On the i.o. Autoreducibility of Algorithmically Random Sequences

Last Updated March 2nd, 2024

1 Algorithmically Random Sequences

A **bit sequence** is an infinite sequence $b_0b_1b_2\cdots$, where $b_i \in \{0,1\}$ for each $i \geq 0$. Moreover, we wish to consider a bit sequence as existing in the same manner as other abstract and infinite objects exist, such as the number π, and be able to prove properties about them. In particular, we wish to define what is meant by a "random bit sequence" where, intuitively, such a sequence is the result of independently tossing a fair coin (0 for heads, 1 for tails) an infinite number of times.

In what follows we now describe how probability and computability theory come together to formalize the concept of a random bit sequence.

1.1 Probability Measure Space

Using probability theory, mathematicians have been able to prove several interesting facts about random bit sequences. For example, the Law of Large Numbers (LLN) implies that, with probability one,

$$\frac{b_0 + b_1 + \cdots + b_{n-1}}{n} \to \frac{1}{2}$$

for increasingly large values of n. But what is meant by "probability one"? For finite probability distributions this means that an event will occur with certainty, meaning that every time one conducts an experiment and tests for the event's occurrence, the test will always result in positive.

But what about an event, such as a random bit sequence satisfying the Law of Large Numbers? Such an event cannot be observed due to its reliance on an infinite amount of information. In this case "probability one" means that the *measure* of those sequences that do *not* satisfy LLN is equal to zero, meaning that, although it is possible that a sequence does not satisfy LLN, the set of all such

sequences can be placed in an arbitrarily small "cover". These concepts are formalized in what is called a **probability measure space** which is a triple $(\mathcal{S}, \mathcal{E}, \mu)$ consisting of a sample space $\mathcal{S}$, an event space $\mathcal{E}$, and a probability measure μ, all three of which are defined in the following sections.

1.2 Sample space

The first ingredient of a probability measure space is referred to as the **sample space** $\mathcal{S}$ which gives the set of possible outcomes of some experiment.

Example 1.1.

The following are some examples of experiments along with their associated sample spaces.

Tossing a Fair Coin $\mathcal{S} = \{\mathrm{H}, \mathrm{T}\}$

Rolling a Pair of Dice $\mathcal{S} = \{2, 3, \ldots, 12\}$

Observing the Location of an Object in 3D Space $\mathcal{S} = \mathcal{R}^3$

□

Example 1.2. For the experiment of generating a bit sequence by tossing a fair coin an infinite number of times, the sample space is $\mathcal{S} = \{0, 1\}^\infty$. Although a sample can never be fully observed, we nonetheless acknowledge its mathematical existence and the existence of the entire sample space. □

1.3 Event Space

The second ingredient of a probability measure space is referred to as the **event space** $\mathcal{E}$ whose members are subsets of $\mathcal{S}$. We say that a set $A \in \mathcal{E}$ is **measurable** in the sense that there is a way to assign a probability to A. For finite and discrete sample spaces, the most commonly used event space is the power set of $\mathcal{S}$, since in this case a probability can be assigned to each possible sample, and thus to each subset of samples (by summing the probabilities of the samples in the subset). In any case, it is required that the members of $\mathcal{E}$ form a **Boolean algebra** meaning that

1. $\emptyset, \mathcal{S} \in \mathcal{E}$
2. If $A \in \mathcal{E}$, then $\overline{A} \in \mathcal{E}$
3. If $A, B \in \mathcal{E}$, then $A \cup B \in \mathcal{E}$ and $A \cap B \in \mathcal{E}$.

Example 1.3. Consider the experiment of rolling a pair of dice. The sample space is $\mathcal{S} = \{2, 3, \ldots, 12\}$, while the event space is $\mathcal{P}(\mathcal{S})$, the power set of $\mathcal{S}$. Examples of events include $\{2\}$ the event of rolling a "2", $\{2, 4, 6, 8, 10, 12\}$, the event of rolling an even number, and $\{7, 11\}$ the event of rolling a winning number in the first roll of a craps game. □

For finite and countable sample spaces a Boolean-algebra event space is usually sufficient for describing the experimental events of interest. However, for an uncountable sample space it is more common to require that the members of $\mathcal{E}$ form a *σ-algebra*.

Definition 1.4. A **σ-algebra** is a Boolean algebra that is closed under *infinite* unions and intersections.

A σ-algebra is usually defined by starting with an existing Boolean algebra $\mathcal{A}$ of "basic" sets and then defining the associated σ-algebra $\mathcal{E}$ to be the smallest σ-algebra that contains $\mathcal{A}$. This idea is well defined since

1. any Boolean algebra $\mathcal{A}$ satisfies
$$\mathcal{A} \subseteq \mathcal{P}(\mathcal{S}),$$
2. the power set $\mathcal{P}(\mathcal{S})$ is a σ-algebra,
3. and the intersection of any number of σ-algebras is itself a σ-algebra.

Therefore, there is a unique "smallest" σ-algebra containing $\mathcal{A}$, namely the intersection of all σ-algebras that contain $\mathcal{A}$. Henceforth, we denote by $\sigma(\mathcal{A})$ the smallest σ-algebra that contains Boolean algebra $\mathcal{A}$.

The preference of having a σ-algebra event space for an uncountable sample space stems from the existence of interesting events that cannot be expressed as a finite union or intersection of basic events.

Example 1.5. For the sample space $\{0,1\}^\infty$, a **minimal prefix set** is one of the form $A\{0,1\}^\infty$, where $A \subseteq \{0,1\}^*$ is a finite **minimal prefix code**, meaning that it is a prefix code and, for any $w \in \{0,1\}^*$, either $w0 \notin A$ or $w1 \notin A$, since otherwise both $w0$ and $w1$ could be replaced by w to obtain a smaller prefix code. Letting $\mathcal{M}$ denote the set of minimal prefix sets, it is an exercise to show that $\mathcal{M}$ is a Boolean algebra where, e.g.,

$$C\{0,1\}^\infty = A\{0,1\}^\infty \cap B\{0,1\}^\infty$$

implies that C is a minimal prefix code and any bit sequence prefixed by a member of C is also prefixed by both a member of A and a member of B.

Letting $A = \{01, 11, 101, 1000\}$, and $B = \{00, 101, 110, 0100\}$, compute $A\{0,1\}^\infty \cup B\{0,1\}^\infty$, $A\{0,1\}^\infty \cap B\{0,1\}^\infty$, and $\overline{A\{0,1\}^\infty}$.

Solution.

1.4 Probability measure

The final ingredient of a probability measure space is a **probability measure** $\mu : \mathcal{E} \to [0, 1]$, where μ satisfies what are referred to as Kolmogorov's axioms of probability, namely that

1. $\mu(\mathcal{S}) = 1$ and
2. If $A_1, A_2, \ldots$ is an infinite sequence of disjoint events, then

$$\mu(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} \mu(A_i).$$

Proposition 1.6. Let $(S, \mathcal{E}, \mu)$ be a probability measure space. Then the following statements hold for arbitrary $A, B \in \mathcal{E}$.

1. $\mu(\emptyset) = 0$
2. $\mu(A \cup B) = \mu(A) + \mu(B) - \mu(A \cap B)$
3. $\mu(\overline{A}) = 1 - \mu(A)$

Proof. We prove the first two statements and leave the third as an exercise. By Axiom 2, $\mu(A) = \mu(A \cup \emptyset) = \mu(A) + \mu(\emptyset)$ which implies $\mu(\emptyset) = 0$.

For proving the second statement, first it is an exercise to show that a σ-algebra is closed under set difference. In other words, if $A, B \in \mathcal{E}$, then $A - B \in \mathcal{E}$. Then by Axiom 2 we have $\mu(A) = \mu(A-B)+\mu(A\cap B)$, which implies $\mu(A-B) = \mu(A)-\mu(A\cap B)$. Similarly, $\mu(B-A) = \mu(B)-\mu(A\cap B)$. Also by Axiom 2,

$$\mu(A\cup B) = \mu(A-B)+\mu(B-A)+\mu(A\cap B) = (\mu(A)-\mu(A\cap B))+(\mu(B)-\mu(A\cap B))+\mu(A\cap B) =$$

$$\mu(A) + \mu(B) - \mu(A \cap B).$$

Notice that, by Axiom 2 and the above proposition, if μ is defined over an algebra $\mathcal{A}$ of sets, then it is also defined over $\sigma(\mathcal{A})$. This is because every countable union of sets belonging to $\mathcal{E} = \sigma(\mathcal{A})$ may be written as a countable union of disjoint sets belonging to $\mathcal{E}$, and thus its probability measure can be computed via Axiom 2. Secondly, the probability measure of a countable intersection of sets may be computed by first computing the measure of its complement and subtracting that value from 1. Moreover, the complement of a countable intersection is a countable union of sets in $\mathcal{E}$ and thus may be computed based on the previous comment.

Example 1.7. For the algebra $\mathcal{M}$ defined in Example 1.5, we may define

$$\mu(A\{0,1\}^\infty) = \sum_{w \in A} 2^{-|w|} \leq 1$$

by Kraft's inequality. □

2 Martin-Löf Random Sequences

Definition 2.1. A set of words $W \subseteq \{0,1\}^*$ is **recursively enumerable (r.e.)** iff there is a computable function $f : \mathcal{N} \to \{0,1\}^*$ for which

$$W = \text{range}(f).$$

Since all computable functions may be effectively listed, we may assume that there is an effective listing of all r.e. sets: $W_0, W_1, \ldots$.

Definition 2.2. Set $\mathcal{B} \subseteq \{0,1\}^\infty$ is said to be a **constructive null set** iff there exists a total computable $g : \mathcal{N} \to \mathcal{N}$ for which

$$\mathcal{B} = \bigcap_{i=0}^{\infty} W_{g(i)}\{0,1\}^\infty$$

and $\mu(W_{g(i)}) \leq \frac{1}{2^i}$. Moreover, the collection $\{W_{g(i)} | i \geq 0\}$ is called a **constructive null cover** for $\mathcal{B}$.

Definition 2.3. The set of bit sequences **NULL** is defined as

$$\mathbf{NULL} = \bigcup \{\mathcal{B} : \mathcal{B} \text{ is a constructive null set}\},$$

while **RAND** is defined as

$$\mathbf{RAND} = \{0,1\}^\infty - \mathbf{NULL}.$$

Any bit sequence in **RAND** is called a **Martin-Löf random sequence** since the above definitions are attributed to Per Martin-Löf. Notice that, since there is only a countable number of constructive null covers and each constructive null set has probability measure 0, it follows that **RAND** has probability measure one.

Martin-Löf's definition of randomness is widely accepted as having captured the essence of randomness among bit sequences. To understand why, consider theorems in probability theory of the given type: "Let $X_0, X_1, \ldots$ be a sequence of binary random variables that are independent and for which X_i is the result of tossing a fair coin, then with probability one...[some property holds with respect to the values assumed by the random variables]." In other words, the set $\mathcal{B}$ of sequences that do *not* have this property has probability measure zero and is called the **null set** for the given property. Moreover, for all such known probability theorems, all null sets can be shown to be constructive, meaning that every sequence in **RAND** satisfies all such known theorems. Conversely, given a constructive null set $\mathcal{B}$, there is a provable theorem (why?) that states "with probability one, the sequence formed from $X_0, X_1, \ldots$ is a member of **RAND**.

Example 2.4. Suppose $s \in \{0,1\}^\infty$ satisfies $s_{2^i} = 1$ for all $i = 0, 1, 2, \ldots$. Then letting

$$V_n = \{0,1\}11\{0,1\}1\{0,1\}^3 1 \cdots \{0,1\}^{2^n-1}1,$$

for $n \geq 0$, we have

$$s \in \bigcap_{n=0}^{\infty} V_n\{0,1\}^\infty$$

and, since $\mu(V_n) = \frac{1}{2^{n+1}}$ and the words in V_n are recursively enumerable, $s \in \mathbf{NULL}$. □

The following theorem is due to Per Martin-Löf.

Theorem 2.5. Bit sequence $s \in \mathbf{RAND}$ iff there is a constant c, independent of n, for which

$$K(s[1 \ldots n]) \geq n - c.$$

3 Autoreducibility of Random Sequences

Definition 3.1. An **oracle** URM program is a URM program that has an additional type of instruction $Q(i)$, called the **query instruction**, which has the effect of replacing the contents of register R_i with a value that is returned by a B**-oracle** that is capable of solving in a single step a natural-number instance of some decision problem B.

Example 3.2. Suppose an oracle program is using a `Prime`-oracle and has the value 5 stored in register R_2. Then, after the execution of instruction $Q(2)$, R_2 will now hold the value 1 since 5 is a positive instance of `Prime`. On the other hand, if R_2 later holds the value 22, then, after executing $Q(2)$, R_2 will now hold the value 0, since 22 is a negative instance of `Prime`.

Definition 3.3. Let A be a decision problem and suppose oracle URM program P computes A's decision function $d_A(x)$ when using a B-oracle. Then we say that A is **Turing reducible** to B, and write

$$A \leq_T B.$$

From here onward we will use the terms "bit sequence" and "decision problem" interchangeably since associated with every decision problem A is the bit sequence

$$d_A(0)d_A(1)d_A(2)\cdots .$$

Similarly, every bit sequence may be viewed as a decision problem for which $d_A(i)$ equals the i th bit of the sequence.

Definition 3.4. Bit sequence A is **autoreducible** iff it is Turing reducible to itself via oracle program P and, for all $x \geq 0$, during the computation of $P(x)$, whenever P queries the A-oracle via query instruction $Q(i)$, x is not stored in R_i. In other words, P never queries A about x in order to compute $P(x)$.

Notice that every decidable decision problem is autoreducible since such a problem can be decided using a non-oracle URM program. But what about a bit sequence $A \in \mathbf{RAND}$? If on input x an oracle program P is unable to query A about x, then, considering that random sequences are formed by independent fair-coin tosses, it would seem that P would have to "guess" $A(x)$ and would be correct about 50% of the time in the long run. Thus, P would incorrectly decide A. On the other hand, what if we relaxed the definition so that P did not have to guess *all* the time, but only had to guess infinitely often.

Definition 3.5. Bit sequence A is **i.o. autoreducible** iff it is Turing reducible to itself via oracle program P and, for an infinite number of $x \geq 0$, during the computation of $P(x)$, whenever P queries the A-oracle via query instruction $Q(i)$, x is not stored in R_i. In other words, for an infinite number of x, P never queries A about x in order to compute $P(x)$.

At first glance it appears that for i.o. autoreducibility we run into the same problem as with total autoreducibility: since P still has to make an infinite number of guesses it is doomed to error 50% of the time and thus could not correctly decide A.

We now prove the following remarkable result.

Theorem 3.6. Every random sequence is i.o. autoreducible.

To prove Theorem 3.6 we need the following lemma.

Lemma 3.7. (Borel-Cantelli Lemma) Given a probability meaure space $(\mathcal{S}, \mathcal{E}, \mu)$ and events $A_1, A_2, \ldots \in \mathcal{E}$, if

$$\sum_{i=1}^{\infty} \mu(A_i) < \infty,$$

then the probability that an infinite number of the events will occur has measure zero. In other words, with probability one, only a finite number of the events will occur.

Proof. The event that an infinite number of the A_i occur can be expressed as an event in $\mathcal{E}$ since it corresponds with the set

$$\bigcap_{n=1}^{\infty} \bigcup_{k \geq n} A_k.$$

Indeed, for a sample of $\mathcal{S}$ that appear in the intersection it must be the case that the sample appears in infinitely many A_i. This is true since, for $n > i$ each A_i is not a member of the inner union and so, if a sample were only in a finite number of the A_i it would eventually be excluded from the inner union.

Finally, the above set indicates that the probability of an infinite number of A_i events occurring is less than or equal to

$$\sum_{k \geq n} \mu(A_k)$$

which converges to 0 as k increases. □

3.1 The Hat Problem

Statement of the Problem. n players on a team are each randomly assigned a hat, either red or blue in color. The team wins a one-million dollar prize provided at least one of the players can guess the color of the hat he is wearing and no player guesses incorrectly. Each player may view the hats of his teammates but cannot view his own hat. Although the players can collaborate to form a guessing strategy before the game begins, they are not allowed to communicate with each other during the game. What strategy should the team adopt in order to maximize its chance of winning?

3.2 Solution to the Hat Problem

To begin, order the players from 1 to n and identify the resulting hat-color sequence with a binary word, called the **observed word**, whose i th bit equals 1 iff player i is assigned a blue hat $i = 1, \ldots, n$. Moreover, we assume that the team adopts a deterministic strategy in the sense that player i is given a **guess book** $G_i \subseteq \{0,1\}^n$ that contains all the possible observed words that would cause him to guess the color of his hat. For example, if $w \in G_i$, then if player i observes that bits 1 through $i-1$ of the observed word equal $w_1 \cdots w_{i-1}$ and bits $i+1$ through n equal $w_{i+1} \cdots w_n$, then he guesses his hat color to be w_i. Now associated with player i's guess book is an **error set** E_i defined as

$$u \in E_i \text{ iff } \exists w \in G_i \text{ such that } u = w \oplus e_i,$$

where e_i denotes the ith **basis vector** of $\{0,1\}^n$ and consists of all zeros except for a one in position i. Thus, we have

$$|E_i| = |G_i|$$

for all $i = 1, \ldots, n$ and thus the solution to the Hat Problem involves

1. minimizing the cardinality of the set
$$\mathcal{E} = \bigcup_{i=1}^{n} E_i,$$
and
2. maximizing the cardinality of the set
$$\mathcal{G} = \bigcup_{i=1}^{n} G_i.$$

We call the members of $\mathcal{E}$ **error words** and call the members of $\mathcal{G}$ **good words**..

Definition 3.8. A **1-ball** with **center** $c \in \{0,1\}^n$ is the set of words

$$\{c, c \oplus e_1, \ldots, c \oplus e_n\}.$$

Each non-center word in a 1-ball is referred to as a **surface word**. Finally, the notation $\mathcal{B}(c)$ represents a 1-ball whose center is $c \in \{0,1\}^n$.

Proposition 3.9. Let $\mathcal{E}$ be a minimum set of error words amongst all possible deterministic guessing strategies. Then

$$|\mathcal{G}| \leq n|\mathcal{E}|,$$

with equality iff the set of 1-balls

$$\{\mathcal{B}(v) | v \in \mathcal{E}\}$$

are all pairwise disjoint and all surface words of each ball belong to $\mathcal{G}$.

Proof. Consider the map $f : \mathcal{G} \to \mathcal{E} \times \{1, \ldots, n\}$ where $f(w) = (v, i)$, where i is the least index for which $w \oplus e_i \in \mathcal{E}$. This map is well defined since, for every good word w, there is a player i who, when viewing $w_1 \cdots w_{i-1}$ and $w_{i+1} \cdots w_n$ guesses his hat color as w_i. Thus, $w \oplus e_i \in \mathcal{E}$. Moreover, f is a one-to-one function (verify!) and therefore

$$|\mathcal{G}| \leq |\mathcal{E} \times \{1, \ldots, n\}| = n|\mathcal{E}|.$$

Finally, notice that equality holds iff

1. for every $w \in \mathcal{G}$, there is a unique v and i for which $w = v \oplus e_i$ and
2. for every $v \in \mathcal{E}$ and index $i \in \{1, \ldots, n\}$ there is a $w \in \mathcal{G}$ for which $f(w) = (v, i)$.

But these two statements are equivalent to the set of 1-balls

$$\{\mathcal{B}(v) | v \in \mathcal{E}\}$$

being pairwise disjoint and all surface words of each ball belonging to $\mathcal{G}$.

Proposition 3.9 suggests that, to optimize their chance of winning, the team must find a set of 1-balls whose surfaces together form a maximum set of words within $\{0,1\}^n$ and for which no two balls overlap. Generally speaking, finding such a set of balls is considered a complex computational problem. However, in the case that $n = 2^k - 1$ for some $k \geq 1$, the problem becomes readily solvable. In this case there exist 2^{n-k} non-overlapping 1-balls whose union is $\{0,1\}^n$. Moreover, each 1-ball has exactly one error word and n good words, yielding an optimal probability equal to $n/(n+1)$ of winning the prize! To prove this we review some matrix theory.

Definition 3.10. The **rank** of a matrix A, denoted $\text{rank}(A)$, is the dimension of the A's column vector space.

Definition 3.11. The **nullity** of a matrix A, denoted $\text{nullity}(A)$, is the dimension of the vector space of vectors $\vec{x}$ for which $A\vec{x} = \vec{0}$.

Theorem 3.12. (Rank Theorem for Matrices) If A is a matrix with n columns, then

$$\text{rank}(A) + \text{nullity}(A) = n.$$

Proof. Let $k = \text{rank}(A) \leq r$ be given, where r is the number of rows of A. To determine A's null space, the standard procedure is to solve the homogeneous system of r linear equations, where the ith equation is

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = 0.$$

Of course, A itself is the matrix associated with this system and, by applying the Gauss-Jordan variable elimination algorithm, we may transform A and the associated system of equations into a matrix/system of the form

$$\begin{pmatrix} I_k & B \\ 0 & 0 \end{pmatrix},$$

where I_k is the $k \times k$ identity matrix and B is a $k \times (n-k)$ matrix. Thus, for any assignment of the variables x_{k+1} through x_{n-k}, there is a unique assignment to the variables x_1 through x_k which satisfies the associated system of equations. Therefore, the column space has dimension k while the NULL space has dimension $n-k$, and the theorem is proved. □

Theorem 3.13. Suppose $n = 2^k - 1$, and consider the $k \times n$ matrix A whose ith column is $(i)_2$, the number i written in binary. Then the following statements are true.

1. $\text{rank}(A) = k$
2. $\text{nullity}(A) = n - k$
3. The NULL space of A over the binary field $\{0, 1\}$ has 2^{n-k} vectors.
4. For every bit vector $\vec{v} \in \{0, 1\}^n$ for which $A\vec{v} \neq \vec{0}$, there is a unique null vector $\vec{c}$ for which $\vec{v} = \vec{c} + \vec{e_i}$, for some $i = 1, \ldots, n$.
5. $\{0, 1\}^n$ consists of 2^{n-k} pairwise disjoint 1-balls, where the center of each ball is a null vector.

Proof. Statements 1 and 2 directly follow from the Rank theorem. Statement 3 follows from the fact that, since the NULL space has $n-k$ basis vectors, there are a total of 2^{n-k} linear combinations that can be formed with these vectors. Moreover, by linear independence of the basis vectors, each linear combination forms a distinct vector and hence there are 2^{n-k} such vectors.

As for Statement 4, first notice that $A\vec{e_i} = (i)_2$ for every i (verify!) and let $\vec{v} \in \{0, 1\}^n$ be given with $A\vec{v} = (i)_2 \neq \vec{0}$. Then, $A(\vec{v} \oplus \vec{e_i}) = \vec{0}$ since

$$A(\vec{v} \oplus \vec{e_i}) = A(\vec{v}) \oplus A(\vec{e_i}) = (i)_2 \oplus (i)_2 = \vec{0}.$$

Thus $\vec{v} \oplus \vec{e_i} = \vec{c}$, for some null vector $\vec{c}$ which means that

$$\vec{v} = \vec{c} \oplus \vec{e_i}.$$

Conversely, if $\vec{v} = \vec{c'} \oplus \vec{e_j}$, for some null vector $\vec{c'}$ and $j \in \{1, \ldots, n\}$, then

$$A(\vec{v}) = (i)_2 = A(\vec{c'} \oplus \vec{e_j}) = A(\vec{c'}) \oplus A(\vec{e_j}) = \vec{0} \oplus (j)_2 = (j)_2 \Rightarrow i = j.$$

But then,

$$\vec{c} \oplus \vec{e_i} = \vec{c'} \oplus \vec{e_i}$$

and we get $\vec{c} = \vec{c'}$ after cancelling $\vec{e_i}$ from both sides. Therefore, every non-null vector $\vec{v}$ corresponds with exactly one surface word of some unique null vector.

Finally, Statement 5 follows from the previous statment which implies that the 1-balls centered at each of the null vectors are pairwise disjoint. Moreover, each of the 2^{n-k} 1-balls contains $n + 1 = 2^k$ vectors for a total of $2^{n-k} \cdot 2^k = 2^n$ total vectors which is all of $\{0, 1\}^n$. □

Corollary 3.14. For a team consisting of $n = 2^k - 1$ players, the optimal strategy is to assign player i a guess book G_i consisting of all 2^{n-k} binary words of the form $c \oplus e_i$, where $\vec{c}$ is a null vector of the matrix A defined in Theorem 3.13.

.

Proof. From Proposition 3.9 and Statement 5 of Theorem 3.13, we see that by assuming the null vectors of A are the error words, the good words must be those words on the surface of each 1-ball centered by an error word. Moreover, based on how each G_i is defined, for each good word, there is exactly one player who will guess in case that word represents the hat color sequence. Moreover, the guess will be correct so long as the color sequence does not represent a null vector. Therefore, the strategy yields the optimal probability of winning: namely $n/(n+1)$ since there is a 1 to n ratio of null vectors to surface words. □

Example 3.15. Provide the guess books for each player in case $n = 3$.

Example 3.16. Suppose the Hat Problem game is played with seven players. Will the team win if the hat color sequence is 1000101? For each i, list the outputs that player i observes when toggling his bit i.

3.3 Proof of Theorem 3.6

Partition the sequence $0, 1, 2, \ldots$ into finite nonoverlapping sequences $s_0, s_1, s_2, \ldots$ so that

$$s_1, s_2, \ldots = 0, 1, 2, \ldots$$

and $|s_i| = 2^i - 1$, for all $i \geq 1$. For given $x \in \mathcal{N}$, let $\text{seq}(x)$ denote the index of the sequence to which x belongs, and $\text{order}(x)$ denote x's order in the sequence. For example, if $x = 12$, then $\text{seq}(x) = 4$ and $\text{order}(x) = 2$. This is because $s_4 = 11, 12, 13, \ldots, 25$ and 12 is the 2nd number in this sequence.

Now consider the following oracle program P.

> Input x.
>
> Let $k = \text{seq}(x)$.
>
> Let $n = |s_k| = 2^k - 1$.
>
> Let M_k denote the $k \times n$ matrix whose columns are the numbers 1 through n expressed in binary.
>
> Let $i = \text{order}(x)$.
>
> Let $x_1, \ldots, x_i, x_{i+1}, \ldots, x_n$ denote the numbers in s_k.
>
> //Learn every bit except for $A(x)$
>
> Query the oracle to obtain its bits $A(x_1), \ldots, A(x_{i-1}), A(x_{i+1}), \ldots, A(x_n)$.
>
> For $i = 0, 1$, let $w_i = A(x_1) \cdots A(x_{i-1}) \cdot i \cdot A(x_{i+1}) \cdots A(x_n)$.
>
> For $i = 0, 1$, if $M_k(\vec{w_i}) = \vec{0}$, then return i. //Guess that $A(x) = i$
>
> Return query(x). //Forgo guessing and query the oracle

The program behaves in a way that, on input x, it behaves like player i in a Hat-Problem game having $n = |s_k| = 2^k - 1$ players. Moreover, with probability $n/(n+1)$ exactly one player will guess, and guess correctly. On the other hand, with probability

$$\frac{1}{n+1} = \frac{1}{2^k}$$

every player in this game will guess incorrectly. Thus, by the Borel-Cantelli lemma and since

$$\sum_{k=1}^{\infty} \frac{1}{2^k} = 1 < \infty,$$

with probability one there will only be a finite number of incorrect guesses despite there being an infinite number of (independent) games.

Finally, suppose we designate $A \in \mathbf{RAND}$ to be the oracle. Then, since the set of sequences that would cause an infinite number of incorrect guesses can be shown to be a constructive null set, it follows that when using A as oracle results in only a finite number of incorrect guesses. Let $y_1, y_2, \ldots, y_r \in \mathcal{N}$ denote the bit places for where an incorrect guess was made. Then modify P to make a new program P' so that it first checks if $x = y_i$ for some $i = 1, \ldots, r$, and, if so, returns $A(y_i)$. Then P' decides A and, in the course of doing so makes an infinite number of bit guesses, with each guess being correct. Therefore, A was aribtrary, every Martin-Löf random sequence is i.o. autoreducible. □