

Chapter 2

Fault-Tolerant Search à la Ulam-Rényi

Are you sitting comfortably? Then I'll begin.

J. S. Lang, Listen with Mother

2.1 Introduction

The problem of efficient search for an unknown element in a finite set S can be reformulated as a game between two players—one deciding the questions to be asked, and the other deciding the answering strategy that makes as hard as possible the first player's task.

In his autobiography *Adventures of a Mathematician*,¹ Stanisław Ulam raised the following question:

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? One clearly needs more than n questions for guessing one of the 2^n objects because one does not know when the lie was told. This problem is not solved in general.

The very same problem was also considered by Alfréd Rényi in his half-fictitious book *A Diary on Information Theory*²

... I made up the following version, which I called “Bar-kochba with lies”. Assume that the number of questions which can be asked to figure out the “something” being thought of is fixed and the one who answers is allowed to lie a certain number of times. The questioner,

¹p. 281, Scribner's, New York (1976).

²Gondolat, Budapest (1976).

of course, doesn't know which answer is true and which is not. Moreover the one answering is not required to lie as many times as is allowed.

For example, when only two things can be thought of and only one lie is allowed, then 3 questions are needed ... If there are four things to choose from and one lie is allowed, then five questions are needed. If two or more lies are allowed, then the calculation of the minimum number of questions is quite complicated ... It does seem to be a very profound problem ...

In the model depicted by Ulam and Rényi, which we shall call the Ulam-Rényi problem, it is also assumed that the player who gives the answers is not fully reliable, or, more pictorially speaking, she is a liar.

Following a consolidated tradition in the area, we shall call the two players Paul (the Questioner) and Carole (the Responder).³ Our basic problem of fault-tolerant search is then formulated as follows.

2.1.1 The Binary Ulam-Rényi Game

Carole and Paul fix a finite set $S = \{0, 1, \dots, M - 1\}$, called the *search space*, and an integer $e \geq 0$; Carole chooses a number x in S and Paul must guess x by asking questions of the form “does x belong to T ?” where T is an arbitrary subset of S . Carole's only possible answers are *yes* or *no*. Then what is the minimum number $N(M, e)$ of questions that Paul has to ask in order to infallibly determine the number x , assuming that Carole can lie at most e times?

For the case $e = 0$, classical binary search yields $N(M, 0) = \lceil \log_2 M \rceil$, the smallest integer not smaller than $\log_2 M$. When lies are allowed to Carole, the situation is slightly more complicated. Suppose Paul's first question is T . In the classical case $e = 0$, if Carole's answer is “*yes*” (resp. “*no*”), Paul will discard all of $S \setminus T$ (resp. T) and reduce his search space from S to T (resp. $S \setminus T$). In contrast, if $e > 0$, then Paul's strategy must be more flexible. In particular, a number can be discarded by Paul if and only if it falsifies more than e of Carole's answers. All the numbers which are not consistent with e or less of Carole's answers are still possible solutions, because Carole could have chosen one of them and decided to lie as much as the rule of the game allows her.

Thus, when $e > 0$ and Paul's question is T , Carole's answer has the following effect on Paul's *state of knowledge*, i.e., on Paul's subsequent assumptions on the set of possible solutions.

- Carole's answer is “*yes*”. Then Paul's search shall continue not only over those elements of the set T which falsify up to e answers, but also over those elements in the complementary set $S \setminus T$ which happened to falsify up to $e - 1$ answers

³See notes at the end of this chapter for some explanation about these names.

before the last question T was asked (since now they still falsify up to e of the answers so far).

- Carole's answer is "no". Then the search shall continue over those elements of the set $S \setminus T$ which falsify up to e answers, as well as over those elements of T which falsified up to $e - 1$ answers before the last question was asked (and now falsify up to e of the given answers).

It is clear that for any number $x \in S$, Paul has to take note of the number of Carole's answers falsified by x , until x happens to falsify more than e answers, and Paul can safely assume that it is not Carole's secret number.

Assume t questions have been answered. For each $j = 0, \dots, e$, let L_j^t denote the set of elements of S falsifying exactly j answers. Thus, before the first question is asked, we can write $L_0^0 = \{0, 1, \dots, M - 1\}$, $L_1^0 = \dots = L_e^0 = \emptyset$. Let T denote the $(t + 1)$ th question. Suppose the answer is "yes" (resp., "no"). Then we can write for each $j = 1, \dots, e$,

$$\begin{cases} L_0^{t+1} = L_0^t \cap T & (\text{resp., } L_0^t \setminus T); \\ L_j^{t+1} = (L_j^t \cap T) \cup (L_{j-1}^t \setminus T) & (\text{resp., } (L_j^t \setminus T) \cup (L_{j-1}^t \cap T)). \end{cases} \quad (2.1)$$

At any stage t of the game, we say that $(L_0^t, L_1^t, \dots, L_e^t)$ is Paul's *state* (of knowledge). Let $x_i = |L_i^t|$ for each $i = 0, 1, \dots, e$. Then the state $\sigma = (L_0^t, L_1^t, \dots, L_e^t)$ is said to be of *type* (x_0, x_1, \dots, x_e) .

We shall be mainly concerned with the problem of minimizing the *number* of questions rather than explicitly formulating these questions as subsets of S .⁴ Then, we can focus on the cardinalities x_i rather than on the sets L_i^t , and by abusing terminology, we call *state* also the $(e + 1)$ -tuple of integers (x_0, \dots, x_e) .

Definition 2.1. A *final state* is a state (x_0, x_1, \dots, x_e) such that $\sum_{j=0}^e x_j \leq 1$.

Final states correspond to ending game conditions. Indeed, $\sum_{j=0}^e x_j = 1$ means that only one number in S is consistent with Carole's answers (but for at most e lies), so it must be the secret number. On the other hand, the condition $\sum_{j=0}^e x_j = 0$ means that no number in S is consistent with Carole's answers, even when assuming that up to e of these answers are lies. In other words, this means that Carole has not been following the rule of the game⁵ and now Paul can realize it, so the game ends.

In the setting where sets are replaced by their cardinalities, a *question* T is completely specified once we know the number t_j of elements in L_j^t quoted by T . Thus a question T shall be denoted by $[t_0, \dots, t_e]$, where $t_j = |T \cap L_j^t|$.

Suppose that Paul's state is (x_0, \dots, x_e) , and the question $\delta = [a_0, \dots, a_e]$ is asked. If Carole's answer is "yes" (resp., "no"), then the resulting state (x'_0, \dots, x'_e) is given by

⁴We shall consider later this issue and the problem it raises in terms of the complexity of representing the strategies.

⁵More precisely, either she did not choose any number and answered randomly in order to fool Paul, or she chose a number but she lied more than e times.

$$\begin{cases} x'_0 = a_0 & (\text{resp., } x'_0 = x_0 - a_0) \\ x'_j = a_j + (x_{j-1} - a_{j-1}) & (\text{resp., } x'_j = x_j - a_j + a_{j-1}) \end{cases} \quad j = 1, \dots, e. \quad (2.2)$$

Given a state $\sigma = (x_0, \dots, x_e)$ and a question δ , the two possible answers to δ determine two more informative states σ^{yes} and σ^{no} . Paul will then adaptively ask the next question and, depending on Carole's answers, he will be left in one of the four possible states $\sigma^{yes,yes}, \sigma^{yes,no}, \sigma^{no,yes}, \sigma^{no,no}$. Proceeding inductively, Paul can build a labelled binary tree \mathcal{T} , rooted at σ , as follows: Any node v is mapped to a question T_v . The two edges stemming from v are labelled *yes* and *no* (the possible answers given by Carole). The nodes which these edges are incident to are labelled by the states resulting from the corresponding answer of Carole to T_v . We say that \mathcal{T} is Paul's *strategy*. We say that the state σ has a *winning strategy of size t* if there exists a binary tree \mathcal{T} of height t , rooted at σ , whose leaves are final states.

Definition 2.2. A *winning n -state* is a state $\sigma = (x_0, x_1, \dots, x_e)$ such that there exists a winning strategy of size n for it. We say that (x_0, x_1, \dots, x_e) is a *borderline winning n -state* if it is a winning n -state but not a winning $(n - 1)$ -state.

Definition 2.3. Let $\sigma = (x_0, \dots, x_e)$ be a state such that x_i is even for each $i = 0, \dots, e$. Then, the question $\delta = [\frac{x_0}{2}, \frac{x_1}{2}, \dots, \frac{x_e}{2}]$ is called an *even splitting question* for σ .

If σ has also odd components, by abuse of terminology, the $(e + 1)$ -tuple of rationals, $\delta = [\frac{x_0}{2}, \frac{x_1}{2}, \dots, \frac{x_e}{2}]$, shall be called a *pseudo even splitting question*. Note that in this case the two n -tuples of rationals resulting from Carole's answer to δ via (2.2) need no longer be states, because their components may be non-integral. However, the dynamic laws given by (2.2) can still be applied without problems.

Definition 2.4. Let $\sigma = (x_0, \dots, x_e)$ and $\sigma' = (y_0, \dots, y_e)$ be two states such that $\sum_{j=0}^k x_j \geq \sum_{j=0}^k y_j$ holds for each $k = 0, \dots, e$. Then, we say that σ' is a *substate* of σ , and we write $\sigma' \leq \sigma$.

2.2 The Volume Bound

The following theorem provides a lower bound on the size of winning strategies.

Theorem 2.1 (Volume Bound). *If $\sigma = (\hat{x}_0, \hat{x}_1, \dots, \hat{x}_e)$ is a winning n -state then*

$$\sum_{i=0}^e \hat{x}_i \sum_{j=0}^{e-i} \binom{n}{j} \leq 2^n.$$

Proof. By hypothesis there exists a strategy \mathcal{T} of size n rooted in σ whose leaves are states (x'_0, \dots, x'_e) with $\sum_{j=0}^e x'_j \leq 1$. Let us relax for the moment the requirement that states be n -tuples of integers. Replacing all questions in \mathcal{T}

by *pseudo* even splitting questions, and formally applying the dynamical rules (2.2), we will still get a tree rooted in σ , of height equal to n , whose leaves are vectors (y'_0, \dots, y'_e) (possibly with rational, non-integral components) and such that $\sum_{j=0}^e y'_j \leq 1$. A simple inductive argument works based on the fact that for any state and any question the sum of the components of one of the resulting states is not smaller than the sum of the components of the resulting state when the question asked is an even splitting.

Thus, the question to be asked in state (x_0, \dots, x_e) will be given by $[\frac{x_0}{2}, \dots, \frac{x_e}{2}]$.

Let us use the notation $a = \frac{1}{2}$. By repeated application of (2.2), the i th component $x_i^{(j)}$ of the vector $(x_0^{(j)}, \dots, x_e^{(j)})$ obtained from $(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_e)$ after j questions is given by

$$\begin{cases} x_0^{(j)} = ax_0^{(j-1)}, & x_0^{(0)} = \hat{x}_0 \\ x_i^{(j)} = ax_i^{(j-1)} + (1-a)x_{i-1}^{(j-1)}, & x_i^{(0)} = \hat{x}_i \quad 1 \leq i \leq e. \end{cases} \quad (2.3)$$

All vectors $(x_0^{(n)}, \dots, x_e^{(n)})$ obtained after n *pseudo* even splitting questions have the following property:

$$\sum_{i=0}^e x_i^{(n)} \leq 1. \quad (2.4)$$

For all $i = 0, \dots, e$, let $F_i(t) = \sum_{j \geq 0} x_i^{(j)} t^j$ denote the *generating function* of the sequence $x_i^{(0)}, x_i^{(1)}, x_i^{(2)}, \dots$. Stated otherwise, $F_i(t) = \sum_{j \geq 0} x_i^{(j)} t^j$. From (2.3) we get:

$$\begin{aligned} F_0(t) &= atF_0(t) + \hat{x}_0 \\ F_i(t) &= atF_i(t) + (1-a)tF_{i-1}(t) + \hat{x}_i \quad 1 \leq i \leq e, \end{aligned}$$

or equivalently,

$$\begin{aligned} F_0(t) &= \frac{\hat{x}_0}{1-at} \\ F_i(t) &= (1-at)^{-1} ((1-a)tF_{i-1}(t) + \hat{x}_i) \quad 1 \leq i \leq e. \end{aligned}$$

It follows that

$$F_i(t) = \sum_{j=0}^i (1-a)^j \frac{t^j \hat{x}_{i-j}}{(1-at)^{j+1}} = \sum_{j=0}^i \sum_{n \geq 0} \binom{n}{j} (1-a)^j a^{n-j} \hat{x}_{i-j} t^n \quad (2.5)$$

for all $0 \leq i \leq e$. Condition (2.4) now becomes

$$[t^n] \left(\sum_{i=0}^e F_i(t) \right) \leq 1, \quad (2.6)$$

where $[t^n]f(t)$ denotes the coefficient of the n th power of t in the power series expansion of $f(t)$. Recalling that $a = \frac{1}{2}$, from (2.5) we see that inequality (2.6) can be reformulated as

$$\sum_{i=0}^e \sum_{j=0}^i \binom{n}{j} \hat{x}_{i-j} \leq 2^n.$$

To complete the proof it is sufficient to write

$$\sum_{i=0}^e \sum_{j=0}^i \binom{n}{j} \hat{x}_{i-j} = \sum_{i=0}^e \hat{x}_i \sum_{j=0}^{e-i} \binom{n}{j}.$$

The above theorem motivates the following definition.

Definition 2.5. The n th volume, $V_n(x_0, \dots, x_e)$, of a state (x_0, \dots, x_e) is defined by

$$V_n(x_0, x_1, \dots, x_e) = \sum_{i=0}^e x_i \sum_{j=0}^{e-i} \binom{n}{j}.$$

The n th volume of a state $\sigma = (x_0, \dots, x_e) = (|L_0|, \dots, |L_e|)$ counts the number of answering strategies available to Carole when the state of the game is σ and n questions are still to be asked.

One can conveniently allow Carole to use a malicious answering strategy. Thus, for instance, x need not be chosen once and for all at the beginning of the game, but can be suitably changed—provided that the new choice comply with the number e of allowed lies—so as to make Paul’s task as hard as possible.

For each one of the x_i many elements of L_i , Carole can still lie up to $e - i$ times. If she decides to lie precisely j times ($j = 0, \dots, e - i$), she can still choose where to lie, in $\binom{n}{j}$ ways. One then easily sees that, given the state (x_0, \dots, x_e) , the overall number of ways for Carole to answer coincides with $V_n(x_0, \dots, x_e)$.

Intuitively, the Volume Bound says that a winning strategy of Paul’s must be large enough to accommodate the number of possible answering strategies of Carole’s. It does not restrict the structure of such a winning strategy.

Trivially, if a state $\sigma = (x_0, x_1, \dots, x_e)$ fails to satisfy the condition

$$V_n(x_0, x_1, \dots, x_e) \leq 2^n, \quad (2.7)$$

then as an immediate consequence of Theorem 2.1, σ cannot be a winning n -state. Instead of saying that a state σ satisfies condition (2.7) we shall henceforth say that σ satisfies the Volume Bound for n questions.

Definition 2.6. The *character* of a state $\sigma = (x_0, \dots, x_e)$ is the smallest integer n such that σ satisfies the Volume Bound for n questions; in symbols,

$$\text{ch}(x_0, x_1, \dots, x_e) = \min\{n \mid V_n(x_0, x_1, \dots, x_e) \leq 2^n\}.$$

A strategy \mathcal{S} of size q for a state σ is said to be *perfect* if \mathcal{S} is winning for σ and $q = \text{ch}(\sigma)$.

A perfect strategy \mathcal{S} which uses the least possible number of questions, as given by the Volume Bound, is an *optimal* winning strategy, in the sense that it cannot be superseded by a shorter winning strategy. On the other hand, we will see several non-perfect optimal winning strategies.

For all integers $M \geq 0$ and $e \geq 0$, let us define

$$N_{\min}(M, e) = \min \left\{ n \mid M \sum_{j=0}^e \binom{n}{j} \leq 2^n \right\}. \quad (2.8)$$

Then by Theorem 2.1 we immediately have the following lower bound on the size of the shortest winning strategy for the Ulam-Rényi game with e lies over a search space of cardinality M :

$$N(M, e) \geq N_{\min}(M, e).$$

Lemma 2.1. Let σ and σ' be two states, with σ a substate of σ' . Then the following conditions hold:

- (a) for all integers $i \geq 0$, $V_i(\sigma) \leq V_i(\sigma')$;
- (b) $\text{ch}(\sigma) \leq \text{ch}(\sigma')$;
- (c) if σ' is a winning k -state then so is σ .

Proof. Conditions (a) and (b) are immediate consequences of the definitions.

Condition (c) follows from a simple restriction of the winning strategy for σ' to the state σ . In order to simplify the proof we resort to thinking of states as vectors of subsets of the search space. Modulo some renaming of the elements, we can assume that the two states $\sigma = (L_0, \dots, L_e)$ and $\sigma' = (L'_0, \dots, L'_e)$ satisfy $\bigcup_{j=0}^k L_j \subseteq \bigcup_{j=0}^k L'_j$ for each $k = 0, \dots, e$. Then, the winning strategy of size k for σ' is also a winning strategy of size k for σ . In fact, in the dynamics induced by a sequence of questions and answers an element x moves rightward through the components of the state σ at the same pace as it does in σ' . Therefore, if as a result of a sequence of questions and answers σ' is emptied of all but at most one element, so is σ .

The proof of the following *conservation law* amounts to a straightforward verification:

Theorem 2.2. *For any state $\sigma = (x_0, \dots, x_e)$ and question δ , let us denote by σ^{yes} and σ^{no} the two states resulting from σ after Carole's answer to δ . Then for all integers $n \geq 0$ we have*

$$V_{n-1}(\sigma^{yes}) + V_{n-1}(\sigma^{no}) = V_n(\sigma).$$

Corollary 2.1. *Suppose $\sigma' = (y_0, \dots, y_e)$ is the state resulting from $\sigma = (x_0, \dots, x_e)$ after an even splitting question. It follows that*

- (a) *If σ satisfies the Volume Bound for n questions, then σ' satisfies the Volume Bound for $n - 1$ questions.*
- (b) $\text{ch}(\sigma') = \text{ch}(\sigma) - 1$.

Proof. From Theorem 2.2 we get $2V_{n-1}(y_0, \dots, y_e) = V_n(x_0, \dots, x_e)$. By hypothesis, $V_n(x_0, \dots, x_e) \leq q^n$, whence $V_{n-1}(y_0, \dots, y_e) = \frac{1}{2}V_n(x_0, \dots, x_e) \leq 2^{n-1}$, which settles (a). Condition (b) immediately follows from (a), by definition of character.

For later purposes we record the following easy result.

Lemma 2.2. *Let $\sigma = (A_0, A_1, \dots, A_e)$, with $\bigcup_{j=0}^e A_j = \{x, y\}$. Let i, j be such that $x \in A_i$ and $y \in A_j$. Then σ is a borderline winning $(2e - (i + j) + 1)$ -state.*

Proof. Let T be the question “Is x_{Carole} equal to x ?”. Let Paul ask $2e - (i + j) + 1$ times the question T . According to whether Carole gives $e - j + 1$ positive answers or $e - i + 1$ negative answers, Paul can safely conclude that the secret number is x or y respectively. For otherwise, Carole has lied more than she was allowed to.

We now prove that, conversely, $2e - (i + j) + 1$ questions are necessary for Paul to always find the secret number. Indeed, we have

$$\begin{aligned} V_{2e-i-j}(\sigma) &= \sum_{k=0}^{e-i} \binom{2e-i-j}{k} + \sum_{k=0}^{e-j} \binom{2e-i-j}{k} \\ &= \sum_{k=0}^{e-i} \binom{2e-i-j}{k} + \sum_{k=e-i}^{2e-i-j} \binom{2e-i-j}{k} \\ &= \sum_{k=0}^{2e-i-j} \binom{2e-i-j}{k} + \binom{2e-i-j}{e-i} > 2^{2e-i-j}. \end{aligned}$$

Then the desired result is a direct consequence of Theorem 2.1.

The following theorem yields a lower bound on the size of the smallest winning strategy in a game with $e + 1$ lies, given the size of the smallest winning strategy in a game with e lies.

Theorem 2.3 (Translation Bound). *Let $\sigma = (x_0, \dots, x_{e-1}, x_e)$ with $\sum_{j=0}^e x_j \geq 3$. If σ is a winning m -state and $\tau = (0, x_0, \dots, x_{e-1})$ is a borderline winning n -state ($n > 0$), then $m \geq n + 3$.*

Proof. The proof is by induction on n .

Induction Base. τ is a winning 1-state. The only possibility is $\tau = (0, 0, \dots, 0, 2)$. Thus $\sigma = (0, 0, \dots, 0, 2, x_e)$, with $x_e \geq 1$. Therefore, $V_3(\sigma) = x_e + 8 \geq 2^3$ for all $x_e \geq 1$, and by Theorem 2.1 σ cannot be a winning 3-state. This settles the case $n = 1$.

Induction Hypothesis. If σ is a winning m' -state and τ is a borderline winning ℓ -state for some $1 \leq \ell \leq n - 1$, then $i \geq \ell + 3$.

Induction Step. Let τ be a borderline n -state with $n > 1$. Suppose (*absurdum hypothesis*) that there exists a winning strategy of size $n + 2$ for σ . Let $\delta = [a_0, a_1, \dots, a_e]$ be the first question in such a strategy. Define the question $\delta' = (0, a_0, \dots, a_{e-1})$.

Denote by $\sigma^{yes} = (x_0^{yes}, \dots, x_e^{yes})$ and $\sigma^{no} = (x_0^{no}, \dots, x_e^{no})$ the two possible states resulting from Carole's answer to the question δ when Paul's state of knowledge is σ . Similarly, denote by τ^{yes} and τ^{no} , the two possible states resulting from Carole's answer to the question δ' when Paul's state of knowledge is τ . It is not hard to verify that $\tau^{yes} = (0, x_0^{yes}, \dots, x_{e-1}^{yes})$ and $\tau^{no} = (0, x_0^{no}, \dots, x_{e-1}^{no})$.

By hypothesis, both σ^{yes} and σ^{no} are winning $(n + 1)$ -states.

Suppose that $\sum_{j=0}^e x_j^{yes} \geq 3$ and $\sum_{j=0}^e x_j^{no} \geq 3$. Then, by induction hypothesis, τ^{yes} and τ^{no} are winning $(n - 2)$ -states, contradicting the hypothesis that no winning strategy of size $n - 1$ exists for τ .

Conversely, suppose that $\sum_{j=0}^e x_j^{yes} \leq 2$. Thus, by hypothesis, we have $\sum_{j=0}^e x_j \geq 3$. Moreover, either $\sum_{j=0}^{e-1} x_j^{yes} \geq 2$ or $\sum_{j=0}^{e-1} x_j^{no} \geq 2$, for otherwise τ^{yes} and τ^{no} are final states contradicting the hypothesis that τ is not a winning 1-state.

Therefore, we get $\sum_{j=0}^e x_j^{no} \geq 3$, which, in turn, implies that $\sigma^{yes} = (0, \dots, 1, 0, \dots, 1)$, i.e., τ^{yes} is a final state. Again, by induction hypothesis, τ^{no} is a winning $(n - 2)$ -state and we reach the contradiction that τ is a winning $(n - 1)$ -state.

The proof is complete.

2.3 Borderline States Satisfying the Volume Bound with Equality

From the previous sections we know that the states satisfying the volume bound with equality are in some sense maximal with respect to the property of being possibly winning. Such states are somehow the most difficult ones. We also know that a substate of a state σ can be solved by using the winning strategy for σ . Suppose now that we have a list of borderline states of “full” volume for which we know the perfect winning strategy. If our list is complete, in the sense that every state that is

Table 2.1 States exactly attaining the volume bound

$\sigma_{i,j}$	$s_{\cdot,1}$	$s_{\cdot,2}$	$s_{\cdot,3}$	$s_{\cdot,4}$	$s_{\cdot,5}$	$s_{\cdot,6}$	$s_{\cdot,7}$	$s_{\cdot,8}$
	2	5	8	11	14	17	20	
$\sigma_{1,\cdot}$	4	8	36	152	644	2728	11556	...
	1	4	7	10	13	16	19	
$\sigma_{2,\cdot}$	2	6	22	94	398	1686	7142	...
	0	3	6	9	12	15	18	
$\sigma_{3,\cdot}$	1	4	14	58	246	1042	4414	...
	0	2	5	8	11	14	17	
$\sigma_{4,\cdot}$	1	1	10	36	152	644	2728	...
	0	0	4	7	10	13	16	
$\sigma_{5,\cdot}$	1	0	5	24	94	398	1686	...
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots

not listed is a substate of one in the list with the same character, then we are done. Our list basically provides us with all the strategies we need for solving an arbitrary instance of the Ulam-Rényi problem.

While the existence of such a “universal” list might be hard to believe, the above idea can be constructively followed to accommodate several interesting and non-finite cases. In particular, we shall show its implementation in the proof of Theorem 2.4.

For any number of lies, Table 2.1 displays an infinite sequence of states admitting a *perfect* winning strategy. Moreover, any such state σ is *maximal*, in the sense that $V_n(\sigma) = 2^n$, where $n = \text{ch}(\sigma)$.

Let $s_{i,j}$ be the (i, j) entry of Table 2.1, occurring in row i and column j . Let $\sigma_{i,j}$ denote the state $(s_{i,1}, s_{i,2}, \dots, s_{i,j})$. Thus $\sigma_{i,j}$ is a state in the game with $j - 1$ lies. To signify that $\sigma_{i,j}$ is a winning n -state we place the integer n above the entry $s_{i,m}$ in the table.

Table 2.1 is constructed as follows: First of all, $s_{i,1} = 1$ and $s_{i,2} = 0$ for all $i = 5, 6, \dots$. For each $i = 1, 2, 3, 4, \dots$, the value $s_{i,1}$ is the largest possible cardinality n of a search space where Paul can successfully search by using $\lceil \log_2 n \rceil$ questions in the game with no lies. The values of $s_{i,2}$ are chosen so as to ensure that the state $(s_{i,1}, s_{i,2})$ is a winning n -state, with $n = \text{ch}(s_{i,1}, s_{i,2})$, and there exists a question δ reducing $\sigma_{i,2}$ to $\sigma_{i+1,2}$ for all $i = 1, 2, 3, 4, \dots$.

The remaining columns of Table 2.1 ($j \geq 3$) are given by the following recurrence:

$$\begin{cases} \text{for } i \geq 3, & s_{i,j} = s_{i-1,j-1} + s_{i-2,j-1} \\ \text{for } i = 2, & s_{2,j} = s_{3,j} + s_{1,j-1} \\ \text{for } i = 1, & s_{1,j} = s_{2,j} + s_{3,j}. \end{cases} \quad (2.9)$$

Lemma 2.3. *With reference to Table 2.1 we have:*

- (a) *For each $i = 1, 2$ and for all $j = 1, 2, \dots$, there exists a question δ for the state $\sigma_{i,j}$ such that the two states resulting from $\sigma_{i,j}$ coincide with $\sigma_{i+1,j}$.*
- (b) *For all $i \geq 3$ and $j = 1, 2, \dots$, there exists a question δ for the state $\sigma_{i,j}$ such that the two resulting states coincide with $\sigma_{i+1,j}$ and $\sigma_{i-2,j-1}$, respectively.*

- (c) For each $j = 1, 2, \dots$, and $i \leq 2j$, the state $\sigma_{i,j}$ is a borderline winning $(3j - i)$ -state.
- (d) For all $i = 1, 2, \dots$, and $j \geq i$, with the exceptions of $s_{1,1}$, $s_{1,2}$, and $s_{2,2}$, the integer $s_{i,j}$ coincides with

$$\left\lfloor 2(2 + \sqrt{5})^j \left(\frac{\sqrt{5} - 1}{2} \right)^{i+2} + \frac{1}{2} \right\rfloor.$$

- (e) For all $i = 1, 2, \dots$, and all integers $j \geq \max\{3, i\}$, the state $\sigma_{i,j}$ satisfies the recurrence law

$$s_{ij} = \left\lfloor s_{i,j-1} (2 + \sqrt{5}) + \frac{1}{2} \right\rfloor.$$

Proof. (d) and (e) are easily proved by using standard techniques for solving recurrences. We now focus on the proof of (a), (b), and (c).

- (a) Let $\sigma = \sigma_{i,j}$ and δ be an even splitting question, i.e., $\delta = [s_{i,1}/2, s_{i,2}/2, \dots, s_{i,j}/2]$. Let σ^{yes} and σ^{no} be the two resulting states. We shall prove that $\sigma^{yes} = \sigma^{no} = \sigma_{i+1,j}$.

Let $\sigma^{yes} = \sigma^{no} = (r_1, \dots, r_j)$. By definition of $s_{i,k}$, we get

$$r_k = s_{i,k}/2 + s_{i,k-1}/2 = \begin{cases} s_{2,k}/2 + s_{3,k}/2 + s_{2,k}/2 - s_{3,k}/2 = s_{2,k} & i = 1, \\ s_{3,k}/2 + s_{1,k-1}/2 + s_{3,k}/2 - s_{1,k-1}/2 = s_{3,k} & i = 2. \end{cases}$$

- (b) Let $\sigma = \sigma_{i,j}$. Let $\delta = [r_1, \dots, r_j]$, be defined by

$$r_k = \begin{cases} 0 & \text{for } k = 1, \\ s_{i-2,k-1} - s_{i,k-1} + r_{k-1} & \text{for } k = 2, \dots, j. \end{cases}$$

Let $\sigma^{yes} = (u_0, u_1, \dots, u_{j-1})$ and $\sigma^{no} = (z_1, z_2, \dots, z_j)$. Thus, recalling (2.2), we have $u_0 = 0$, and for $k = 1, 2, \dots, j - 1$ it holds that $u_k = r_{k+1} + s_{i,k} - r_k = s_{i-2,k} - s_{i,k} + r_k + s_{i,k} - r_k = s_{i-2,k}$. Hence $\sigma^{yes} = (0, s_{i-2,1}, s_{i-2,2}, \dots, s_{i-2,j-1}) = \sigma_{i-2,j-1}$.⁶

On the other hand, we have $z_1 = s_{i,1} - r_1 = 1 = s_{i+1,1}$, and for $k = 2, 3, \dots, j$ it holds that $z_k = s_{i,k} - r_k + r_{k-1} = s_{i,k} - s_{i-2,k-1} + s_{i,k-1} - r_{k-1} + r_{k-1} = s_{i-1,k-1} + s_{i,k-1} = s_{i+1,k}$. Hence $\sigma^{no} = \sigma_{i+1,j}$, as desired.

- (c) By induction on j . The claim is easily true for $j = 1, 2$. Let $j \geq 3$ and $i \leq 2j$. Then by (a) and (b) there exists a question such that the two resulting states are either both equal to $\sigma_{i+1,j}$, or are respectively $\sigma^{yes} = \sigma_{i-2,j-1}$ and $\sigma^{no} =$

⁶Note that any state (s_0, s_1, \dots, s_j) in the game with j lies is the same as the state $(0, s_0, s_1, \dots, s_j)$ in the game with $j + 1$ lies.

Table 2.2 Other states exactly attaining the volume bound

$\tau_{i,j}$	$u_{\cdot,1}$	$u_{\cdot,2}$	$u_{\cdot,3}$	$u_{\cdot,4}$	$u_{\cdot,5}$	$u_{\cdot,6}$
$\tau_{1,\cdot}$	³ 8	⁷ 64	¹¹ 744	¹⁵ 8512	¹⁹ 97416	...
$\tau_{2,\cdot}$	² 4	⁶ 36	¹⁰ 404	¹⁴ 4628	¹⁸ 52964	...
$\tau_{3,\cdot}$	¹ 2	⁵ 20	⁹ 220	¹³ 2516	¹⁷ 28796	...
$\tau_{4,\cdot}$	⁰ 1	⁴ 11	⁸ 120	¹² 1368	¹⁶ 15656	...
$\tau_{5,\cdot}$	⁰ 1	³ 4	⁷ 67	¹¹ 744	¹⁵ 8512	...
$\tau_{6,\cdot}$	⁰ 1	² 1	⁶ 35	¹⁰ 407	¹⁴ 4628	...
$\tau_{7,\cdot}$	⁰ 1	⁰ 0	⁵ 16	⁹ 222	¹³ 2519	...
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots

$\sigma_{i+1,j}$. Suppose that $\sigma^{yes} = \sigma^{no} = \sigma_{i+1,j}$; hence, a fortiori, $i \leq 2$. Then we have the desired result by induction, since by hypothesis $j \geq 3$; hence $i - 1 \leq 2j$. Indeed, $\sigma_{i-1,j}$ is a winning $(3j - i - 1)$ -state and a fortiori we have the desired result for $\sigma_{i,j}$.

Conversely, let $\sigma^{yes} = \sigma_{i-2,j-1}$ and $\sigma^{no} = \sigma_{i+1,j}$. Again, the desired result will follow from both σ^{yes} and σ^{no} being winning $(3j - i - 1)$ -states.

Indeed, σ^{yes} is a winning $(3j - i - 1)$ -state by induction. On the other hand, if $i + 1 \leq 2j$ then again an inductive argument shows that σ^{no} is a winning $(3j - i - 1)$ -state. Finally, if $i + 1 \geq 2j + 1$, then σ^{no} is a final state; hence it is also a winning $(3j - i - 1)$ -state. The proof is complete.

As a matter of fact, Table 2.1 is a special case of the following more general construction. Let $t = 3, 4, 5, \dots$, and for all positive integers i and j define recursively the quantity $a_{i,j}$ as follows:

$$a_{i,1} = 2^{\max\{t-i,0\}}, \quad (2.10)$$

$$a_{i,2} = \max\{0, 2^{2t-i} - 2^{\max\{t-1,0\}}(2t-i+1)\}, \quad (2.11)$$

$$a_{i,j} = \begin{cases} \sum_{k=i+1}^t a_{k,j} + \sum_{k=1}^i i-1 a_{k,j-1} & i = 1, 2, \dots, t-1 \\ \sum_{k=1}^{t-1} a_{i-k,j-1} & i \geq t. \end{cases} \text{ for } j \geq 3. \quad (2.12)$$

The interesting property is that any state $\alpha_{i,j} = (a_{i,1}, \dots, a_{i,j})$ is a winning $(tj - i)$ -state, with $ch(\alpha_{i,j}) = tj - i$, and $V_{(tj-i)}(\alpha_{i,j}) = 2^{(tj-i)}$. In other words, any state in the table has a perfect winning strategy and is *maximal* in the previously defined terms.

Another useful characteristic of such a table is that the increase in the number of necessary and sufficient questions when translating from the state $\alpha_{i,j}$ to $\alpha_{i,j+1}$ is equal to t .

Table 2.1 is the one obtained in the particular case $t = 3$. For later, we also precisely list the table obtained in the case $t = 4$, which coincides with Table 2.2.

2.4 The Solution of the 20 Question Game with Lies

We shall now use the original instance of the problem given by Ulam ($M = 2^{20}$) to show how the tables introduced in the previous section can help analyzing Paul's strategies and constructing optimal once. Recall that $N(M, e)$ denotes the size of the shortest winning strategy for the Ulam-Rényi game over a search space of cardinality M when up to e answers may be mendacious. We shall prove the following.

Theorem 2.4. *The values of $N(2^{20}, e)$ are given by the following table:*

e	0	1	2	3	4	5	6	7	8	9	...	e	...
$N(2^{20}, e)$	20	25	29	33	37	40	43	46	50	53	...	$3e + 26$...

For all $e \geq 8$, we have $N(2^{20}, e) = 3e + 26$.

Proof. Starting from the initial state $\sigma_e^{(0)} = (2^{20}, \underbrace{0, \dots, 0}_{e \text{ zeros}})$, and asking 20 even splitting questions, Paul will be in a sequence of states $\sigma_e^{(1)}, \dots, \sigma_e^{(20)}$, where

$$\sigma_e^{(i)} = \left(2^{20-i}, i2^{20-i}, \binom{i}{2}2^{20-i}, \dots, \binom{i}{j}2^{20-i}, \dots, \binom{i}{e}2^{20-i} \right)$$

for each $i = 1, \dots, 20$. In particular, after these initial 20 questions are answered, and independently of Carole's answers, Paul's state is given by

$$\chi_e = \left(1, 20, \binom{20}{2}, \dots, \binom{20}{j}, \dots, \binom{20}{e} \right). \quad (2.13)$$

Clearly, there exists no even splitting question for Paul in such a state. As we shall see, all he has to do is to reduce the state (2.13) to a substate of some state in Table 2.1.

We shall argue by cases:

Case 1. $e \geq 8$.

Then, by direct inspection, $\text{ch}(\sigma_8^{(0)}) = 50$. By the Translation Bound and Corollary 2.1, for all $e \geq 8$ no winning strategy exists for χ_e using $< 6 + 3e$ questions. Thus we have only to prove that a winning strategy of size $6 + 3e$ exists.

We now use *quasi* even splitting questions in the next six questions. For any state $\sigma = (x_0, x_1, \dots, x_e)$ the question $\delta = [\lceil \frac{x_0}{2} \rceil, \lceil \frac{x_1}{2} \rceil, \dots, \lceil \frac{x_e}{2} \rceil]$ is said to be *quasi-even splitting*. A *quasi-even splitting* corresponds to an even splitting in the particular case when all the components of the state σ are even.

As a result, the 64 states obtained after these six quasi-even splitting questions will be substates of

$$\sigma^\sharp = (1, 1, 5, 41, 233, 1028, 3597, 10278, 24411, 48821, \dots).$$

Let us display the winning $3e$ -state $\sigma_{3,e+1}$ in Table 2.1 as follows:

$$\sigma_{3,e+1} = (1, 4, 14, 58, 246, 1042, 4414, 18698, \dots).$$

In light of Lemma 2.1, it is sufficient to prove that σ^\sharp is a substate of $\sigma_{3,e+1}$ for all $e \geq 8$. For $e = 8$ the claim follows by direct inspection. Proceeding by induction, and letting a_i be the i th component of σ^\sharp , we have $a_i \leq s_{3,i}$ for all $i = 1, 2, \dots, 9$. It is not hard to see that for all $i \geq 10$,

$$a_i < \left\lfloor a_{i-1} (2 + \sqrt{5}) + \frac{1}{2} \right\rfloor.$$

Since each component of σ^\sharp grows at a smaller rate than its corresponding component in $\sigma_{3,e+1}$, we conclude that

$$a_i < \left\lfloor a_{i-1} (2 + \sqrt{5}) + \frac{1}{2} \right\rfloor \leq \left\lfloor s_{i-1} (2 + \sqrt{5}) + \frac{1}{2} \right\rfloor = s_i,$$

thus settling the present case.

Case 2. $e \leq 7$.

The cases $e = 1, e = 2, e = 3$ and $e = 4$ are settled arguing as for Case 1 in the light of Table 2.2. More precisely, it is easily checked that for each $e = 1, 2, 3, 4$, the state χ_e is a substate of $\tau_{3,e+1}$ in Table 2.2, which is a winning $(4e + 1)$ -state. Thus we immediately have the desired result that $\sigma_e^{(0)}$ is a winning $(4e + 21)$ -state. The cases $e = 5, 6, 7$ will be settled using ad hoc strategies, based on ideas described in Sect. 2.6.

The proof is complete.

Remark. Ulam's instance $M = 10^6$ can now be settled without much effort. In the exceptional case $e = 4$, we have $\text{ch}(10^6, 0, 0, 0, 0) = 36$ and $\text{ch}(2^{20}, 0, 0, 0, 0) = 37$. As a matter of fact, there exists a *perfect* strategy for the state $(10^6, 0, 0, 0, 0)$, whence $N(10^6, 4) = 36$. For the remaining values of e one has $N(10^6, e) = N(2^{20}, e)$, because $\text{ch}(10^6, 0, \dots, 0) = \text{ch}(2^{20}, 0, \dots, 0)$ and by Lemma 2.1 any winning strategy for $(2^{20}, 0, \dots, 0)$ trivially yields a winning strategy for $(10^6, 0, \dots, 0)$.

2.5 Asymptotics for the Ulam-Rényi Problem

In the previous section we presented a family of searching strategies for the original Ulam instance. The common idea underlying these strategies is summarized in the following steps:

1. Use even splitting questions for the first 20 questions.
2. Use quasi-even splitting questions until the resulting state of knowledge σ is a substate of some $\alpha_{i,j}$ and $\text{ch}(\alpha_{i,j}) = \text{ch}(\sigma)$.
3. According to Lemma 2.1, transform the perfect strategy for $\alpha_{i,j}$ into a perfect strategy for σ .

The basic ingredient is a question that splits the volume of the current state as evenly as possible. Indeed, this is exactly what happens, when even splitting questions are used. Moreover, the possibility to define questions which exactly split the volume is one of the main features of the states $\alpha_{i,j}$ in Tables 2.1 and 2.2.

It is natural to wonder whether such a strategy exists for all possible states. Unfortunately, this is not the case. Consider, for example, the state $\sigma = (5, 0)$. We have $\text{ch}(\sigma) = 5$, and $V_5(\sigma) = 30 \leq 2^5$. The question that best splits the volume of σ is $[2, 0]$ (or, equivalently the symmetric question $[3, 0]$). Now if Carole answers “no”, the resulting state is $\sigma^{no} = (3, 2)$, and we have $V_2(\sigma^{no}) = 17 > 2^4$, so it is no longer possible to finish the search within four more questions. We conclude that five questions, although necessary, are not sufficient to guess an unknown number in a set of cardinality 5 when one of the answers is a lie.

Unlike the cases considered in the previous section, quasi-even splitting questions are not always effective. So it is generally a hard task to find the best question for a given state. Equivalently, no general rule exists to determine the length of the shortest searching strategy for arbitrary cardinality of the search space and number of lies.

Nonetheless, for any fixed number of lies, e , we can provide asymptotic conditions on the states of knowledge which allow a perfect winning strategy.

Theorem 2.5. *There exist constants K_e and Q_e (depending on e) having the following property: for all integers $n \geq Q_e$, if a state (x_0, \dots, x_e) satisfies $V_n(x_0, \dots, x_e) \leq 2^n$ and $x_e \geq K_e n^e$, then the state is n -winning.*

*Proof (Sketch).*⁷ Let $\sigma = (x_0, x_1, \dots, x_e)$, with $\text{ch}(\sigma) = n$. For $i = 0, 1, \dots, e - 1$, let a_i be chosen as $\lfloor \frac{x_i}{2} \rfloor$ or $\lceil \frac{x_i}{2} \rceil$ on an alternate basis. Then define a_e as the integer minimizing the quantity $\Delta = (2a_e - x_e) + \sum_{j=0}^{e-1} (x_j - 2a_j) \binom{n-1}{e-j}$. Let $\delta_\sigma = [a_0, a_1, \dots, a_e]$.

The standing hypothesis on x_e guarantees the possibility of balanced volume splitting. In fact, because of the large number of elements whose weight in the volume is 1, Paul can use them to cope with the possible

⁷In Chap. 4, we provide a stronger result, of which this is a special case.

unbalance due to the use of quasi-even splitting. Thus, Paul can effectively⁸ make use of the above rule in defining his questions until the resulting state is of the form $\sigma' = (0, \dots, 0, 1, 0, \dots, 0, x_e)$. This is a substate of $(0, \dots, 0, 1, 0, \dots, 0, 2^{\text{ch}(\sigma')} - \sum_{j=0}^e \binom{\text{ch}(\sigma')}{j})$, which is evenly splitted by the question of type $\delta' = [0, \dots, 0, 1, 0, \dots, 0, 2^{\text{ch}(\sigma')-1} - \sum_{j=0}^e \binom{\text{ch}(\sigma')-1}{j}]$.

By recursively applying questions of type δ' , Paul *perfectly* gets through to the end.

This theorem has an immediate consequence on the existence of perfect strategies for specific instances of the game: Let us fix two integers $e, m \geq 0$. Let S be a search space of cardinality $M = 2^m$. Then, up to finitely many exceptional m 's, there exists a *perfect* winning strategy for Paul. Stated differently, Paul can win the game over S with e lies using n questions, with n being the smallest integer satisfying the Volume Bound. Trivially, no such winning strategy can use less than n questions.

In fact, for all sufficiently large m , starting from the initial state $\sigma = (2^m, 0, \dots, 0)$ with $\text{ch}(\sigma) = n$, after the first m even splitting questions, the resulting state is $\sigma' = (1, m, \binom{m}{2}, \dots, \binom{m}{e})$ with $\text{ch}(\sigma') = n - m$, which satisfy the hypothesis of Theorem 2.5, since it can be shown that $\binom{m}{e} \sim m^e \gg (n - m)^e$.

2.6 Heuristics for the Ulam-Rényi Problem

Despite its far-reaching generality, the asymptotic result of Theorem 2.5 does not provide the ultimate solution to the Ulam-Rényi problem. Of practical interest is also the question of generating the winning strategy for any small instance. Optimal algorithms are known to find the exact solution of the Ulam-Rényi problem over an arbitrary search space when the number of allowed lies is small. For the general case of an arbitrary number of lies, only heuristic-based algorithms have been proposed. In this section we shall present two such heuristics.

Algorithm 1

Let $\sigma = (x_0, x_1, \dots, x_e)$, with $\text{ch}(\sigma) = n$. Define $\delta = [a_0, a_1, \dots, a_e]$ by recursively choosing $a_i \in \{0, 1, \dots, x_i\}$ in order to minimize $\left| \sum_{j=0}^i \binom{n-1}{e-j} (x_j - 2a_j) \right|$, i.e.,

$$a_i = \underset{0 \leq a_i \leq x_i}{\operatorname{argmin}} \left| \sum_{j=0}^i \binom{n-1}{e-j} (x_j - 2a_j) \right| \quad i = 0, 1, \dots, e.$$

⁸Here, we mean that on a state σ , Paul's questions δ_σ will result in two states σ^{yes} and σ^{no} , such that $\text{ch}(\sigma^{yes}), \text{ch}(\sigma^{no}) \leq \text{ch}(\sigma) - 1$.

This rule slightly refines the strategy depicted in Theorem 2.5 by a component-wise balancing of the resulting states' volume.

This heuristic has one main drawback: it does not take into account the translation bound. In fact, the second heuristic we present is based on this observation and tries to take into account both the volume and the translation bound.

Algorithm 2

Let $\sigma = (x_0, x_1, \dots, x_e)$. We will denote by $\Theta(\sigma)$ the state $(0, x_0, x_1, \dots, x_{e-1})$. This state is obtained by shifting the components of σ one place to the right. This is also equivalent to *translating* σ into the corresponding state for the game with $e - 1$ lies. For any $i = 1, 2, \dots, e$, we define $\Theta^i(\sigma) = \Theta(\Theta^{i-1}(\sigma))$, with $\Theta^0(\sigma) = \sigma$.

Let the function $\mathcal{G} : \mathbf{N}^{e+1} \mapsto \mathbf{N}$ be recursively defined by

$$\mathcal{G}(x_0, \dots, x_e) = \begin{cases} \text{ch}(x_0, \dots, x_e) & \text{if } \sum_{i=0}^e x_i \leq 2; \\ \max\{\mathcal{G}(0, x_0, \dots, x_{e-1}) + 3, \text{ch}(x_0, \dots, x_e)\} & \text{otherwise.} \end{cases} \quad (2.14)$$

Further, with any state $\sigma = (x_0, \dots, x_e)$ we associate the vector $\Gamma(\sigma) = (\gamma_0, \gamma_1, \dots, \gamma_e)$ defined by

$$\gamma_i = \max\{\mathcal{G}(\Theta^{e-i}(\sigma)), \mathcal{G}(\sigma) - 3(e - i)\} \quad \text{for } i = 0, 1, \dots, e. \quad (2.15)$$

An immediate property of the vector $\Gamma(\sigma)$ is given by the following lemma.

Lemma 2.4. *Let $\sigma = (x_0, x_1, \dots, x_e)$ and $\Gamma(\sigma) = (\gamma_0, \gamma_1, \dots, \gamma_e)$ be defined as in formulae (2.14) and (2.15) above. Let $j = \min\{i \mid \sum_{k=0}^i x_k \geq 3\}$. Then, for any $i \geq j$ we have*

$$\gamma_i = \gamma_e - 3(e - i).$$

Proof. By definition we have $\gamma_e = \mathcal{G}(\sigma)$. Hence, for $i = 0, 1, \dots, e$ we also have $\gamma_i = \max\{\mathcal{G}(\Theta^{e-i}(\sigma)), \gamma_e - 3(e - i)\}$. Then we only need to prove $\gamma_e - 3(e - i) \geq \mathcal{G}(\Theta^{e-i}(\sigma))$, or, equivalently $\gamma_e \geq \mathcal{G}(\Theta^{e-i}(\sigma)) + 3(e - i)$, for all $i \geq j$.

Indeed, by hypothesis we have $\sum_{k=0}^i x_k \geq 3$, for all $i \geq j$. Then (2.14) yields

$$\gamma_e = \mathcal{G}(\sigma) \geq \mathcal{G}(\Theta^1(\sigma)) + 3 \geq \mathcal{G}(\Theta^2(\sigma)) + 6 \geq \dots \geq \mathcal{G}(\Theta^{e-i}(\sigma)) + 3(e - i),$$

which gives the desired result.

Intuitively, γ_e yields a lower bound for the length of the shortest winning strategy for σ , considering both Volume and Translation Bound. In addition, γ_{e-i} stands for an upper bound on the length of the shortest winning strategy for $\Theta^i(\sigma)$ provided that there exists a winning strategy for σ with γ_e questions. This is proved by the following corollary.

Corollary 2.2. Let $\sigma = (x_0, x_1, \dots, x_e)$ be a state. Let $(\gamma_0, \gamma_1, \dots, \gamma_e) = \Gamma(\sigma)$.

- (a) If there exists a winning strategy for σ with q questions, then $q \geq \gamma_e$;
- (b) If $q = \gamma_e$, then for all $i = 1, \dots, e$, there exists a winning strategy for $\Theta^i(\sigma) = (0, \dots, 0, x_0, \dots, x_{e-i})$ with γ_{e-i} questions.

The following lemma states that in most cases the optimal winning strategy for σ (with respect to the lower bound γ_e) implicitly behaves like an optimal winning strategy for the states $\Theta^i(\sigma)$ (with respect to the upper bound γ_{e-i}).

Lemma 2.5. Let $\sigma = (x_0, x_1, \dots, x_e)$ be a state. Let $(\gamma_0, \gamma_1, \dots, \gamma_e) = \Gamma(\sigma)$. Let Σ be a winning strategy for σ using exactly γ_e questions. Moreover, let $\delta = [a_0, a_1, \dots, a_e]$ be the first question in the winning strategy Σ and σ^{yes} and σ^{no} be the resulting states from a positive and a negative answer to δ , respectively. Then, for all $i = 1, \dots, e$ such that $\sum_{j=0}^{e-i} x_j \geq 3$, we have

$$V_{\gamma_{e-i}-1}(\Theta^i(\sigma^{yes})) \leq 2^{\gamma_{e-i}-1} \text{ and } V_{\gamma_{e-i}-1}(\Theta^i(\sigma^{no})) \leq 2^{\gamma_{e-i}-1}.$$

Proof. By $\sum_{j=0}^{e-i} x_j \geq 3$, Lemma 2.4 yields $\gamma_e = \gamma_{e-i} + 3i$. Suppose by contradiction that for some $i \in \{1, \dots, e\}$ it holds that $V_{\gamma_{e-i}-1}(\Theta^i(\sigma^{yes})) > 2^{\gamma_{e-i}-1}$. Then, setting $(\gamma'_0, \gamma'_1, \dots, \gamma'_e) = \Gamma(\sigma^{yes})$, we have $\gamma'_{e-i} \geq \gamma_{e-i}$. Let $\sigma^{yes} = (y_0, \dots, y_e)$ as in (2.2). Thus $\sum_{j=0}^{e-i+1} y_j \geq 3$ and we have $\gamma'_e \geq \gamma'_{e-i} + 3i \geq \gamma_{e-i} + 3i = \gamma_e$. Therefore, at least γ_e questions are necessary to reach a final state from σ^{yes} . Since $\gamma_e - 1$ questions are left in Σ , it cannot be a winning strategy.

Symmetrically, we have the proof for the condition on $\Theta^i(\sigma^{no})$.

The above Lemma 2.5 and Corollary 2.2 formalize the intuition behind our heuristic. By Corollary 2.2, given a state $\sigma = (x_0, \dots, x_e)$ and the corresponding $(e+1)$ -tuple $(\gamma_0, \dots, \gamma_e) = \Gamma(\sigma)$, the first question $\delta = [a_0, \dots, a_e]$ in any optimal winning strategy (i.e., one which uses *exactly* γ_e questions) must satisfy the following property:

for $i = 0, \dots, e$, $\sigma_i^{yes} = (0, \dots, 0, y_0, \dots, y_i)$ and $\sigma_i^{no} = (0, \dots, 0, n_0, \dots, n_i)$, the two states resulting from the state $(0, \dots, 0, x_0, \dots, x_i)$ on question $\delta_i = (0, \dots, 0, a_0, \dots, a_i)$ must be both winning $(\gamma_i - 1)$ -states; y_j and n_j are computed according to (2.2).

Thus Paul would seem to be well advised to define the component a_i for any $i = 0, \dots, e$ such that the quantities $V_{\gamma_i-1}(\sigma_i^{yes})$ and $V_{\gamma_i-1}(\sigma_i^{no})$ are as nearly equal as possible.

Unfortunately, this is not sufficient, since the choice of a_i affects the states σ_{i+j}^{yes} and σ_{i+j}^{no} for $j = 1, \dots, e - i$.

Let us define for any $i = 0, \dots, e - 1$ and $j = 1, \dots, e - i$ the *pseudo states*

$$\alpha_{i,j} = (0, \dots, 0, y_0, \dots, y_i, x_i - a_i, \underbrace{0, \dots, 0}_{j-1}) \quad \beta_{i,j} = (0, \dots, 0, n_0, \dots, n_i, a_i, \underbrace{0, \dots, 0}_{j-1}).$$

Such states record what is already known about the states σ_{i+j}^{yes} and σ_{i+j}^{no} as soon as the component a_i has been chosen but the components a_{i+1}, \dots, a_{i+j} have not been chosen yet.

The strategy also tries to take care of the states σ_{i+j}^{yes} and σ_{i+j}^{no} while deciding the component a_i by requiring that $V_{\gamma_i+j-1}(\alpha_{i,j}) \leq 2^{\gamma_i+j-1}$ and $V_{\gamma_i+j-1}(\beta_{i,j}) \leq 2^{\gamma_i+j-1}$.

This is the best we can do if we decide to choose the components of the question step by step, without allowing backtracking.

We then define a_i as the integer $x \in \{0, 1, \dots, x_i\}$ such that for the question $\delta_i = (0, \dots, 0, a_0, a_1, \dots, a_{i-1}, x)$ the difference

$$|V_{\gamma_i-1}(\sigma_i^{yes}) - V_{\gamma_i-1}(\sigma_i^{no})|$$

is minimum and for any $j = 1, \dots, e - i$ it holds that

$$V_{\gamma_i+j-1}(\alpha_{i,j}) \leq 2^{\gamma_i+j-1} \quad \text{and} \quad V_{\gamma_i+j-1}(\beta_{i,j}) \leq 2^{\gamma_i+j-1}.$$

Such a strategy corresponds to the one which first aims at being a winning strategy for the state $\Theta^{e-i}(\sigma) = (0, \dots, 0, x_0, \dots, x_i)$, for $i = 0, \dots, e - 1$ with γ_i questions.

Notice that this condition need not hold in general, since Lemma 2.5 only constrains the states $\Theta^{e-i}(\sigma)$ that satisfy $\sum_{j=0}^i x_j \geq 3$. However, the heuristic still yields optimal results in many different cases.

2.6.1 Experimental Validation of the Heuristics

Algorithm 2 was experimentally proved to provide optimal strategies for the Ulam-Rényi game over a search space of cardinality 2^m with e lies, for each $m = 1, \dots, 16$ and each $e = 1, \dots, 9$.

The lengths of the corresponding strategies are summarized in Table 2.3. When tested over the same sample of instances, Algorithm 1 does not give *optimal* results for the cases

$$(m, e) \in \{(8, 5), (8, 6), \dots, (8, 9), (11, 6), (11, 7), (11, 8), (11, 9), (14, 7), (14, 8), (14, 9)\}.$$

Nonetheless, it is much simpler to implement, and in all the above cases it provides solutions which differ by at most two questions.

We close this section by mentioning a result obtained by combining the theoretical techniques of the previous sections and the heuristics. It is known that for $e \geq 6$ and a search space of cardinality $M = 2^{14}$ the optimal solution requires exactly $3e + 17$ questions. The proof of this fact relies on the solution of the case $M = 2^{14}$ and $e = 14$ given by Algorithm 2 and the general technique of Sect. 2.4 to extend this result to arbitrary value of $e \geq 6$.

Table 2.3 Optimal solution (minimum number of queries) for $|S| = 2^m$ and e lies

e	m															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	3	5	6	7	9	10	11	12	13	14	15	17	18	19	20	21
2	5	8	9	10	12	13	14	15	17	18	19	20	21	22	24	25
3	7	11	12	13	15	16	17	18	20	21	22	23	25	26	27	28
4	9	14	15	16	18	19	20	21	23	24	25	27	28	29	30	32
5	11	17	18	19	21	22	23	24	26	27	28	30	31	32	34	35
6	13	20	21	22	24	25	26	27	29	30	31	33	34	35	37	38
7	15	23	24	25	27	28	29	30	32	33	34	36	37	38	40	41
8	17	26	27	28	30	31	32	33	35	36	37	39	40	41	43	44
9	19	29	30	31	33	34	35	36	38	39	40	42	43	44	46	47

The exact value of $N(2^m, e)$ for $m = 1, 2, \dots, 16$ and $e = 1, \dots, 9$

2.7 Bibliographic Notes

The specific names Paul and Carole were not randomly chosen. The initials P and C refer to Pusher-Chooser games investigated by Spencer in [196]. Paul may be considered the great questioner Paul Erdős. Carole may be thought of as her anagram: Oracle!

The Volume Bound was first proved by Berlekamp [22]. Subsequently, Rivest et al. [185] re-proved it for the continuous case. Pelc [162] also gave his own proof of this bound. We included here a novel, induction-free proof, originally presented in [58], for the more general case of q -ary search. An alternative proof of Volume Bound can be found in [5].

Different proofs of Theorem 2.3 and Lemma 2.2 can be found in [22], where they first appeared. Berlekamp's n th Volume of a state is also defined as the n th weight, or, more simply, the weight of a state, in most of the later papers on the topic of the Ulam-Rényi problem.

For $e = 1, 2, 3$, the exact value of $N(2^{20}, e)$ was computed in the papers [83, 157, 162]. In the same papers the reader can find the exact value of $N(2^m, e)$ for all integers $m \geq 0$, for the cases $e = 1$, $e = 2$, $e = 3$, respectively. For $e = 1, 2, 3$, evaluation of $N(M, e)$ for all integers $M \geq 1$ can be found in [89, 115, 162], respectively. Hill et al. [119–121] were the first to give complete solutions for the Ulam-Rényi problem over a search space of cardinalities 2^{20} and 10^6 . More precisely, with respect to the presentation given in this chapter, they settled the case $e = 5$ in the light of Berlekamp's Tables (Figs. 9 and 11 of [22]). The use of infinite tables of winning states was started by Berlekamp [22]. As a matter of fact, Tables 2.1 and 2.2 first appeared in [22], together with a more general description also given in this chapter. We have partially deviated from Berlekamp's original presentation, and preferred to give our own construction, principally based on the simplification and correction given in [121] (see also [119, 120]).

For the details on Theorem 2.5, refer to [198]. Algorithm 1 is presented in [138], while Algorithm 2 was originally presented in [52].

2.8 Exercises

1. Show that there is no strategy matching the volume bound for the Ulam-Rényi game with 2 lies over a search space of size 4. In particular, this means that no winning strategy can exist using at most 7 questions.
2. Show that for any variant of the Ulam-Rényi game with e lies there exists a strategy using $(2e + 1)N^*(M)$ questions, where $N^*(M)$ denotes the minimum number of questions sufficient to win the game for Paul when Carole is not allowed to lie.
3. Is the Volume Bound still valid for the variant of the game where Carole is allowed to answer *sincerely* at most e times?
4. Consider the Ulam-Rényi game with one lie. Suppose that, after having chosen the secret number, Carole answers each question insincerely with probability $1/2$, as long as she still has the possibility to lie.

What is the expected number of questions Paul needs as a function of the search space size n . What about the case of $e > 1$ lies?

5. Describe what is in general the best answering strategy for Carole. What is the space and time complexity of such a strategy?
6. In a variant of the Ulam-Rényi game, Carole wants to finish the game as soon as possible and Paul wants to continue as long as possible. What is the minimum length (in terms of number of questions) of such a game, if the number of objects is 4 and Carole is allowed to lie at most twice?
7. Consider the Ulam-Rényi game with 1 lie over a search space of cardinality 10^6 . Assume that the only questions allowed are of the form “Is x in Q ?”, where $|Q| \leq 20$. What is now the minimum number of questions that Paul has to ask in order to identify Carole’s secret number?
8. With reference to the model in the previous exercise, assume now that the search space has cardinality q^m and each question cannot be of cardinality larger than m . Provide upper and lower bounds on the size of a minimum size strategy for Paul as a function of m .
9. Professor Quick has thought of the following alternative heuristics for solving any instance of the Ulam-Rényi game where the search space cardinality is of the form 2^m : Ask m even splitting questions and let σ be the resulting state. Find the most appropriate values of t, i, j , such that the state (a_{i1}, \dots, a_{ij}) from the table defined in (2.10)–(2.12) is the “closest” superstate of σ . Then, use Lemma 2.1 to complete the strategy.

Try to analyze this strategy by providing an upper bound on how many more questions are needed with respect to the character of the original state. Try to estimate the complexity of the resulting algorithm.

Fault-Tolerant Search Algorithms

Reliable Computation with Unreliable Information

Cicalese, F.

2013, XV, 207 p. 14 illus., Hardcover

ISBN: 978-3-642-17326-4