Written work about
A martingale approach to the study of occurrence of sequence patterns in repeated experiments [1]

Victor Klötzer
Erasmus student
Matrikel number 107637

1 Introduction

In this document, in the context of the Randomised Algorithms course proposed by Prof. Dr. Dirk Sudholt and Dr. Duc Cuong Dang at the University of Passau (Germany), I try to detail most of the results presented in the article of Shuo-Yen Robert Li called *A martingale approach to the study of occurrence of sequence patterns in repeated experiments*, see [1]. In this article, the author tries to generalize some results about expected waiting time for patterns to occur for the first time in a random process by using the martingale concept.

We consider a indefinite repetition of one same experiment that has a finite set of outcome values, and we then look for the first occurrence of a specific pattern of the outcome values. For instance, looking at a repeated fair coin toss until some pattern like heads-tails-heads-tails appears for the first time. In this context, Shuo-Yen Robert Li's article seeks to answer two main questions: What is the expected waiting time for the first occurrence of a given pattern? And what is the chance of a pattern appearing before another?

The article gives a new approach to deal with these questions using martingales. It gives a generalization of some well known results concerning the study of the occurrence of patterns, such as the case of a (symmetric) Bernoulli process treated in the book of William Feller, or an interesting phenomenon occurring in a coin toss game introduced by Walter Penney and presented in the Example 3.1 of this written work. In the last section of Shuo-Yen Robert Li's article [1], a direct relation to Conway's leading numbers is also given, those leading numbers being binary numbers used to compute the odds between two patterns in a balanced k-sided die repeating experiment. However, in Shuo-Yen Robert Li's article and thus in this written work, we only assume that the process has a finite set of outcome values and do not need any further assumptions about the distribution.

In this document, I first present the martingale approach used to find the result on the expected waiting time of the first appearance of a pattern in a repeated experiment, and I try to explain the intuition behind the use of martingales in the proof of Lemma 2.2. In a second step, I describe the results for competing sequences, i.e., the probability of one pattern occurring before another. For each of the results presented, I have tried to illustrate them and to explore them a little bit within the different examples of this document.

(As a side node, I noticed two small mistakes in the version of the paper mentioned here [1]. In Example 2.3 of this article, the operation $A{*}B$ should be equal to 12 and not 14. Also, in the end of the proof of the Theorem 3.3, the probabilities $\mathbb{P}(N_A = N_B)$ and $\mathbb{P}(N_A \neq N_B)$ have been inverted.)

In this document, the following notation and objects will be used:

- Z is an arbitrary but fixed discrete random variable
- Σ is the set of possible values of Z
- $Z_1, Z_2, \dots$ represents a sequence of independent, identically distributed (i.i.d.) random variables having the same distribution as Z
- $A = (a_1, a_2, \dots, a_m)$ and $B = (b_1, b_2, \dots, b_n)$ are two finite sequences over Σ
- N_B denotes the waiting time until B occurs the first time in a run of the process $Z_1, Z_2, \dots$, i.e., $N_B := \min\{k \geq 1 \mid (Z_{k-n+1}, Z_{k-n+2}, \dots, Z_k) = (b_1, b_2, \dots, b_n)\}$
- N_{AB} is the waiting time until B occurs for the first time given that the sequence A was already observed at the beginning of the process. We also assume that B is not a connected subsequence of $(a_1, a_2, \dots, a_{m-1})$.

  More formally: $N_{AB} := \min\left\{ k \geq 1 \;\middle|\; \begin{array}{c} (Z_1, Z_2, \dots, Z_m) = (a_1, a_2, \dots, a_m) \quad \text{and} \\ (Z_{k+m-n+1}, Z_{k+m-n+2}, \dots, Z_{k+m}) = (b_1, b_2, \dots, b_n) \end{array} \right\}$

Notice that the random variables N_B and N_{AB} are both stopping time, since for all $k \geq 1$ we have $\{N_B = k\} = \{Z_{k+m-n+1} = b_1, Z_{k+m-n+2} = b_2, \dots, Z_{k+m} = b_n\}$ and $\{N_{AB} = k\} = \{Z_1 = a_1, Z_2 = a_2, \dots, Z_m = a_m, Z_{k-n+1} = b_1, Z_{k-n+2} = b_2, \dots, Z_k = b_n\}$, i.e., the events $\{N_B = k\}$ and $\{N_{AB} = k\}$ are determined by realizations of Z that have occurred at most at time k.

2 Expected waiting time of a sequence

Before presenting the result about the expected waiting time for a sequence to occur, we need first to introduce an operation on two sequences.

Definition 2.1. *For two sequences $A = (a_1, a_2, \dots, a_m)$ and $B = (b_1, b_2, \dots, b_n)$ over Σ, we define the following operation:*

$$A{*}B := \sum_{k=1}^{m} \prod_{c \,\in\, diag(\Delta,k)} c$$

where

$$\Delta := [\delta_{ij}]_{i=1,\dots,m,\ j=1,\dots,n}$$

is a matrix of $\mathbb{R}^{m\times n}$ with

$$\delta_{ij} := \frac{1}{\mathbb{P}(Z = a_i = b_j)} 1_{\{a_i = b_j\}}$$

and

$$diag(\Delta, k) := \{\delta_{k,1}\,,\ \delta_{k+1,2}\,,\ \dots\}$$

is the set containing the coefficients of the k^{th} downward diagonal of the matrix Δ (the main diagonal being the 1^{st} downward diagonal).

Example 2.1. *Let $\Sigma = \{x, y, z\}$ with $\mathbb{P}(Z = x) = \frac{1}{2}$, $\mathbb{P}(Z = y) = \frac{1}{3}$ and $\mathbb{P}(Z = z) = \frac{1}{6}$. We consider the sequences $A = (y, x, x, z)$ and $B = (x, z, x)$.*

For instance, we then have $\Delta_{AB} := \begin{array}{c} \\ y \\ x \\ x \\ z \end{array}\!\!\begin{array}{c} \begin{array}{ccc} x & z & x \end{array} \\ \begin{pmatrix} 0 & 0 & 0 \\ 2 & 0 & 2 \\ 2 & 0 & 2 \\ 0 & 6 & 0 \end{pmatrix} \end{array}$ *and* $\Delta_{BB} := \begin{array}{c} \\ x \\ z \\ x \end{array}\!\!\begin{array}{c} \begin{array}{ccc} x & z & x \end{array} \\ \begin{pmatrix} 2 & 0 & 2 \\ 0 & 6 & 0 \\ 2 & 0 & 2 \end{pmatrix} \end{array}$.

Thus, $A*B = 0 \cdot 0 \cdot 2 + 2 \cdot 0 \cdot 0 + 2 \cdot 6 + 0 = 12$ *and* $B*B = 2 \cdot 6 \cdot 2 + 0 \cdot 0 + 2 = 26$*. Moreover, one should notice that* $*$ *is not commutative, e.g., here* $B*A = 0 \neq A*B$*.*

This $*$ operation is the key computation tool to get the expecting waiting time result of the next lemma.

Lemma 2.2. *Given a starting sequence* A*, the expected waiting time for a sequence* B *(that is not a connected subsequence of* A*) to occur for the first time is:*

$$\mathbb{E}(N_{AB}) = B*B - A*B$$

In particular if there is no starting sequence, then the expected waiting time of the sequence B *is* $\mathbb{E}(N_B) = B*B$*.*

In order to prove this lemma, we will use martingales and in particular a theorem about stopping time with martingales of Doob. Here is a recall of the definition of a martingale and of this theorem.

Definition 2.3. *A process* $X_1, X_2, \ldots$ *is called a martingale if for all* $k \geq 1$*,* X_k *is integrable, i.e.,* $\mathbb{E}(\,|X_k|\,) < \infty$*, and if for all* $k \geq 1$*,* $\mathbb{E}(X_{k+1} \,|\, X_k, X_{k-1}, \ldots, X_1) = X_k$*.*

Theorem 2.4 (Doob)**.** *Let* $X_1, X_2, \ldots$ *be a martingale and* N *be a stopping time. If* $\mathbb{E}(\,|X_N|\,) < \infty$ *and*

$$\liminf_{k \to \infty} \int_{\{N > k\}} |X_k| d\mathbb{P} = 0$$

then $X_1, \ldots, X_N$ *is a martingale and hence* $\mathbb{E}(X_N) = \mathbb{E}(X_1)$*.*

The second assumption of this theorem is mainly a constraint on the stopping time. Actually, this Doob theorem on stopping times can also be presented with stronger assumptions on the stopping time N, for example that it must be almost surely bounded. But we will need the weaker assumption given here for the proof of the lemma.

Proof of Lemma 2.2. For $k \in \mathbb{N}_0$, let ω_k denote the sequence $(Z_1, Z_2, \ldots, Z_k)$. Thus ω_k is a random sequence over Σ, ω_0 being the empty sequence. We define then the random variable

$$X_k := [A, \omega_k]*B - k$$

where $[A, \omega_k]$ is the sequence $A = (a_1, a_2, \ldots, a_m)$ followed by the sequence ω_k.

We want first to show that the stopped process $\{X_{k \wedge N_{AB}}\}_{k=0,1,2,\ldots}$ is a martingale. To do so, we define the following random variables for $k \geq 0$ and $j \geq 1 - m$:

$$M_k^{(j)} := \begin{cases} 0 & \text{if } j > k \\ -1 + \displaystyle\prod_{i=1}^{k-j+1} \frac{1}{\mathbb{P}(Z = b_i)} & \begin{array}{l}\text{if the last } k-j+1 \text{ elements in the sequence } [A, \omega_k] \text{ are} \\ \text{identical with } (b_1, \ldots, b_{k-j+1}) \text{, i.e., if } [A, \omega_k]_{m+j} = b_1, \\ [A, \omega_k]_{m+j+1} = b_2, \ \ldots, \ [A, \omega_k]_{m+k} = b_{k-j+1}\end{array} \\ -1 & \text{otherwise} \end{cases}$$

For $k \leq N_{AB}$, the variables $M_k^{(j)}$ can be interpreted as the net gain of gamblers in a fair game, where every gambler is waiting for the same pattern B to occur.

Consider for example the fair game where a die is repeatedly rolled showing the elements x, y or z with respective probabilities $\frac{1}{2}$, $\frac{1}{3}$ and $\frac{1}{6}$, and every gambler is waiting for the same pattern $B := (x, z, x)$. Each gambler starts with 1 euro and continues to bet successively for each of the elements in the pattern until the complete pattern appears or a wrong element is drawn, in which case they lose their entire bet. When the element is correct, the gambler receives their bet multiplied by the expected waiting time for the pattern value to appear; for example, a gambler first bets 1 euro for x to appear and receives $2 \cdot 1 = 2$ euros if x is actually drawn, because x has an expected waiting time of 2 (expectation of a geometric distribution). The gambler then directly places their entire bet on the next element of the pattern, i.e. 2 euros on z, etc.

The number k above can be seen as a round counter, and at each round one new gambler j enters the game. Thus, as long as the gambler j did not join the game, i.e., when $k < j$, their net gain is $M_k^{(j)} = 0$ at round k. As long as the gambler j is lucky and the elements of the pattern appear one after another, they have a net gain of the gross gain minus the starting stake of 1. And in the case where the gambler j lost because a wrong element was drawn, their net gain at round k is $M_k^{(j)} = -1$.

This game is fair in the sense that, in expectation, the casino and the gamblers have the same chance of winning. This is due to the fair design of the game, i.e. the way in which the stakes and payoffs have been set. The concept of martingale can therefore be used here: the expected gain at the end of each round is the same, and is identical to the starting gain. Thus, this shows for every fixed j, that the stopped process $\{M_{k\wedge N_{AB}}^{(j)}\}_{k=0,1,2,\dots}$ is a martingale.

Then,

$$\begin{aligned}
\sum_{j=1-m}^{\infty} M_k^{(j)} &= \sum_{j=1-m}^{k} M_k^{(j)} + \underbrace{\sum_{j\geq k+1} M_k^{(j)}}_{=0} \\
&= \sum_{j=1-m}^{k} \left(-1 + 1_{\{[A,\omega_k]_{m+j}=b_1,\, [A,\omega_k]_{m+j+1}=b_2,\, \dots,\, [A,\omega_k]_{m+k}=b_{k-j+1}\}} \prod_{i=1}^{k-j+1} \frac{1}{\mathbb{P}(Z=b_i)} \right) \\
&= -(k+m) + \sum_{j=1-m}^{k} \prod_{i=1}^{k-j+1} \frac{1}{\mathbb{P}(Z=b_i)} 1_{\{[A,\omega_k]_{m+j+i-1}=b_i\}} \\
&= -(k+m) + [A,\omega_k]{*}B \\
&= X_k - m
\end{aligned}$$

Hence $\{X_{k\wedge N_{AB}}\}_{k=0,1,2,\dots}$ is also a martingale, since it is a sum of martingales $\left\{\{M_{k\wedge N_{AB}}^{(j)}\}_{k=0,1,2,\dots}\right\}_j$ and of a constant m. Moreover, we then have

$$X_{N_{AB}} = [A, \omega_{N_{AB}}]{*}B - N_{AB} = B{*}B - N_{AB} \tag{1}$$

The last equality holds because $B = (b_1, b_2, \dots, b_n)$ appears for the first time at the end of $[A, \omega_{N_{AB}}]$ and therefore only the n last terms of the sum in $[A, \omega_{N_{AB}}]{*}B$ are non-zero (consider for instance the matrix notation as in Example 2.1), so $[A, \omega_{N_{AB}}]{*}B = B{*}B$.

We can now check that the assumptions of Theorem 2.4 of Doob hold. The random variable Z assumes every value in Σ with a positive probability, thus N_{AB} is dominated by a geometric random variable (by considering all the elements of the sequence B as independent for instance). Therefore $\mathbb{E}(N_{AB})$ is finite, and

$$\mathbb{E}(\,|X_{N_{AB}}|\,) = \mathbb{E}(\,|B{*}B - N_{AB}|\,) \leq \mathbb{E}(\,|B{*}B| + |N_{AB}|\,) = B{*}B + \mathbb{E}(N_{AB}) < \infty$$

by the triangle inequality. Hence $X_{N_{AB}}$ is integrable.

Furthermore, on the set $\{N_{AB} > k\}$, we have

$$|X_k| = |[A, \omega_k]{*}B - k| \leq |[A, \omega_k]{*}B| + |k| = [A, \omega_k]{*}B + k \leq B{*}B + N_{AB}$$

using again the triangle inequality and the fact that if the sequence B did not appear (completely), we have $[A, \omega_k]{*}B \leq B{*}B$.

This implies that $\lim\limits_{k\to\infty} \int_{\{N_{AB}>k\}} |X_k| d\mathbb{P} \leq \lim\limits_{k\to\infty} \int_{\{N_{AB}>k\}} \underbrace{(B{*}B + N_{AB})}_{\substack{\text{independent from } k \text{ and} \\ \text{bounded in expectation}}} d\mathbb{P} = 0.$

Thus, also $\liminf\limits_{k\to\infty} \int_{\{N_{AB}>k\}} |X_k| d\mathbb{P} = 0$, so that we can apply Doob's theorem and get that:

$$\mathbb{E}(X_{N_{AB}}) = \mathbb{E}(X_0) = [A, \omega_0]{*}B + 0 = A{*}B \ .$$

Hence, from (1) we obtain the final result

$$\mathbb{E}(N_{AB}) = \mathbb{E}(B{*}B - X_{N_{AB}}) = B{*}B - \mathbb{E}(X_{N_{AB}}) = B{*}B - A{*}B \ .$$

$\square$

Example 2.2. *As in Example 2.1, let $\Sigma = \{x, y, z\}$ with $\mathbb{P}(Z = x) = \frac{1}{2}$, $\mathbb{P}(Z = y) = \frac{1}{3}$, $\mathbb{P}(Z = z) = \frac{1}{6}$ and consider the sequences $A = (y, x, x, z)$ and $B = (x, z, x)$. Using the Lemma 2.2, given that the sequence A was observed, we can find the expected waiting time for the sequence B to appear by just computing $B{*}B = 26$ and $A{*}B = 12$. Thus,*

$$\mathbb{E}(N_{AB}) = B{*}B - A{*}B = 26 - 12 = 14 \ .$$

If no starting sequence A was observed, then we the expected waiting time to see the pattern B is

$$\mathbb{E}(N_B) = B{*}B = 26$$

which is larger than $\mathbb{E}(N_{AB})$. This makes intuitively sense, since at the end of the sequence A there is a part of the beginning of the pattern B, which can "help" to find B faster.

Example 2.3. *Consider a very simple AI (as clever as a monkey randomly hitting keys on a typewriter) that tries to generate a sentence given its start. This AI only uses the letter frequency in the English language to randomly pick a letter, see for instance [2] from where we can approximate the probabilities with $p_a = 8.12\%$, $p_b = 1.49\%$, $p_c = 2.71\%$, ..., $p_z = 0.07\%$. We want this AI to generate the sentence "to be or not to be" (without the blanks) given the start "to". How long can we expect to wait before this entire sentence is generated?*

Instead of considering each generation trial separately, we can consider all trials as a sequence of letters and be interested in when the desired pattern is expected to be created. We can therefore apply the above lemma with $A = (t, o)$, $B = (t, o, b, e, o, r, n, o, t, t, o, b, e)$ and the probabilities given in [2].

Thus, the expected time to create the sentence is

$$\begin{aligned}
\mathbb{E}\begin{pmatrix}\textit{generate}\\ \textit{"tobeornottobe"}\\ \textit{given "to"}\end{pmatrix} &= B{*}B - A{*}B \\
&= \left(\frac{1}{p_t \cdot p_o \cdot p_b \cdot p_e \cdot p_o \cdot p_r \cdot p_n \cdot p_o \cdot p_t \cdot p_t \cdot p_o \cdot p_b \cdot p_e} + \frac{1}{p_t \cdot p_o \cdot p_b \cdot p_e}\right) \\
&\quad - \left(\frac{1}{p_t \cdot p_o}\right) \\
&\simeq 2.84 \cdot 10^{15} + 7.99 \cdot 10^{4} - 143 \\
&\simeq 2.84 \cdot 10^{15}
\end{aligned}$$

A few observations can be made here. Since the sequence to be found is much longer than the starting sequence, having this "to" sequence to begin with does not significantly reduce the expected time to generate "to be or not to be". Moreover, compared to a simple coin toss, the state of possible values Σ is quite large here since there are 26 letters, so the probability of each value is low, so that in the end the time to generate the sequence becomes very large. Finally, even if the sentence "to be or not to be" has a somewhat specific structure, since a sub-pattern characterizes the sentence, the final expected waiting time depends mainly on the sole presence of the letters in "to be or not to be" without any order. In fact, for this particular case, considering only the probabilities and the number of occurrences of each letter without any order could be considered as a pretty good approximation.

3 Competing sequences

Let $B_1, B_2, \dots, B_k$ be k sequences over Σ. We are now interested in the probability that any sequence B_i precedes all others $k-1$ sequences in a realization of the process $Z_1, Z_2, \dots$. We assume again that none of these sequences contain any other sequence as a connected subsequence.

Let also A be a sequence given at the beginning of the process, and for all $i = 1, 2, \dots, k$, write N_i for N_{AB_i}. Then, let $N := \min\{N_1, N_2, \dots, N_k\}$ be the waiting time for the first of the k sequences to appear. In the next theorem, we want to compute the probabilities $\mathbb{P}(N = N_i)$ for all $i = 1, 2, \dots, k$.

Theorem 3.1. *Let Z and $Z_1, Z_2, \dots$ be discrete i.i.d. random variables, and $B_1, B_2, \dots, B_k$ be k finite sequences of Σ not containing each other. Let A be another such sequence not containing any B_i. Let N_i be the waiting times for sequence B_i to occur for the first time in the process. Given A as starting sequence, let p_i be the probability that B_i precedes the remaining $k-1$ sequences in a realization of the process $Z_1, Z_2, \dots$. Then, for every sequence i*

$$\sum_{j=1}^{k} p_j \, B_j{*}B_i = \mathbb{E}(N) + A * B_i$$

where $N := \min\{N_1, N_2, \dots, N_k\}$ is the stopping time for any B_i to occur.

This result can also be presented in a matrix form which shows a system of $k+1$ equations:

$$M\alpha = \beta \;\; :\Longleftrightarrow \;\; \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ -1 & B_1{*}B_1 & B_2{*}B_1 & \cdots & B_k{*}B_1 \\ -1 & B_1{*}B_2 & B_2{*}B_2 & \cdots & B_k{*}B_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -1 & B_1{*}B_k & B_2{*}B_k & \cdots & B_k{*}B_k \end{bmatrix} \begin{bmatrix} \mathbb{E}(N) \\ p_1 \\ p_2 \\ \vdots \\ p_k \end{bmatrix} = \begin{bmatrix} 1 \\ A{*}B_1 \\ A{*}B_2 \\ \vdots \\ A{*}B_k \end{bmatrix}$$

In particular, if no starting sequence is given, i.e. A is void, then we have $\beta = (1, 0, \ldots, 0)^\top \in \mathbb{R}^{k+1}$ *or in other word, for every i*

$$\sum_{j=1}^{k} p_j \, B_j{*}B_i = \mathbb{E}(N) \ .$$

The proof of this theorem is easily done using the Lemma 2.2.

Proof. Using the formula of total probability, notice first that for all $i = 1, 2, \ldots, k$, we have

$$\begin{aligned} \mathbb{E}(N_i) = \mathbb{E}(N) + \mathbb{E}(N_i - N) &= \mathbb{E}(N) + \textstyle\sum_{j=1}^{k} \mathbb{P}(N = N_j)\mathbb{E}(N_i - N \mid N = N_j) \\ &= \mathbb{E}(N) + \textstyle\sum_{j=1}^{k} p_j \, \mathbb{E}(N_i - N \mid N = N_j) \end{aligned}$$

Then, from Lemma 2.2 we get that $\mathbb{E}(N_i) = B_i{*}B_i - A{*}B_i$, and since $N_i - N \mid N = N_j$ represents the waiting time for B_i to occur for the first time given the starting sequence B_j, we also have $\mathbb{E}(N_i - N \mid N = N_j) = B_i{*}B_i - B_j{*}B_i$.

By substituting these results in the first equation, we get

$$\begin{aligned} B_i{*}B_i - A{*}B_i &= \mathbb{E}(N) + \textstyle\sum_{j=1}^{k} p_j \, (B_i{*}B_i - B_j{*}B_i) \\ &= \mathbb{E}(N) + B_i{*}B_i \underbrace{\textstyle\sum_{j=1}^{k} p_j}_{=1} - \textstyle\sum_{j=1}^{k} p_j \, B_j{*}B_i \\ &= \mathbb{E}(N) + B_i{*}B_i - \textstyle\sum_{j=1}^{k} p_j \, B_j{*}B_i \end{aligned}$$

Hence, we obtain $\sum_{j=1}^{k} p_j \, B_j{*}B_i = \mathbb{E}(N) + A * B_i$. □

From Theorem 3.1, we can derive the nice corollary below giving the odds between two sequences to occur before each other.

Corollary 3.2. *Let B_1 and B_2 be two not connected subsequence of each other. Then, the odds that sequence B_2 precedes sequence B_1 in a realization of the process $Z_1, Z_2, \ldots$ are*

$$(B_1{*}B_1 - B_1{*}B_2) : (B_2{*}B_2 - B_2{*}B_1) \ .$$

Proof. We directly apply Theorem 3.1 without starting sequence A, and with $k = 2$ for the sequences B_1 and B_2. By using the matrix notation, we get

$$\begin{bmatrix} 0 & 1 & 1 \\ -1 & B_1{*}B_1 & B_2{*}B_1 \\ -1 & B_1{*}B_2 & B_2{*}B_2 \end{bmatrix} \begin{bmatrix} \mathbb{E}(N) \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

where p_1 is the probability of B_1 to occur the first, and $p_2 = 1 - p_1$ the probability of B_2 to occur the first. Thus, we get the following system of equations

$$\begin{cases} -\mathbb{E}(N) + p_1 \, B_1{*}B_1 + p_2 \, B_2{*}B_1 = 0 \\ -\mathbb{E}(N) + p_1 \, B_1{*}B_2 + p_2 \, B_2{*}B_2 = 0 \end{cases} .$$

By subtracting the first equation from the second one, we obtain

$$p_1(B_1{*}B_1 - B_1{*}B_2) - p_2(B_2{*}B_2 - B_2{*}B_1) = 0 \ .$$

Hence,

$$\frac{p_2}{p_1} = \frac{B_1 * B_1 - B_1 * B_2}{B_2 * B_2 - B_2 * B_1} \ .$$

$\square$

Example 3.1. *Using the above corollary and the result of the Lemma 2.2, we want to look at an a priori unintuitive fact introduced by Walter Penney and mentioned in part 1. Outline and background of the article [1]. In the case of an indefinite coin toss process with a fair coin showing either heads "H" or tails "T", we can compute the expected waiting for the patterns $A := (H,T,H,T)$, $B_1 = (T,H,T,T)$, $B_2 = (H,T,H,H)$ and $B_3 = (H,H,T,H)$ using the lemma (without starting sequence):*

$$\begin{array}{c c} A\backslash A & \begin{array}{cccc} H & T & H & T \end{array} \\ \begin{array}{c} H \\ T \\ H \\ T \end{array} & \begin{pmatrix} 2 & \cdot & \cdot & \cdot \\ 0 & 2 & \cdot & \cdot \\ 2 & 0 & 2 & \cdot \\ 0 & 2 & 0 & 2 \end{pmatrix} \end{array} \implies \begin{aligned} \mathbb{E}(N_A) &= A*A \\ &= 16+4 \\ &= 20 \end{aligned}$$

$$\begin{array}{c c} B_1\backslash B_1 & \begin{array}{cccc} T & H & T & T \end{array} \\ \begin{array}{c} T \\ H \\ T \\ T \end{array} & \begin{pmatrix} 2 & \cdot & \cdot & \cdot \\ 0 & 2 & \cdot & \cdot \\ 2 & 0 & 2 & \cdot \\ 2 & 0 & 2 & 2 \end{pmatrix} \end{array} \implies \begin{aligned} \mathbb{E}(N_{B_1}) &= B_1*B_1 \\ &= 16+2 \\ &= 18 \end{aligned}$$

$$\begin{array}{c c} B_2\backslash B_2 & \begin{array}{cccc} H & T & H & H \end{array} \\ \begin{array}{c} H \\ T \\ H \\ H \end{array} & \begin{pmatrix} 2 & \cdot & \cdot & \cdot \\ 0 & 2 & \cdot & \cdot \\ 2 & 0 & 2 & \cdot \\ 2 & 0 & 2 & 2 \end{pmatrix} \end{array} \implies \begin{aligned} \mathbb{E}(N_{B_2}) &= B_2*B_2 \\ &= 16+2 \\ &= 18 \end{aligned}$$

$$\begin{array}{c c} B_3\backslash B_3 & \begin{array}{cccc} H & H & T & H \end{array} \\ \begin{array}{c} H \\ H \\ T \\ H \end{array} & \begin{pmatrix} 2 & \cdot & \cdot & \cdot \\ 2 & 2 & \cdot & \cdot \\ 0 & 0 & 2 & \cdot \\ 2 & 2 & 0 & 2 \end{pmatrix} \end{array} \implies \begin{aligned} \mathbb{E}(N_{B_3}) &= B_3*B_3 \\ &= 16+2 \\ &= 18 \end{aligned}$$

So, the expected waiting time for the sequence HTHT is 20, while the three sequences THTT, HTHH and HHTH all have the same smaller expected waiting time of 18. The fact that HTHT has a very repetitive and thus more complex structure than for instance THTT explains that its expected waiting time is a bit larger. Also, that THTT, HTHH and HHTH all have the same expected waiting time makes sense, since these three patterns have the same structure and the tosses are fair.

Now, one might expect each of the latter three patterns to win out over the HTHT pattern since their expected waiting times are smaller, but in fact this question is a bit more difficult to answer. Therefore, let us look at three different odds using the corollary.

To use the Corollary 3.2, we first have to compute all the crossed $*$ operations between A and the three other sequences:

$$\begin{array}{c c} B_1\backslash A & \begin{array}{cccc} H & T & H & T \end{array} \\ \begin{array}{c} T \\ H \\ T \\ T \end{array} & \begin{pmatrix} 0 & \cdot & \cdot & \cdot \\ 2 & 0 & \cdot & \cdot \\ 0 & 2 & 0 & \cdot \\ 0 & 2 & 0 & 2 \end{pmatrix} \end{array} \implies B_1*A = 0$$

$$\begin{array}{c c} A\backslash B_1 & \begin{array}{cccc} T & H & T & T \end{array} \\ \begin{array}{c} H \\ T \\ H \\ T \end{array} & \begin{pmatrix} 0 & \cdot & \cdot & \cdot \\ 2 & 0 & \cdot & \cdot \\ 0 & 2 & 0 & \cdot \\ 2 & 0 & 2 & 2 \end{pmatrix} \end{array} \implies \begin{aligned} A*B_1 &= 8+2 \\ &= 10 \end{aligned}$$

Using the same technique, we also have $B_2*A = 2$, $A*B_2 = 4$, $B_3*A = 10$ and $A*B_3 = 0$. Finally, we can use the result of the corollary and get the following odds:

For A against B_1: $(B_1*B_1 - B_1*A) : (A*A - A*B_1) = (18-0) : (20-10) = 9:5$.

For A against B_2: $(B_2*B_2 - B_2*A) : (A*A - A*B_2) = (18-2) : (20-4) = 1:1$.

For A against B_3: $(B_3*B_3 - B_3*A) : (A*A - A*B_3) = (18-10) : (20-0) = 2:5$.

These three examples show that even though $\mathbb{E}(N_{B_1}) = \mathbb{E}(N_{B_2}) = \mathbb{E}(N_{B_3}) = 18$, the odds against the sequence A can be in any direction, including also fair odds! Thus, when looking at the odds between *two sequences, not only the expected waiting time is a factor, but also the overlap of the sequences in one direction and in the other.*

A Python simulation of these results can be found in Appendix A.

For completeness, notice that a last result is presented in the studied article of Shuo-Yen Robert Li in the Theorem 3.3 [1]. It attempts to deal with the case of a tie between two sequences by looking at the probability for a sequence $A = (a_1, a_2, a_3, \ldots, a_m)$ to occur at the same time as a nested sequence of A which is $(a_2, a_3, \ldots, a_m)$.

References

[1] Shuo-Yen Robert Li. "A Martingale Approach to the Study of Occurrence of Sequence Patterns in Repeated Experiments". In: *The Annals of Probability* 8.6 (Dec. 1980), pp. 1171–1176. ISSN: 0091-1798. DOI: 10.1214/aop/1176994578. URL: https://projecteuclid.org/journals/annals-of-probability/volume-8/issue-6/A-Martingale-Approach-to-the-Study-of-Occurrence-of-Sequence/10.1214/aop/1176994578.full.

[2] *Frequency Table*. URL: https://pi.math.cornell.edu/~mec/2003-2004/cryptography/subs/frequencies.html.

A Appendix: Python simulation of the results of Example 3.1

These two Python 3 functions allow to simulate a repeated fair coin toss in order to approximate the expected waiting time of sequences as well as to get the odds between two patterns.

```python
import random
import pandas as pd

def first_hitting_time_of(pattern1, pattern2):
    """ Returns the first time each pattern was observed in a realization
        of the process Z_1, Z_2, ..., where Z_i represents a coin toss ("H" or "T")
        and all tosses are i.i.d. with Prob("H") = Prob("T") = 0.5 """

    T1, T2 = 0, 0 # first time to see pattern1 and pattern2
    found1, found2 = False, False # becomes True when a pattern is found
    memory_size = max(len(pattern1), len(pattern2))
    seq = "" # will store the last 'memory_size' tosses of the ongoing process
    t = 0
    while not found1 or not found2:
        seq += random.choice(["H","T"]) # perform a coin flip
        t += 1
        seq = seq[-memory_size:] # keep only the last 'memory_size' tosses
        if seq == pattern1 and not found1:
            T1 = t
            found1 = True
        if seq == pattern2 and not found2:
            T2 = t
            found2 = True

    return T1, T2

def mean_hitting_time_and_odds_for(pattern1, pattern2, n=100):
    """ Returns n hitting time values for both patterns and
        prints an approximation for the waiting times and the odds
        using the Law of Large Numbers """

    # Create a DataFrame containing n hitting time values for both patterns
    Ts = [first_hitting_time_of(pattern1, pattern2) for _ in range(n)]
    df = pd.DataFrame(Ts, columns=[pattern1, pattern2])

    # Print the expected waiting times and the odds
    print('Expected waiting time for the first occurrence:\n',
          f'  E("{pattern1}") = {df[pattern1].mean():.3},',
          f'E("{pattern2}") = {df[pattern2].mean():.3}')
    s1 = sum(df[pattern1] < df[pattern2])
    s2 = sum(df[pattern1] >= df[pattern2])
    print(f'Odds for "{pattern1}" against "{pattern2}":\n',
          f'  {s1/n*100:.3} %  :  {s2/n*100:.3} %')

    return df
```

For instance, for the sequence $A =$"HTHT" and $B =$"THTT", we can use the function as follows:

```python
# a pattern can have any length but must be a string composed only of "H" and "T"
A = "HTHT"
B = "THTT"
df = mean_hitting_time_and_odds_for(A, B, n=10_000)
```

and then get this output which suits the theoretical results obtained in Example 3.1:

```
> Expected waiting time for the first occurrence:
>    E("HTHT") = 20.2, E("THTT") = 18.0
> Odds for "HTHT" against "THTT":
>    64.1 %  :  35.9 %
```