erroneous answer to the ith question, for any $i\leqslant n$, if and only if $s(i)=1$. (Notice that while the lie pattern chosen by the Responder is unknown to the Questioner, it is fixed throughout the game.) The problem is for which n and $\mathscr{S}$ the Questioner has a strategy which permits him/her to carry out the search on a space of size M, regardless of the Responder's choices.

Searching with arbitrary error patterns was first investigated by Ravikumar and Lakshmanan [103] in the different context of continuous search, hence we postpone the description of their result to Section 6.1. Pelc [89] gave characterizations of sets of lie patterns which make possible search for M being a power of 2, and for Boolean combinations of bit questions. Czyzowicz, Lakshmanan and Pelc [29] studied the problem of searching with a *forbidden* lie pattern. This is the following variation of the above setting. A binary sequence γ of length k is fixed a priori, and is known to both players. The Responder may choose any lie pattern which does not contain γ as a substring. More precisely, lies of the Responder may be arbitrary, as long as no k consecutive answers follow the lie pattern γ. (Notice that under this scenario the length n of the entire game is not fixed a priori.) The authors of [29] characterized forbidden lie patterns for which search is feasible. These are precisely strings (0), (1), (01), and (10). (The first two forbidden lie patterns correspond to trivial situations of never saying the truth or never lying.) For these four forbidden lie patterns, a Questioner's strategy of minimum length was proposed in [29].

A game related to searching with a specified set of allowed lie patterns was considered by Yaglom and Yaglom [119]. Given cities A, B, and C, such that people in A always say the truth, those in B always lie, and those in C alternate one lie and one correct answer, determine the minimum number of yes–no questions to find out in which city we are.

5.4. *Half-errors*

We finally consider the following limitation of the Responder's lies: the total number of lies is at most e but all "yes"-answers are correct—lies are limited to "no"-answers. This asymmetric assumption is justified, e.g., by the application of Rényi–Ulam games in communication through a noisy channel. In optical communication (cf. [99]), a photon sent through the channel can remain undetected at the receiving end but detecting a photon that has not been sent is impossible. This variation of the game is referred to as *searching with e half-lies*.

Adaptive searching with e half-lies, using comparison questions, was first studied by Rivest et al. [107]. The authors showed that their expression $\log M + e\log\log M + \mathrm{O}(e\log e)$ for the minimum length of the Questioner's strategy, established for the case of "full" lies, also holds for half-lies. Cicalese and Mundici [22] studied adaptive searching with one half-lie and arbitrary yes–no questions, in the case when the size of the search space is a power of 2. They gave upper and lower bounds on the minimum length of the Questioner's strategy, differing only by one. More precisely, they proved that this length is either $q-1$ or $q-2$, where q is the volume bound for the case of one ("full") lie. (Understandably, since the lying power of the Responder is decreased, the Questioner may have a strategy shorter than the lower bound for the "full" lie game.) Moreover, Cicalese and Mundici [22] showed a Questioner's strategy which has minimum length for infinitely many sizes of the search space.

6. Searching in infinite spaces

In the previous sections we considered Rényi–Ulam games played on a finite space of size M. In this case the minimum length of the Questioner's strategy depends on M and on parameters characterizing lies, e.g., the upper bound e on the total number of lies. We now discuss two variants of the game played on infinite spaces: *continuous* *search* in which the search space is the interval $(0,1)$, and *unbounded search* played on the set N of positive integers.

6.1. Continuous search

Continuous search is conducted on the interval $(0,1)$, and permissible questions are either arbitrary questions of the form "$x \in A$?", where—in order to avoid technical problems—subsets A of $(0,1)$ must be Lebesgue measurable, or comparison questions of the form "$x < a$?", where $0 < a < 1$. It is clear that the unknown number x cannot be found in finitely many questions in the worst case, so the aim of the search is modified by requiring only to find a subset $U \subset (0,1)$ of Lebesgue measure at most ε, where ε is a fixed positive real, such that $x \in U$. Lies are limited in one of the ways discussed previously, and the problem is to determine the minimum length of the Questioner's strategy, depending on the *accuracy* ε and on the parameter limiting lies. Optimal questioning strategies for continuous search are usually easier to design than in the finite case because there is no problem in equal splitting of the volume of the current state (defined analogously as for the finite search space).

Continuous search with at most e lies was first discussed by Rivest et al. [107]. They showed that the minimum length of the Questioner's strategy equals

$$\min\left\{q:\ \varepsilon \geqslant 2^{-q} \cdot \sum_{i=0}^{e} \binom{q}{i}\right\},$$

both for arbitrary and for comparison questions.

Continuous search in the model with prefix-bounded error fraction and comparison questions was considered by Pelc [96]. He showed a Questioner's strategy of length $O(\log n)$, where $\varepsilon = 1/n$, for any fixed error fraction $p < 1/2$. From this result he derived a similar strategy for the game with random errors. More precisely, for any $p < 1/2$, $q < 1$, and $\varepsilon = 1/n$, he showed a Questioner's strategy of length $O(\log n)$, performing the task with probability at least q.

Sereno [109] considered continuous search with variable cost questions: questions of arbitrary yes–no format with at most e lies, assuming that at most B questions can get answer "yes". For accuracy ε he showed that the minimum length of the Questioner's strategy is

$$\min\left\{q:\ \varepsilon \geqslant \frac{\sum_{i=0}^{e} \binom{q}{i}}{\sum_{i=0}^{B} \binom{q}{i}}\right\}.$$

Ravikumar and Lakshmanan [103] studied a variation of continuous search for arbitrary sets of lie patterns. They considered a Rényi–Ulam game on $(0,1)$ with exactly n arbitrary yes–no questions, and a set $\mathscr{S}$ of lie patterns. The main result of [103] was establishing the minimum worst-case size (Lebesgue measure) of a set U to which the Questioner can confine the unknown number. This minimum size is $2^{-n} \cdot |\mathscr{S}|$.

6.2. Unbounded search

Unbounded search is the task of finding an unknown positive integer, using either arbitrary yes–no questions, or comparison questions. Unbounded search with reliable answers was investigated, e.g., by Bentley and Yao [17] and Beigel [15]. Clearly, only adaptive search is feasible in this context, and the complexity of searching algorithms is given as a function of the unknown number m.

Unbounded search with errors, in the model with prefix-bounded error fraction p and comparison questions, was first investigated by Pelc [96]. He proved that search is possible if and only if $p<1/2$, and proposed a Questioner's strategy of length $O(\log m)$, whenever $p<1/3$. For $1/3 \leqslant p<1/2$ the Questioner's strategy given in [96] has length $O(m^{\log(1/(1-2p))})$. Pelc stated the problem of whether there exists a Questioner's strategy of length $O(\log m)$ for $1/3 \leqslant p<1/2$. This problem was further studied by Aslam and Dhagat [9]. They improved the length of the Questioner's strategy to $O((m\log^2 m)^{\log(1/(1-p))})$ for comparison questions, and showed a Questioner's strategy of length $O(\log m)$ for arbitrary yes–no questions.

Pelc [96] also showed that his result in the model with prefix-bounded error fraction gives an analogous result in the random error model: a Questioner's strategy of length $O(\log m)$ with arbitrary reliability $q<1$, whenever $p<1/3$. Likewise, the above result of Aslam and Dhagat [9] implies the existence of a Questioner's strategy of length $O(\log m)$ with arbitrary reliability $q<1$, for any $p<1/2$, if questions are arbitrary.

7. Other search problems with errors

Searching for an unknown number in a set is only one among many examples of search problems. Other examples include searching for an element of a set equipped with some structure, such as a linear order, or the structure of a graph. In this framework, sorting is equivalent to searching for an unknown permutation, and selection is searching for the index of the largest element of a sequence. Likewise, many graph problems can be formulated in terms of searching for a node or a set of nodes with specified properties. In this section we consider such problems, assuming that answers to questions (tests) used in the process of finding the unknown objects may be erroneous.

7.1. Sorting

Sorting can be viewed as an (adaptive) search game in which the Responder chooses a sequence $(x_1,\ldots,x_n)$ of distinct integers, and the Questioner has to sort it (i.e., find a permutation π of $1,\ldots,n$, such that $x_{\pi(i)}<x_{\pi(i+1)}$, for all $i<n$), using comparison questions of the form "$x_i<x_j$?". Sorting with erroneous answers was first investigated by Lakshmanan et al. [60]. They assumed that the Responder may lie at most e times (where e can be a function of n), and considered both the "full-lie" and "half-lie" versions of the game (cf. Section 5.4). They proved that for the "half-lie" version the

minimum length of the Questioner's strategy is $O(n \log n + e)$, by giving asymptotically matching upper and lower bounds. For the "full-lie" version, they gave the lower bound $\Omega(n \log n + en)$ and the upper bound $O(n \log n + en + e^2)$, which are asymptotically tight if e is linear in n. The above upper bound was subsequently strengthened independently by Bagchi [13] and by Long [64] to $O(n \log n + en)$, which is optimal to within a constant factor.

A result on sorting in the prefix-bounded error model was obtained by Aigner [5], as a corollary to his results on finding the maximum, which was the main subject of [5]. For the error fraction bounded by $p < 1/2$ he showed a sorting strategy using $O((1/(1-p))^{O(n \log n)})$ comparisons, and derived a lower bound $\Omega((1/(1-p))^n)$. This lower bound was proved before by Borgstrom and Kosaraju [18] in a stronger form: they showed that it holds in the prefix-bounded error model even for the easier task of verifying if a list is sorted. They also observed that an algorithm solving this easier problem using $O((1/(1-p))^n)$ questions can be constructed.

Sorting with random errors was studied by Feige et al. [41]. They assumed that answers to comparison questions are erroneous with probability $p < 1/2$, independently for all questions, and required sorting to be correct with probability at least $1-q$, where $0 < q < 1/2$. They proved that the minimum length of the Questioner's strategy is $\Theta(n \log(n/q))$, both when the strategy is deterministic, and when randomization in the Questioner's decisions is allowed.

A related problem is that of sorting using comparator networks containing faulty comparators. A comparator is a two-input, two-output device which receives a number on each of its inputs, and outputs these numbers in a sorted order. A sorting comparator network is built of comparators, receives as input any permutation of numbers $1, \dots, n$, and outputs these numbers in a sorted order. Registers store elements to be sorted, and comparators connect pairs of registers. All comparators in a network can be partitioned into levels, with no more than one comparator per level connected to any register. The number of levels is the *depth* of the network, while the total number of comparators in the network is its *size*. The depth of a comparator network is proportional to the time it uses to accomplish its task, as comparators at the same level can act simultaneously. Hence it is important to construct comparator networks of smallest possible depth. For obvious economy reasons it is also important to minimize the size of the network. A simple network of depth $O(\log^2 n)$ to sort n elements was given by Batcher [14], and a network of asymptotically optimal depth $O(\log n)$ was given by Ajtai et al. [6].

Comparators can be subject to two kinds of faults: benign, which cause the output to remain unchanged, as if the input did not pass through a comparator at all, and destructive, which may output the input permuted arbitrarily, or even output one of the input values on both outputs. The study of sorting comparator networks with benign faults was initiated by Yao and Yao [120], under two scenarios, one assuming random faults, and the other imposing a fixed upper bound on their number. The latter scenario was further investigated, e.g., by Piotrów [100] and Stachowiak [115]. Sorting comparator networks with destructive faults were later studied, e.g., by Assaf and Upfal [10] and by Leighton and Ma [63]. In each of these papers the goal was to minimize the size and/or depth of the network while guaranteeing correct sorting (with high probability, if faults are random) under the given fault assumptions.

7.2. Selection

Similarly as sorting, the selection problem, i.e., finding the maximum of a sequence of numbers, can be viewed as an (adaptive) search game in which the Responder chooses a sequence $(x_1,\dots,x_n)$ of distinct integers and the Questioner has to find the index of the largest of them, using comparison questions of the form "$x_i < x_j$?".

The selection problem with erroneous answers was first studied by Ravikumar et al. [102]. They considered the game with at most e erroneous answers, under the "full-lie" and the "half-lie" scenarios. For the 'full-lie" scenario they proved that the minimum length of the Questioner's strategy is $(e+1)n-1$, and showed such an optimal strategy. For the "half-lie" scenario (when all lies are confined to "no" answers) they showed a strategy using at most $e+1$ questions, if $n=2$, and at most $2n+2e-4$ questions, for $n\geqslant 3$ and $e\geqslant 1$. Moreover they showed that the length of this strategy exceeds the lower bound by at most 2, for any n and e.

Aigner [5] considered the problem of simultaneously finding the maximum and the minimum of a sequence of numbers, assuming at most e erroneous answers. He gave upper and lower bounds on the minimum length of the Questioner's strategy for this task.

Aigner [5] also considered finding the maximum in models with bounded error fraction. His main result concerned the model with prefix-bounded error fraction p: the Questioner has a winning strategy if and only if $p<1/2$, and in this case the minimum length of the Questioner's strategy is $\Theta((1/(1-p))^n)$. Aigner [5] also considered the model with globally bounded error fraction p, both in the adaptive and in the nonadaptive versions. For the adaptive version he showed that the Questioner has a winning strategy if and only if $p<1/(n-1)$, and in this case he/she can win using $n-1$ questions. For the nonadaptive version Aigner [5] showed that the Questioner has a winning strategy if and only if $p<1/\binom{n}{2}$, and in this case he/she can win using $\binom{n}{2}$ questions. In both cases the length of the Questioner's strategy cannot be improved, even in the error-free setting.

Selection with random errors was studied by Feige et al. [41]. They assumed that answers to comparison questions are erroneous with probability $p<1/2$, independently for all questions, and required selection to be correct with probability at least $1-q$, where $0<q<1/2$. They proved that the minimum length of the Questioner's strategy is $\Theta(n\log(1/q))$, both when the strategy is deterministic, and when randomization in the Questioner's decisions is allowed. They also generalized this result for the problem of selecting the kth largest element, and proved tight bounds $\Theta(n\log(n/q))$ for the task of merging.

The problem of finding the minimum using faulty comparator networks was considered by Denejko et al. [34]. They assumed that the number of faulty comparators is bounded from above by a constant, and considered benign faults (cf. Section 7.1). In [34] upper and lower bounds on the minimum depth of such networks were proved.

A problem related to selecting the largest element with errors was investigated by Adler et al. [1]. The authors considered tournaments whose aim is selecting the best player with probability at least q, assuming that the best player wins a game with any other player with probability larger than $1/2$ (possibly different for different players). Outcomes of games between other players are decided according to various rules, giving rise to numerous models. The most general is the *adversary model* in which the adversary controls outcomes of all games that do not involve the best player. The goal of [1] was minimizing the number of rounds in the tournament, where in each round each player participates in at most one game. The authors gave upper and lower bounds on the minimum number of rounds necessary to select the best player with probability at least q, under each of the investigated models.

7.3. Searching in graphs

The problem of *searching in graphs with uncertainty* is formulated as follows. A token (object or piece of information sought) is located in an unknown node of an undirected graph. An agent starts at some node of the graph and has to find the token travelling along edges of the graph, using as few edge traversals as possible. In every node which does not hold the token, the agent gets an *advice* which is the edge on the shortest path to the token, incident to this node. This advice, however, is unreliable.

This problem was introduced by Kranakis and Krizanc [57]. In this paper they assumed that the graph is either a ring or a torus, and that advice is correct with probability p in every node, independently for different nodes. The authors proposed near optimal algorithms of searching for the token under this scenario. In [51, 55] the same problem was investigated for complete graphs. In these papers optimal searching strategies, both deterministic and randomized, were investigated, depending on the amount of local memory available at nodes. In [46], Hanusse et al. studied the problem of searching in graphs under the assumption that the number of liars, i.e., nodes that give a wrong advice, is bounded by a constant. For this scenario they proved upper and lower bounds on the optimal number of agent's steps for complete graphs, rings, tori, hypercubes, and trees.

7.4. Group testing

Group testing is the problem of finding several unknown objects in a search space, using subsets of the search space as tests. Depending on the type of feedback obtained, there are several possible models. One of the most popular models assumes that response to a test T is 1 if at least one of the unknown objects is in T, and 0 otherwise (cf. the book by Du and Hwang [40] for a comprehensive survey of group testing). The aim of group testing is to identify all unknown objects.

Du and Hwang [40] considered group testing with d unknown objects, assuming the prefix-bounded error model with constant p, i.e., under the hypothesis that for any positive integer i, there are at most pi errors in responses to the sequence of i initial tests. They proved that if $I(pd) < \log((d+1)/d)$, where $I(x) = x\log(1/x) + (1-x)\log(1/(1-x))$ is the entropy of the probability distribution $(x, 1-x)$, then there exists an algorithm to find all unknown objects in an M-element search space, using $O(\log M)$ (adaptive) tests.

De Bonis et al. [33] studied the problem of confining an unknown point in the unit square to a set of smallest possible (two-dimensional Lebesgue) measure, in the presence of unreliable tests. They viewed the problem as a variant of search for two unknown objects (coordinates of the point) in the unit interval. Possible tests are arbitrary measurable subsets of the unit interval, and two types of feedback giving rise to two models are considered. One is as described above, and the other is *parity* feedback, giving answer 1 to a test T if exactly one of the unknown objects x and y is in T, and answer 0 otherwise. De Bonis et al. [33] assumed that a total of q tests are available, and that at most e among them can give an erroneous answer. They showed that the smallest measure of a subset to which the unknown point can be confined under these assumptions is $2^{-(q+1)} \cdot \sum_{i=0}^{e} \binom{q}{i}$, and showed that this can be achieved by a nonadaptive strategy.

7.5. Error detection

Pelc [95] studied the following variations of the Rényi–Ulam game with e lies, on a search space of size M, with arbitrary yes–no questions. The rules of the game are as described in Section 3.1, except for the definition of the Questioner's win. In [95] two different definitions of win are considered, giving rise to two games called $D(M,e)$ and $D^{\#}(M,e)$. In the game $D(M,e)$ the Questioner wins if he/she can either find the unknown number or prove that the Responder lied. In the game $D^{\#}(M,e)$ the Questioner wins if he/she can either find the unknown number or determine the exact number of the Responder's lies, possibly without being able to tell where the lies occurred. It is easy to see that a Questioner's winning strategy in the nonadaptive variant of the game $D(M,e)$ is equivalent to constructing an e-error-detecting code of size M, i.e., a code of minimum Hamming distance $e+1$. Likewise, a Questioner's winning strategy in the nonadaptive variant of the game $D^{\#}(M,e)$ is equivalent to constructing a code of size M and minimum Hamming distance $2e$. Similarly as for error-correcting codes, the minimum length of a code with the above properties is unknown for most pairs of parameters M and e. Hence the same is true for the minimum length of the Questioner's winning strategy in nonadaptive variants of games $D(M,e)$ and $D^{\#}(M,e)$. Pelc [95] investigated this minimum length in adaptive variants of these games. He showed that this length is $\lceil \log M \rceil + e$ for the game $D(M,e)$, and it is between $\lceil \log M \rceil + e$ and $\lceil \log M \rceil + 3e$ for the game $D^{\#}(M,e)$. He also showed that the game $D^{\#}(M,2)$ is strictly more difficult for the Questioner than the game $D(M,2)$: the Questioner's win requires $\lceil \log M \rceil + 3$ questions in $D^{\#}(M,2)$.

7.6. Fault diagnosis

Fault diagnosis is one of the important problems in the design of fault-tolerant multiprocessor systems. Its aim is to precisely locate all faulty processors in the system, i.e., to answer the question which processors are faulty and which are fault free. The classic approach to fault diagnosis was originated by Preparata et al. [101]. The system is modeled as an undirected graph whose nodes are processors, and edges are links along which tests can be performed. Processors perform tests on their neighbors in the system, and diagnosis is based on the collection of test results. It is assumed that fault-free processors always give correct test results, while tests conducted by faulty processors are totally unpredictable: a faulty tester can output any test result, regardless of the status of the tested processor. More precisely, the outcome of a test performed by a processor u on its neighbor v is: "faulty" if u is fault-free and v is faulty, "fault-free" if both u and v are fault-free, and arbitrary otherwise. Faults are assumed permanent, i.e., the fault-status of a processor does not change during testing and diagnosis. Hence, under this model, fault diagnosis can be viewed as searching for faulty processors with unreliable tests. Indeed, an equivalent way of stating the problem is to find liars in a group containing honest people and liars, by asking some people about honesty of others.

In [101] a worst-case scenario was adopted: it was assumed that at most t processors are faulty and that they are placed in locations most detrimental for diagnosis. Later (see, e.g., surveys [69, 62]) many variations of the above model were considered by various authors. Indeed, several hundreds of papers were written on this subject in the last thirty years. Other fault scenarios were proposed, including the assumption that a fault-free tester detects a fault in a faulty neighbor only with some bounded probability (i.e., even tests conducted by a fault-free tester need not be reliable). Instead of imposing an upper bound on the number of faulty processors, some authors considered random faults (cf. [62]). Both adaptive and nonadaptive ways of testing were investigated, comparing efficiency of diagnosis in both cases. Finally, two measures of efficiency were proposed. One is the total number of tests used for diagnosis, and the other is the *time* of testing, i.e., the number of testing rounds, assuming that in the same round only tests involving disjoint pairs of processors can be conducted. In most of the papers in this domain, testing and diagnosis procedures were constructed, working under one of the numerous scenarios, and efficient with respect to one or both of the above criteria.

8. Related issues

In this section, we briefly mention some issues related to searching games with errors. The common feature of these topics is their fault-tolerant flavor: a task has to be completed efficiently, in spite of erroneous information obtained during the execution of the algorithm. The number and variety of fault-tolerant problems in computer science is much too large to even list them here. Hence we chose only a sample of them, coming from diverse domains.

8.1. Computing Boolean functions with errors

This problem can be also formulated as a two-person game. Let f be a Boolean function of n variables $x_1,\dots,x_n$, known to both players. The Responder knows values of $x_1,\dots,x_n$, and the Questioner has to find $f(x_1,\dots,x_n)$ by asking questions of the form "$x_i = 0$?". Some of the Responder's answers may be erroneous, and the Questioner wants to minimize the (worst-case) number of questions.

Computing Boolean functions with errors was first investigated by Kenyon and Yao [54], assuming that the total number of lies is at most e. They proved an upper bound on the minimum length of the Questioner's strategy for arbitrary Boolean functions, and a lower bound for monotone functions.

Computing Boolean functions with random errors was later investigated, e.g., by Feige et al. [41], Reischuk and Schmeltz [104] and Kenyon and King [57], They assumed that answers to questions "$x_i = 0$?" are erroneous with probability $p < 1/2$, independently for all questions, and required computation to be correct with probability at least $1 - q$, where $0 < q < 1/2$. In these papers upper and lower bounds on the minimum length of the Questioner's strategy were proved for various classes of Boolean functions.

The relations between Rényi–Ulam games and Boolean functions with errors (particularly the Maxsat problem) were also explored by Mundici [78].

8.2. Stochastic approximation

The theory of stochastic approximation also deals with computation procedures in which test results are prone to errors. However, in this context errors are defined differently: they are meant as measurement inaccuracies. The problem is to find zeros and extreme values of real-valued functions of one variable. Function values are to be determined by measurements which are subject to random errors. The goal is to construct a process which under some assumptions stochastically converges to the zero value which has to be found. The issue of efficiency is addressed by studying the rate of convergence. An exposition of stochastic approximation problems and the relevant literature can be found in the book by Ahlswede and Wegener [2].

8.3. Fault-tolerant communication

Algorithmic problems concerning communication in networks some of whose components are faulty, have been extensively investigated. Networks are often represented as graphs, and various fault models are considered. The two most commonly studied types of faults are *crash* and *Byzantine* faults. If the fault is a crash, the faulty node does not send or receive messages or the faulty link does not transmit messages: faulty components do not alter transmitted messages. Such faults are relatively benign.

Byzantine faults, on the other hand, are a worst-case fault scenario: faulty components can behave arbitrarily (even maliciously), by either stopping, rerouting, or altering transmitted messages in a way most detrimental to the communication process. Byzantine faults correspond to lies in searching games, as they provide unreliable information to communication algorithms.

Similarly as for searching games with errors, some limitations on the number of possibly faulty components must be imposed, otherwise no communication is possible. Two commonly used fault models are the *bounded* model and the *probabilistic* model. In the bounded model, an upper bound k is imposed on the number of faulty components, and their worst-case location is assumed. In the probabilistic model, faults are assumed to occur randomly and independently of each other, with specified probability.

Many communication tasks were investigated in the presence of faults. Some examples are:

- broadcasting: one fault-free node has a piece of information which has to reach all other fault-free nodes;
- information exchange (or gossiping): each fault-free node has a message, and all fault-free nodes have to learn all messages;
- consensus: every node has an initial value, and all fault-free nodes must agree on the same value in a *valid* way (if all nodes have the same initial value, they must agree on this value).

Two most important and widely studied efficiency measures of communication algorithms are the running time and the number of message transmissions used in the communication process. A lot of research has been devoted to construction and analysis of algorithms for various communication tasks, optimizing one or both of these measures, under different fault scenarios. The literature on fault-tolerant broadcasting and gossiping was surveyed, e.g., by Pelc [98], and comprehensive discussions of results on consensus, as well as many other problems in fault-tolerant distributed computing, can be found, e.g., in books by Tel [116] and Lynch [65].

8.4. Rényi–Ulam games, logic and algebra

Rényi–Ulam games turn out to have a natural correspondence to many-valued logic. These relations were extensively studied by Mundici [71–77, 79] and Mundici and Panti [80]. The approach to Rényi–Ulam games adopted in these papers significantly differs from that discussed previously. While most papers in this domain are concerned with efficiency of searching under various error scenarios, Mundici views Rényi–Ulam games as a tool for interpreting some problems in logic. Thus the analysis of the rules of the game becomes more important for this purpose than finding a good way of playing it.

The game with e lies corresponds to $(e+2)$-valued logic. We sketch this correspondence for $e=1$. The state of the Questioner's knowledge after k questions and answers can be represented as a triple (A_0, A_1, A_2) of subsets of the search space, where A_i is the set of elements which falsify exactly i previous answers (the triple of sizes of these sets is the state of the game at this point). An equivalent representation of this state of knowledge is a function $\psi : \{0, \ldots, M-1\} \to \{0, 1/2, 1\}$, where

- $\psi(x) = 1$ if $x \in A_0$ (x satisfies all previous answers),
- $\psi(x) = 1/2$ if $x \in A_1$ (x satisfies all but one previous answers),
- $\psi(x) = 0$ if $x \in A_2$ (x falsifies at least 2 previous answers).

This can be interpreted by saying that numbers in A_0 are possible solutions with logical value 1, those in A_1 with logical value 1/2, and those in A_2 with logical value 0. (This should be compared to the interpretation of the error-free game in classic two-valued logic: those numbers which satisfy all previous answers are possible solutions, and others are not.) Given this interpretation, all connectives of Lukasiewicz three-valued propositional logic can be defined using states of the Questioner's knowledge. For details we refer the reader to the above cited papers and references therein. Mundici [79] also discussed MV_3 algebras and cubic algebras in relation to Rényi–Ulam games with 1 lie, considering states of the Questioner's knowledge as elements of such algebras. (MV_3 algebras are algebraic interpretations of Lukasiewicz three-valued logic, similarly as Boolean algebras are interpretations of two-valued logic). The problem of rational betting in Rényi–Ulam games, and its connection with Lukasiewicz logic was also addressed by Gerla [42].

A different connection of Rényi–Ulam games with logic, more precisely with recursion theory, was investigated by Kummer and Stephan [58]. They considered the following two notions related to computational complexity of functions. The *query complexity* of a function $f : N \to N$, recursive in the oracle $X \subseteq N$, where N is the set of natural numbers, is the minimum number of queries to X sufficient to compute this function. The *enumeration complexity* of $f : N \to N$ is the minimum number k of values such that there exists an algorithm which, for any natural x, outputs a set of size at most k containing $f(x)$. Query complexity of a function is known to be uniquely determined by its enumeration complexity. Kummer and Stephan [58] consider the fault-tolerant version of query complexity, based on *e-robust* algorithms. Such algorithms are required to output the correct value even if answers to up to e queries are incorrect. The authors characterized e-robust query complexity in terms of enumeration complexity and of the minimum length of the Questioner's strategy for the Rényi–Ulam game with e lies. They showed that e-robust query complexity is equal to this minimum length for the Rényi–Ulam game on a search space of size equal to enumeration complexity.

9. Open problems

Proposing open problems in a domain so vast and so extensively studied as searching games with errors is a formidable task. The choice of such problems must be inevitably biased by the interests of the proposer, and can hardly reflect any objective criteria of importance, if such criteria exist at all. Also, such a list of problems can hardly pretend to completeness, mainly due to limits of knowledge and imagination of the proposer, and also due to lack of space. Therefore we present the following sample of open problems simply as a collection of questions which this author finds interesting. In order to limit the scope, we decided to eliminate from the list problems concerning well established domains neighboring searching games with errors, such as, e.g., the theory of error-correcting codes, fault diagnosis, or fault-tolerant communication. The order of problems follows the order of presentation of the corresponding topics in this survey.

Let us first consider binary search with a fixed number of errors and arbitrary questions. The growing complexity of the formulations of the results of Pelc [91], Guzicki [44] and Deppe [35] concerning the minimum length of the Questioner's strategy for an arbitrary size of the search space, and for one, two, and three lies, respectively, does not seem to give hope for a closed formula describing this length in the general case. Spencer's [113] result gives the precise value of this length only for some pairs of the size M of the search space and the number e of lies. Hence one way of formulating a problem whose solution would, in some sense, close the Rényi–Ulam problem for arbitrary parameters would be:

- Construct an efficient algorithm which, given parameters M and e, returns the minimum length of the Questioner's strategy for the Rényi–Ulam game with e lies, on a search space of size M.

A "companion problem" to the above is:

- Construct an efficient algorithm which, given parameters M and e, simulates the optimal Questioner in the Rényi–Ulam game with e lies, on a search space of size M.

Our next set of problems concerns adaptabiity. While Hamming perfect codes show that the minimum length of the Questioner's strategy is the same in the adaptive and nonadaptive games, for M being a power of 2 and $e=1$ lie, this fact does not hold for other values of M and e. Cicalese and Mundici [23] showed that for M being a power of 2 and $e=2$ lies, the minimum length of the Questioner's strategy is the same in the 2-batch game and in the fully adaptive game. This naturally leads to the following problem:

- Given arbitrary parameters M and e, what is the minimum number k of batches, such that the minimum length of the Questioner's strategy is the same in the k-batch game and in the fully adaptive game?

This seems a difficult question, since the minimum length of the Questioner's strategy in the fully adaptive game is not known for arbitrary M and e. However, the question seems interesting already for small values of e, where the optimal Questioner's strategy in the adaptive game is known. In particular, the following problem is open:

- Is the minimum length of the Questioner's strategy the same in the fully adaptive and in the 2-batch game with at most 1 lie, for all values of M?

More generally, it seems interesting to explore the role of adaptability in other settings as well:

- Given any Rényi–Ulam game, with lies limited in some way (bounded error fraction, random errors, etc.), and some format of questions (arbitrary yes–no questions, comparison questions, q-ary search, etc.), what is the minimum number k of batches, such that the minimum length of the Questioner's strategy is the same in the k-batch game and in the fully adaptive game?

The next problem is fairly specific, and concerns unbounded search in the prefix-bounded error model. Pelc [96] asked if there exists a Questioner's strategy of length $O(\log m)$, for an arbitrary bound $p<1/2$ on the error fraction. Aslam and Dhagat [9] showed such a strategy for arbitrary yes–no questions. For comparison questions they showed a strategy of length $O((m\log^2 m)^{\log(1/(1-p))})$, and this seems to be the best result to date. Hence the following problem remains open:

- Fix an arbitrary $p<1/2$. Is it possible to find an unknown natural number m using $O(\log m)$ comparison questions, if for any positive integer i, there are at most pi errors in the sequence of i initial answers?

The format of comparison questions is the most extensively studied, and chronologically the first restriction of questions in Rényi–Ulam games. In the case of 1 lie, the result of Spencer [112] implies that the minimum lengths of the Questioner's strategies for the games with comparison questions and with arbitrary yes–no questions differ by at most 1, for any size of the search space. However, the following problem remains open:

- Consider the adaptive Rényi–Ulam game with one lie. Is the minimum length of the Questioner's strategy in the variant of the game with comparison questions the same as in the variant with arbitrary yes–no questions?

An interesting related question can be asked on the basis of the result of Mundici and Trombetta [81]. They studied the impact of the complexity of the question format on the efficiency of searching in Rényi–Ulam games. More precisely, they compared the minimum length of the Questioner's strategy in the adaptive Rényi–Ulam game with bi-interval questions to that with arbitrary yes–no questions. They proved that this minimum length is the same for both variations, in the special case of the search space whose size is a power of 2, and of two lies. This gives rise to the following more general problem:

- Consider an adaptive Rényi–Ulam game with e lies, on a search space of arbitrary size. Are the minimum lengths of the Questioner's strategies in the variation with bi-interval questions and with arbitrary yes–no questions the same?

Recall that Mundici and Trombetta [81] showed that if bi-interval questions are replaced by interval questions, then the answer is no, even for search spaces of size which is a power of 2, and for two lies.

We conclude this section by a fairly general problem inspired by the papers [66] by Macula and [109] by Sereno. The former considered Rényi–Ulam games with questions "$x\in A$?", where A is a "small" subset of the search space, and the latter—a particular example of questions with variable cost under restricted budget. These two aspects can be related in some applications. Indeed, it may happen that the cost of a test depends on its size. For example, in the search for a faulty component of a complex mechanism, a test involving a large subset of components is likely to be costly, as all of them may, e.g., have to be connected to an electric circuit. Hence we may have a cost function $c:\{1,\dots,M\}\to R^+$ describing the charge

$c(|A|)$ for a test "$x \in A$?". Under this assumption the following question seems natural:

- Given any Rényi–Ulam game with arbitrary yes–no questions, and with errors limited in some way (fixed number of errors, bounded error fraction, random errors, etc.), and given a cost function as above, what is the minimum cost of the Questioner's strategy?

The classic question about the minimum length of the Questioner's strategy is a special case of the above, for the constant cost function 1.

10. Previous surveys and this one

We conclude this paper by comparing it to previous surveys of the domain of Rényi–Ulam games. Chronologically the first was the 1993 book by Du and Hwang [40], and more precisely Chapters 8 and 10.3 of it. Next was the 1995 survey article by Hill [47], and quite recently the 2001 book chapter [25] by Cicalese, Mundici and Vaccaro, and the 2001 survey article by Deppe [36]. Cicalese's Ph.D. Thesis [20] also contains a technical survey of Rényi–Ulam games and their connections with broadcasting and computational learning.

The reason of writing yet another survey of searching games with errors is the belief that the scope and goals of the present work are different from the above cited surveys, and therefore this paper may satisfy different, and hopefully complementary needs. In the above-mentioned surveys the authors decided to select a few topics in this vast domain, giving *in-depth* presentation of the results, techniques, proofs, and examples concerning the chosen problems. The goal of the present work is different. This survey is mainly meant as a guide. Hence we tried to cover the results concerning searching games with errors as *broadly* as possible, discussing many variations of the underlying models, and showing ramifications of Rényi–Ulam games going beyond the classic search problem, and dealing with other error-prone search tasks, such as sorting, selection, graph search, or diagnosis. This choice of showing a "large perspective" implied—due to space limitations—the tough decision to omit all proofs. We hope that the reader, knowing what was done and where, will refer to the cited original papers to learn the techniques and the methodology.

Our other aim was giving a historic perspective. Being involved in the domain of searching games with errors (as a contributor, observer and referee) for the last fifteen years, the present author witnessed many times independent discoveries or rediscoveries of results, incorrect citations of some previous work, and sometimes even some important papers falling into oblivion. The most notable case is that of Rényi's paper [105] which—to the best of our knowledge—is the first paper mentioning a searching game with errors as a game (as opposed, e.g., to earlier studied error-correcting codes which are only implicitly equivalent to a variation of such games), precedes other papers in the domain by many years, and yet is not cited in the literature of the domain. One of our concerns was to show chronological development of various topics in the domain of searching games with errors, describing not only the last, strongest result but often the incremental developments leading to the final solution of a problem. We hope that this survey gives proper credit in such cases and contributes to eliminate some of the chronological inaccuracies, without creating too many new ones. As far as errors in this paper are concerned, we hope that the surveyed work describing ways of learning the truth in spite of errors can be applied also to this article in a self-referential manner.

Acknowledgements

The author is grateful to Ferdinando Cicalese, Ugo Vaccaro, and to the anonymous referee for their useful suggestions and comments.

References

[1] M. Adler, P. Gemmell, M. Harchol-Balter, R.M. Karp, C. Kenyon, Selection in the presence of noise: the design of playoff systems, Proc. 5th Annu. ACM-SIAM Symp. on Discrete Algorithms 1994, pp. 564–572.
[2] R. Ahlswede, I. Wegener, Search Problems, Wiley, New York, 1987.
[3] M. Aigner, Ulams Millionenspiel, Math. Semesterber. 42 (1995) 71–80.
[4] M. Aigner, Searching with lies, J. Combin. Theory Ser. A 74 (1996) 43–56.
[5] M. Aigner, Finding the maximum and minimum, Discrete Appl. Math. 74 (1997) 1–12.
[6] M. Ajtai, J. Komlos, E. Szemeredi, An $O(n \log n)$ sorting network, Combinatorica 3 (1983) 1–19.
[7] F. Albers, P. Damaschke, Delayed correction—binary search with errors made very simple but efficient, in: X. Lyn (Ed.), Proc. 4th Auskalasian Theory Symp., CATS'98, Australian Computer Science Communications, vol. 20(3), Springer, Berlin, pp. 97–105.
[8] A. Ambainis, S.A. Bloch, D.L. Schweizer, Playing twenty questions with a procrastinator, Proc. 10th Ann. ACM-SIAM Symp. on Discrete Algorithms, 1999, pp. S844–S845.
[9] J.A. Aslam, A. Dhagat, Searching in the presence of linearly bounded errors, Proc. 23rd Annu. ACM Symp. on Theory of Computing, 1991, pp. 486–493.
[10] S. Assaf, E. Upfal, Fault-tolerant sorting networks, Proc. 32nd Annu. IEEE Symp. on Foundations of Computer Science, 1990, pp. 275–284.
[11] V. Auletta, A. Negro, G. Parlati, Some results on searching with lies, Proc. 4th Italian Conf. on Theoretical Computer Science, 1992, pp. 24–37.
[12] V. Auletta, A. Negro, G. Parlati, Solution of Ulam's problem on binary search with four lies, Technical Report, Columbia University, 1993, CUCS-034-93.
[13] A. Bagchi, On sorting in the presence of erroneous information, Inform. Process. Lett. 43 (1992) 213–215.
[14] E.B. Batcher, Sorting networks and their applications, Proc. AFIPS Spring Joint Computer Conference, Montvale, NJ, June 1968, pp. 307–314.
[15] R. Beigel, Unbounded searching algorithms, SIAM J. Comput. 19 (1990) 522–537.
[16] J.L. Bentley, A.C. Yao, An almost optimal algorithm for unbounded searching, Inform. Process. Lett. 5 (1976) 82–87.
[17] E.R. Berlekamp, Block coding for the binary symmetric channel with noiseless, delayless feedback, in: H.B. Mann (Ed.), Error-Correcting Codes, Wiley, New York, 1968, pp. 61–85.
[18] R.S. Borgstrom, S.R. Kosaraju, Comparison based search in the presence of errors, Proc. 25th Annu. ACM Symp. on Theory of Computing, 1993, pp. 130–136.
[19] F. Cicalese, *Q*-ary searching with lies, in: U. Vaccaro et al. (Eds.), Proc. 6th Italian Conf. on Theoretical Computer Science, Prato, Italy, World Scientific, Singapore, 1998, pp. 228–240.
[20] F. Cicalese, Reliable computation with unreliable information, Ph.D. Thesis, University of Salerno, 2001.

[21] F. Cicalese, D. Mundici, Optimal binary search with two unreliable tests and minimum adaptiveness, Proc. 7th Annu. European Symp. on Algorithms, Lecture Notes in Computer Science, Vol. 1643, Berlin, Springer, 1999, pp. 257–266.
[22] F. Cicalese, D. Mundici, Optimal coding with one asymmetric error: below the sphere packing bound, Proc. Annu. Int. Conf. on Computing and Combinatorics (COCOON 2000), Sydney, Australia, Lecture Notes in Computer Science, Vol. 1858, Springer, Berlin, July 2000, pp. 159–169.
[23] F. Cicalese, D. Mundici, Perfect two-fault tolerant search with minimum adaptiveness, Adv. Appl. Math. 25 (2000) 65–101.
[24] F. Cicalese, D. Mundici, U. Vaccaro, Least adaptive optimal search with unreliable tests, Proc. Scand. Workshop on Algorithm Theory (SWAT 2000), Lecture Notes in Computer Science, Vol. 1851, Springer, Berlin, 2000, pp. 547–562.
[25] F. Cicalese, D. Mundici, U. Vaccaro, Rota-Metropolis cubic logic and Ulam–Renyi games, in: D. Senato, H. Crapo (Eds.), Algebraic Combinatorics and Computer Science—A Tribute to Giancarlo Rota, Springer, Italia, Milano, 2001, pp. 197–244.
[26] F. Cicalese, U. Vaccaro, Optimal strategies against a liar, Theoret. Comput. Sci. 230 (1999) 167–193.
[27] F. Cicalese, U. Vaccaro, An improved heuristic for the "Ulam–Renyi game", Inform. Process. Lett. 73 (2000) 119–124.
[28] F. Cicalese, U. Vaccaro, Coping with delays and time-outs in binary search procedures, Proc. 11th Annu. Internat. Symp. on Algorithms and Computation (ISAAC 2000), Lecture Notes in Computer Science, Vol. 1969, Springer, Berlin, 2000, pp. 96–107.
[29] J. Czyzowicz, K.B. Lakshmanan, A. Pelc, Searching with a forbidden lie pattern in responses, Inform. Process. Lett. 37 (1991) 127–132.
[30] J. Czyzowicz, K.B. Lakshmanan, A. Pelc, Searching with local constraints on error patterns, European J. Combin. 15 (1994) 217–222.
[31] J. Czyzowicz, D. Mundici, A. Pelc, Solution of Ulam's problem on binary search with two lies, J. Combin. Theory Ser. A 49 (1988) 384–388.
[32] J. Czyzowicz, D. Mundici, A. Pelc, Ulam's searching game with lies, J. Combin. Theory Ser. A 52 (1989) 62–76.
[33] A. De Bonis, L. Gargano, U. Vaccaro, Group testing with unreliable tests, Inform. Sci. 96 (1997) 1–14.
[34] P. Denejko, K. Diks, A. Pelc, M. Piotrow, Reliable minimum finding comparator networks, Proc. 19th Internat. Symp. on Mathematical Foundations of Computer Science (MFCS-94), Kosice, Slovakia, Lecture Notes in Computer Science, Vol. 841, Springer, Berlin, August 1994, pp. 306–315.
[35] C. Deppe, Solution of Ulam's searching game with three lies or an optimal adaptive strategy for binary three-error-correcting codes, Discrete Math. 224 (2000) 79–98.
[36] C. Deppe, Coding with feedback and searching with lies, in preparation.
[37] D. DesJardins, Ph.D. Thesis, University of California, Berkeley, in preparation.
[38] A. Dhagat, P. Gacs. P. Winkler, On playing "Twenty Questions" with a liar, Proc. 3rd Annu. ACM-SIAM Symp. on Discrete Algorithms, 1992, pp. 16–22.
[39] R.L. Dobrushin, Information transmission in a channel with feedback, Theory Probab. Appl. 34 (1958) 367–383.
[40] D.-Z. Du, F.K. Hwang, Combinatorial Group Testing and its Applications, World Scientific, Singapore, 1993.
[41] U. Feige, D. Peleg, P. Raghavan, E. Upfal, Computing with noisy information, SIAM J. Comput. 23 (1994) 1001–1018.
[42] B. Gerla, Conditioning a state by a Lukasiewicz event: a probabilistic approach to Ulam games, Theoret. Comput. Sci. 230 (2000) 149–166.
[43] M.J.E. Golay, Notes on digital coding, Proc. IEEE 37 (1949) 657.
[44] W. Guzicki, Ulam's searching game with two lies, J. Combin. Theory Ser. A 54 (1990) 1–19.
[45] R.W. Hamming, Error detecting and error correcting codes, Bell Systems Tech. J. 29 (1950) 147–160.
[46] N. Hanusse, E. Kranakis, D. Krizanc, Searching with mobile agents in networks with liars, Proc. Europar 2000, Lecture Notes in Computer Science, Vol. 1900, Springer, Berlin, 2000, pp. 583–590.
[47] R. Hill, Searching with lies, in: P. Rowlinson (Ed.), Surveys in Combinatorics 1995, Cambridge University Press, Cambridge, 1995, pp. 41–70.
[48] R. Hill, J.P. Karim, Searching with lies: the Ulam problem, Discrete Math. 106/107 (1992) 273–283.

[49] R. Hill, J. Karim, E. Berlekamp, The solution of a problem of Ulam on searching with lies, Proc. IEEE Internat. Symp. on Information Theory (ISIT'98), MIT, August 1998.
[50] D. Innes, Searching with a lie using only comparison questions, preprint.
[51] A. Kaporis, L.M. Kirousis, E. Kranakis, D. Krizanc, Y. Stamatiou, E. Stavropulos, Locating information with uncertainty in fully interconnected networks with applications to World Wide Web retrieval, Computer J., in preparation.
[52] G.O.H. Katona, Search with small sets in presence of a liar, in preparation.
[53] C. Kenyon, V. King, On boolean decision trees with faulty nodes, Random Struct. Algorithms 5 (1994) 453–464.
[54] C. Kenyon, A. Yao, On evaluating Boolean functions with unreliable tests, Internat. J. Foundations Comput. Sci. 1 (1990) 1–10.
[55] L.M. Kirousis, E. Kranakis, D. Krizanc, Y.C. Stamatiou, Locating information with uncertainty in fully interconnected networks, Proc. 14th Internat. Conf. on Distributed Computing (DISC 2000), Toledo Spain, Lecture Notes in Computer Science, Vol. 1914, Springer, Berlin, October 2000, pp. 283–296.
[56] D.M. Kline, A. Mehrez, G.E. Steinberg, Information availability vs. solution space size in a knowledge acquisition problem, Comput. Oper. Res. 26 (1999) 281–292.
[57] E. Kranakis, D. Krizanc, Searching with uncertainty, Proc. 6th Internat. Colloq. on Structural Information and Communication Complexity, SIROCCO'99, Lacanau-Ocean, France, July 1999, pp. 194–203.
[58] M. Kummer, F. Stephan, effective search problems, Mathematical Logic Quart. 40 (1994) 224–236.
[59] E. Kushilevitz, Y. Mansour, Computation in noisy radio networks, Proc. 9th Annu. ACM-SIAM Symp. on Discrete Algorithms (SODA'98), San Francisco, USA, January 1998, pp. 236–243.
[60] K.B. Lakshmanan, B. Ravikumar, K. Ganesan, Coping with erroneous information while sorting, IEEE Trans. Comput. 40 (1991) 1081–1084.
[61] E.L. Lawler, S. Sarkissian, An algorithm for "Ulam's Game" and its application to error correcting codes, Inform. Process. Lett. 56 (1995) 89–93.
[62] S. Lee, K.G. Shin, Probabilistic diagnosis of multiprocessor systems, ACM Comput. Surveys 26 (1994) 121–139.
[63] T. Leighton, Y. Ma, Breaking the $\Theta(n \log^2 n)$ barrier for sorting with faults, Proc. 34th IEEE Annu. Symp. on Foundations of Computer Science, 1993, pp. 734–743.
[64] P.M. Long, Sorting and searching with a faulty comparison oracle, Technical Report UCSC-CRL-92–15, University of California at Santa Cruz, November 1992.
[65] N.A. Lynch, Distributed Algorithms, Morgan Kaufmann, Los Altos, CA, 1996.
[66] A.J. Macula, A nonadaptive version of Ulam's problem with one lie, J. Statist. Plann. Inference 61 (1997) 175–180.
[67] F.J. MacWilliams, N.J.A. Sloane, The Theory of Error-Correcting Codes, Elsevier Science, New York, 1977.
[68] A. Malinowski, K-ary searching with a lie, Ars Combin. 37 (1994) 301–308.
[69] G.M. Masson, D.M. Blough, G.F. Sullivan, System diagnosis, in: D.K. Pradhan (Ed.), Fault-Tolerant Computer System Design, Prentice-Hall, Englewood Cliffs, NJ, 1996, pp. 478–536.
[70] A. Mathur, E.M. Reingold, Generalized Kraft's inequality and discrete k-modal search, SIAM J. Comput. 25 (1996) 420–447.
[71] D. Mundici, Algebras of Ulam's games with lies, Atti del Congresso Nuovi problemi della logica e della filosofia della scienza Viareggio, Italy, 1990, pp. 151–162.
[72] D. Mundici, The complexity of adaptive error-correcting codes, Proc. 4th Workshop on Computer Science Logic (CSL'90), Lecture Notes in Computer Science, Vol. 533, Springer, Berlin, October 1990, pp. 300–307.
[73] D. Mundici, The logic of Ulam's game with lies, in: C. Bicchieri, M.L. Dalla Chiara (Eds.), Knowledge, Belief and Strategic Interaction, Cambridge University Press, Cambridge, 1992, pp. 275–284.
[74] D. Mundici, Logic and algebra in Ulam's searching game with lies, in: Yearbook 1991 of the Kurt Godel Society, Wien, 1993, pp. 109–114.
[75] D. Mundici, Ulam games, Lukasiewicz logic, and AF C^*-algebras, Fund. Inform. 18 (1993) 151–161.
[76] D. Mundici, Logic of infinite quantum systems, Internat. J. Theoret. Phys. 32 (1993) 1941–1955.
[77] D. Mundici, Lukasiewicz normal forms and toric desingularizations, in: W. Hodges, M. Hyland, C. Steinhorn, J. Truss (Eds.), Logic: from Foundations to Applications, Clarendon Press, Oxford, 1996, pp. 401–423.

[78] D. Mundici, Ulam game, the logic of Maxsat, and many-valued partitions, in: D. Dubois, H. Prade, E.P. Klement (Eds.), Fuzzy Sets, Logics and Reasoning about Knowledge, Kluwer Academic Publishers, Dordrecht, 1999, pp. 121–137.
[79] D. Mundici, Fault-tolerance and Rota-Metropolis cubic logic, in: W. Carnielli, M. Coniglio, I.M.L. D'Ottaviano (Eds.), Proc. 2nd World Congr. on Paraconsistency, Juquey, SP, Brazil, May 2000, Marcel Dekker, New York, 2001.
[80] D. Mundici, G. Panti, Twenty questions with many-valued answers, in: X. Caicedo, C. H. Montenegro (Eds.), Models, Algebras and Proofs, Marcel Dekker, New York, 1999, pp. 295–313.
[81] D. Mundici, A. Trombetta, Optimal comparison strategies in Ulam's searching game with two errors, Theoret. Comput. Sci. 182 (1997) 217–232.
[82] S. Muthukrishnan, On optimal strategies for searching in the presence of errors, Proc. 5th Annu. ACM-SIAM Symp. on Discrete Algorithms, 1994, pp. 680–688.
[83] A. Negro, G. Parlati, P. Ritrovato, Optimal adaptive search: reliable and unreliable models, Proc. 5th Italian Conf. on Theoretical Computer Science, 1995, World Scientific, Singapore, pp. 211–231.
[84] A. Negro, M. Sereno, Solution of Ulam's problem on binary search with three lies, J. Combin. Theory Ser. A 59 (1992) 149–154.
[85] A. Negro, M. Sereno, Ulam's searching game with three lies, Adv. Appl. Math. 13 (1992) 404–428.
[86] I. Niven, Coding theory applied to a problem of Ulam, Math. Mag. 61 (1988) 275–281.
[87] A. Pedrotti, Searching with a constant rate of malicious lies, Proc. Internat. Conf. Fun with Algorithms, Carleton Scientific, June 1999, pp. 137–147.
[88] A. Pedrotti, Playing by searching: two strategies against a linearly bounded liar, Theoret. Computer Science, in preparation.
[89] A. Pelc, Lie patterns in search procedures, Theoret. Comput. Sci. 47 (1986) 61–69.
[90] A. Pelc, Coding with bounded error fraction, Ars Combin. 24 (1987) 17–22.
[91] A. Pelc, Solution of Ulam's problem on searching with a lie, J. Combin. Theory Ser. A 44 (1987) 129–140.
[92] A. Pelc, Prefix search with a lie, J. Combin. Theory Ser. A 48 (1988) 165–173.
[93] A. Pelc, Weakly adaptive comparison searching, Theoret. Comput. Sci. 66 (1989) 105–111.
[94] A. Pelc, Detecting a counterfeit coin with unreliable weighings, Ars Combin. 27 (1989) 181–192.
[95] A. Pelc, Detecting errors in searching games, J. Combin. Theory Ser. A 51 (1989) 43–54.
[96] A. Pelc, Searching with known error probability, Theoret. Comput. Sci. 63 (1989) 185–202.
[97] A. Pelc, Searching with permanently faulty tests, Ars Combin. 38 (1994) 65–76.
[98] A. Pelc, Fault-tolerant broadcasting and gossiping in communication networks, Networks 28 (1996) 143–156.
[99] J.R. Pierce, Optical channels: Practical limits with foton counting, IEEE Trans. Comm. COM-26 (1978) 1819–1821.
[100] M. Piotrów, Depth optimal sorting networks resistant to k passive faults, Proc. 7th ACM-SIAM Symp. on Discrete Algorithms, 1966, pp. 242–251.
[101] F. Preparata, G. Metze, R. Chien, On the connection assignment problem of diagnosable systems, IEEE Trans. Electron. Comput. 16 (1967) 848–854.
[102] B. Ravikumar, K. Ganesan, K.B. Lakshmanan, On selecting the largest element in spite of erroneous information, Proc. 4th Annu. Symp. on Theoretical Aspects of Computer Science, Lecture Notes in Computer Science, Vol. 247, Springer, Berlin, 19–21 February 1987, pp. 88–99.
[103] B. Ravikumar, K.B. Lakshmanan, Coping with known patterns of lies in a search game, Theoret. Comput. Sci. 33 (1984) 85–94.
[104] R. Reischuk, B. Schmeltz, Reliable computation with noisy circuits and decision trees—a general $n \log n$ lower bound, Proc. 32nd IEEE Symp. on Foundations of Computer Science, pp. 602–611.
[105] A. Rényi, On a problem of information theory, MTA Mat. Kut. Int. Kozl. 6B (1961) 505–516.
[106] A. Rényi, Napl'o az információelméletről, Gondolat, Budapest, 1976. (English translation: A Diary on Information Theory, Wiley, New York, 1984.)
[107] R.L. Rivest, A.R. Meyer, D.J. Kleitman, K. Winklmann, J. Spencer, Coping with errors in binary search procedures, J. Comput. System Sci. 20 (1980) 396–404.
[108] J.P.M. Schalkwijk, A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback, IEEE Trans. Inform. Theory 17 (1971) 283–287.
[109] M. Sereno, Binary search with errors and variable cost queries, Inform. Process. Lett. 68 (1998) 261–270.

[110] C.E. Shannon, A mathematical theory of communication, Bell Systems Tech. J. 27 (1948) 379–423 and 623–656.

[111] C.E. Shannon, Zero-capacity of noisy channels, IEEE Trans. Inform. Theory IT2 (1956) 8–19.

[112] J. Spencer, Guess a number—with lying, Math. Mag. 57 (1984) 105–108.

[113] J. Spencer, Ulam's searching game with a fixed number of lies, Theoret. Comput. Sci. 95 (1992) 307–321.

[114] J. Spencer, P. Winkler, Three thresholds for a liar, Combin. Probab. Comput. 1 (1992) 81–93.

[115] G. Stachowiak, Proc. Scand. Workshop on Algorithm Theory, SWAT 2000, Lecture Notes in Computer Science, Vol. 1851, Springer, Berlin, 2000, pp. 535–548.

[116] G. Tel, Introduction to Distributed Algorithms, Cambridge University Press, Cambridge, 1994.

[117] S.M. Ulam, Adventures of a Mathematician, Scribner, New York, 1976, p. 281.

[118] J. Von Neumann, Probabilistic logic and the synthesis of reliable organisms from unreliable components, in: C. Shannon, J. McCarty (Eds.), Automata Studies, Princeton University Press, Princeton, NJ, 1956, pp. 43–98.

[119] A.M. Yaglom, I.M. Yaglom, Probability and Information, Reidel, Dordrecht, 1983.

[120] A.C. Yao, F.F. Yao, On fault-tolerant networks for sorting, SIAM J. Comput. 14 (1985) 120–128.