BSc Thesis Applied Mathematics

A Markov process analysis of and a proposal for adjustments to the Leitner system

Anouk Beursgens

Supervisor: Werner Scheinhardt

February, 2022

Department of Applied Mathematics
Faculty of Electrical Engineering,
Mathematics and Computer Science

UNIVERSITY OF TWENTE.

Preface

I would like to thank my supervisor dr. ir. Werner Scheinhardt for imagining that the Leitner system could be modelled as a Markov process and suggesting this as a research topic. Furthermore, I would like to thank him and those closest to me for their feedback.

A Markov process analysis of and a proposal for adjustments to the Leitner system

Anouk Beursgens*

February, 2022

Abstract

The Leitner system is a review scheme for flashcards that uses spaced repetition. Although there has been quite some research performed in ways to implement spaced repetition on computers, no existing simple mathematical model for the physical use of the Leitner system could be found. The Leitner system is modelled as a discrete-time inhomogeneous Markov process. The system is analysed using three performance measures, taking into account the efficiency of the system and the distribution of the workload. The relation between the number of words in the system and both the number of words mastered after as well as reviewed on day t is reasoned to be linear. Moreover, the ratio between the number of words reviewed on two days of a cycle is shown to be fixed. Furthermore, the influence of the global difficulty on the results is examined. Several adjustments to the Leitner system are proposed to make the system more user friendly, both in equalizing the distribution of workload over days and in adjusting it to weekly cycles. Finally, the lessons learned from those adjustments are used to propose an alternative to the Leitner system. However, the influence of the adjustments to the system on the long-term retention of the studied words could not be examined and needs further research.

Keywords: *Leitner system, time-inhomogeneous Markov process*

1 Introduction

Many students struggle with learning words of a foreign language in high school. Even after high school quite some people try to learn a new language. There has been a lot of research into the working of the human brain and memory models. From this research, it became apparent that a particular learning strategy called spaced repetition is beneficial for learning words. As the name already suggests, the repetition of words is spaced out, causing more words to be stored in the long-term memory. Spaced repetition is in stark contrast to cramming, where words are repeated very often in one single learning session. There is an abundance of research, like [3], [4] and [7], showing that spaced repetition gives better learning results than cramming. An excellent implementation of spaced repetition is the use of so-called flashcards. The student writes a word on the front of the card and its translation on the back. When practising with flashcards, the student tries to come up with the translation and then flips the card over to check the given answer.

One of the first persons to develop a good system for learning new items with the use of flashcards is Sebastian Leitner. In his book *So lernt man lernen* [6] from 1972 he

*Email: a.e.beursgens@student.utwente.nl

introduced a repetition scheme which is widely used and rising in popularity again in the past few years. Although in his *Lernkartei* he uses a large box with numbered sections of varying sizes which indicate when certain words should be reviewed, the widely used Leitner system has a fixed schedule, the Leitner calendar, which prescribes which deck needs to be reviewed when so that no special box is needed. In this report the second form will be considered. Each flashcard will be located in one of the seven decks. When studying a deck of flashcards, the student checks for each word in the deck whether he knows the translation of the word. If he knows the translation, the card goes one deck up, else it goes back to deck one, as illustrated in Figure 1. Words reviewed correctly in the last deck are considered mastered and will leave the system. This means that they are not reviewed anymore.

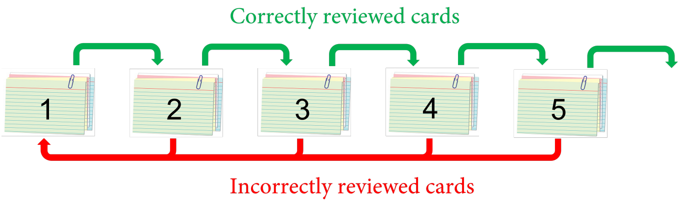


FIGURE 1: Transition scheme for review of words.

Every deck has a different repetition interval. Lower decks are repeated often, deck 1 even daily, while higher decks are repeated less. So, in the long run, a student spends more time on words that are difficult for him. The Leitner calendar is given in Figure 2.

Day	Decks
1	1 2
2	1 3
3	1 2
4	1 4
5	1 2
6	1 3
7	1 2
8	1
9	1 2
10	1 3
11	1 2
12	1 5
13	1 2 4
14	1 3
15	1 2
16	1
17	1 2
18	1 3
19	1 2
20	1 4
21	1 2
22	1 3
23	1 2
24	1 6
25	1 2
26	1 3
27	1 2
28	1 5
29	1 2 4
30	1 3
31	1 2
32	1
33	1 2
34	1 3
35	1 2
36	1 4
37	1 2
38	1 3
39	1 2
40	1
41	1 2
42	1 3
43	1 2
44	1 5
45	1 2 4
46	1 3
47	1 2
48	1
49	1 2
50	1 3
51	1 2
52	1 4
53	1 2
54	1 3
55	1 2
56	1 7
57	1 2
58	1 3
59	1 2 6
60	1 5
61	1 2 4
62	1 3
63	1 2
64	1

FIGURE 2: The Leitner calendar for a learning cycle of 64 days, after which the system repeats itself.

In past years quite some research in optimising human learning has been performed. Some studies focus on implementing spaced repetition, like [9], [10] and [14]. However, their approaches are quite complex and they focus on the selection of one item per time. This is alright for implementation in online flashcard tools but not ideal for a physical version. Furthermore, no found research matches the Leitner system as described above. This paper is therefore the first accessible mathematical analysis of the Leitner system. More specifically, in this paper, the following research question is discussed:

How can the Leitner system be made more user friendly while at the same time maintaining the usefulness of spaced repetition?

To answer this research question, a distinction is made between two ways to improve the Leitner system. The first one is to make sure that the expected number of words to be studied on each day of a learning cycle is approximately steady. The second one is to make the review schedule easier to remember by adjusting it to biweekly learning cycles.

In this paper, the Leitner system as described above is modelled. The transition matrices and results are provided for a Leitner system with five decks. A learning cycle will then take 16 days and is the same as the left column of Figure 2. As Figure 1 might suggest, the Leitner system can be modelled as a Markov process. The time steps are discrete, namely a day. After a certain number of days the review scheme is the same as if one would start again from day 1, so the Leitner system is modelled as a cyclic Markov process. Due to this cyclic behaviour, it is often necessary to refer to a certain day of each cycle. Therefore, throughout this paper an asterisk will be used to indicate that the mentioned day y refers to the y^{th} day of each cycle and not only to day y itself. That is, y^* denotes all days t for which $t = y + mT$, $y \leq T, m = 0, 1, 2, 3, \ldots$, where T is the length of a learning cycle, here 16 days.

Some theoretical background about cyclic Markov processes is given in Section 2. This section also contains a small literature review on the recall probability that is needed to determine transition probabilities. After the model is set up in Section 3, the model is analysed by means of three analytic performance measures in Section 4. Finally, in Section 5, several possible adjustments to the Leitner system are discussed and their performances are compared. The section ends with proposing an alternative to the Leitner system. In Section 6, the limitations of this research are discussed and suggestions for future research are made. An overview of the main symbols used in this paper can be found in Appendix A.

2 Theoretical Background

In this section some theoretic background is presented that is needed in this research.

2.1 Markov process

Most material on Markov processes immediately focuses on time-homogeneous Markov processes, called Markov chains. In this report, however, a time-*inhomogeneous* Markov process is used.

Consider a stochastic process $\{X_t, t = 1, 2, \ldots\}$ with discrete time. As can be found in most textbooks on Markov chains, for example [11] and [15], $X_t = i$ denotes that the process is in state i at time point t and this stochastic process is a Markov process if

$$P(X_{t+1} = j | X_t = i, X_{t-1} = i_{t-1}, \ldots, X_1 = i_1, X_0 = i_0) = P(X_{t+1} = j | X_t = i)$$

for all states i_0, i_1,…,i_{t-1}, i, j and all $t \geq 0$. An important difference with a time-homogeneous Markov chain is that the transition probability $p_{i,j}(t)$ depends on time t. That is,

$$p_{i,j}(t) := P(X_{t+1} = j | X_t = i)$$

subject to

$$p_{i,j}(t) \geq 0, \quad i, j \geq 0 \qquad \text{and} \qquad \sum_{j=0}^{\infty} p_{i,j}(t) = 1, \quad i = 0, 1, \ldots, \quad \forall t \leq 0.$$

At each time step t, the Markov process has an associated transition matrix P_t with $[P_t]_{i,j} = p_{i,j}(t)$. This is in contrast to a Markov chain for which only a single transition matrix is needed. It can happen that a part of the process repeats itself every T time steps. That is, the Markov process is cyclic with period $T \in \mathbb{N}$, where T is the smallest number such that $p_{i,j}(mT + y) = p_{i,j}(y) \quad \forall m \in \mathbb{N},\ 0 < y \leq T$. The Markov process can then be described by T transition matrices according to [12]. The possible transitions after one cycle are given by the matrix product $\mathbf{P} = P_1 P_2 \cdots P_T$. This matrix $\mathbf{P}$ does not depend on time anymore. Due to the Chapman-Kolmogorov equations for this time-homogeneous $\mathbf{P}$, the t-step transition matrix with $t = mT + y$ is given by

$$P^{(t)} = P_1 \cdots P_{t-1} P_t = (P_1 \cdots P_{T-1} P_T)^m P_1 \cdots P_y = \mathbf{P}^m P_1 \cdots P_y.$$

In this paper, a Markov process with an absorbing state is considered. Since absorption is certain, the question remains how long it takes until the process enters the absorbing state. That is, the mean time to absorption. Please note that the method described here to determine the mean time to absorption assumes a time-homogeneous matrix, but this is still useful due to the time-homogeneous matrix $\mathbf{P}$. Let Q be the submatrix of $\mathbf{P}$ from which the rows and columns corresponding to absorbing states are removed. This matrix Q is used to calculate the fundamental matrix $F := (I - Q)^{-1}$, where I denotes the identity matrix of the same size as Q and $^{-1}$ the inverse. The entry $F_{i,j}$ denotes the average number of time steps the process is in transient state j before absorption when started in transient state i. The inverse of $(I - Q)$ exists by Theorem 3.2.1 of [5]. Let $\boldsymbol{\alpha}_0$ denote the initial state distribution at time $t = 0$, where $[\boldsymbol{\alpha}_0]_i = P(X_0 = i)$. Summing up all entries of $\alpha_0 F$ yields the mean time to absorption.

2.2 Recall probability

To be able to determine realistic transition probabilities $p_{i,j}(t)$, some background information about the probability that a certain item is reviewed correctly is needed. This probability is called the recall probability and will be denoted by $P_{\text{recall}}(t)$. It is assumed that a word will be reviewed correctly if the user still knows the translation of the word. To determine $P_{\text{recall}}(t)$, a straightforward idea would be to use the forgetting curve. The forgetting curve was first introduced by Ebbinghaus in 1885 [3]. It shows the decline of memory retention in time by plotting the percentage of learning material retained against time. The corresponding model for the percentage retained, $P_{\text{recall}}(t)$, is

$$P_{\text{recall}}(t) = \frac{100k}{c \cdot \log_{10}(t) + k}, \tag{1}$$

where t is the time since last reviewed and c and k are constants. Unfortunately, both the research of Ebbinghaus as well as a replicating study [8] were performed with only one data subject and concerned the learning of meaningless syllables. It is therefore unknown how well Equation (1) generalizes.

A variant on the 'Ebbinghaus Forgetting Equation' is examined in the paper *Unbounded Human Learning: Optimal Scheduling for Spaced Repetition* [10] where the authors look at different forms of exponential powers and assume that recall is binary (a user either completely recalls or forgets an item). They test the different formulas on empirical data of Mnemosyne, which is a popular flashcard software tool, to see which one fits best. Based on their observations, they adopt the following exponential forgetting curve

$$P_{\text{recall},z}(t) = e^{-\frac{\theta \cdot t}{k_z}}, \tag{2}$$

where θ is the global difficulty, t is the time since last reviewed and k_z is the current deck for a word z. By means of the global difficulty θ, Equation (2) can be adjusted to take into account the level of the student, but also the general difficulty of a language. To illustrate the latter, most Dutch people will have more trouble learning Chinese words than learning German words, so for them θ for Chinese will be higher than for German. Since in this research flashcards are used and Equation (2) is based on flashcard data, Equation (2) will form the basis for the recall probability in this research.

3 Model

The Leitner system, as given in the first column of Figure 2, is modelled as a discrete inhomogeneous Markov process, where the time steps are days. In the original Leitner system words can arrive to a deck on different days before the next review. To make sure that the basic setup of the model is understood well, first a naive model is given in which these fragmented arrivals of words is not taken into account yet. That is, the recall probability does not take into account the exact time since the last review of an individual word. In this chapter, the states, transition probabilities and transition matrices for this naive model are explained thoroughly in Section 3.2, Section 3.3 and Section 3.4 respectively. After that, in Section 3.5, a distinction is made between batches of words arriving at different times, which makes the described Markov process suitable as a model for the Leitner system. Only the adjustments to the naive model are mentioned. Before the naive model is described further, the assumptions that are made during the modelling are listed in the next paragraph. Throughout this paper, let d denote the number of decks and w the number of words in the system.

3.1 Assumptions

There is one 'fixed' review session per day. During this review session all the decks that are on the Leitner calendar for that day are reviewed entirely. Since time is measured in days, there is exactly one day between two consecutive review sessions. This means that a difference in hours that could be caused by the first review session taking place in the evening and the next one in the morning in reality is neglected.

Each word will be reviewed at most once a day. Since every day deck 1 is studied, often together with a higher deck, it can happen that a word moves to a deck that is reviewed after the first deck of that day. The word would then be reviewed only shortly before the second encounter, so the chance that the student still knows the word is significantly higher. Therefore it is decided that each word can at most be reviewed once a day.

The recall probability is deck-specific. A deck-specific recall probability p_k is used, since research, [10], shows that the recall probability depends strongly on the word complexity. A uniform recall probability p would therefore be unrealistic. However, a recall probability depending on word difficulty would be too difficult and not suited for this approach, since then the intrinsic difficulty of each word needs to be determined. Moreover, the recall probability does not take into account how often a certain word has been reviewed before. This is necessary to have independence of the past (the way a word arrived to a certain deck), which is needed to be able to model the system as a Markov process, without having to store the number of past reviews of each word in the state description, which would increase the size of the state space too much.

The recall of words is independent. There is no dependence between a single recall of words in a certain deck and there is also no dependence between the recall of words of two different decks.

All words start in deck 1 and the student has studied them once. Before the first day, all words are in the deck 1. The day before the Leitner calendar starts, the student has studied all the words once. This is a realistic assumption since the student has probably written the flashcards on that day. This guarantees that there can be spoken of a recall on the first day.

3.2 States

It is important to know how many words are in each deck. Let n_k denote the number of words in deck k, n_m the number of words mastered and c_k the number of words that are reviewed correctly in deck k, $k \in \{1, 2, \ldots, d\}$. The state description is a vector $\mathbf{n} = (n_1, n_2, \ldots, n_d, n_\mathrm{m})$ containing the number of words in each deck at the end of the day, after a review session. Storing n_m in the state is both convenient to keep track of the number of mastered words, which tells something about the efficiency of the systems, as well as to ensure that all entries of a state always sum up to the number of words in the system. So, for all $\mathbf{n}$, we have $\sum_{k=1}^{d} n_k + n_\mathrm{m} = w$. The state space S therefore consists of $\binom{w+d}{w}$ vectors.

3.3 Transition probabilities

Each word will be recalled correctly with deck-specific recall probability, p_k. Since the recall probability used here is deck-specific, Equation (2) is slightly adjusted to

$$p_k := P_{\mathrm{recall},k}(t) = e^{-\frac{\theta \cdot t_k}{k}}, \tag{3}$$

where k is the number of the reviewed deck and t_k is the number of days since the words in deck k are last reviewed. To all decks, except deck 1, words can arrive on different days before the next review. In the naive model, the average between the arrival days is taken. For example, words correctly reviewed in deck 2 on day 7* and 9* both are next reviewed in deck 3 on day 10*. So, $t_3 = \frac{3+1}{2} = 2$. In Section 3.5 the naive model is adjusted so that a distinction between these different arrivals is made. For $k \in \{2, \ldots, d\}$, t_k will become a vector with the different arrival days that occur in deck k.

The recall probabilities are not the same as the transition probabilities between states, since multiple words, and even decks, are reviewed in one review session. The transition probability uses the probability that a certain amount of words, c, is recalled correctly from a total amount of words, n, where the probability of a correct recall is the recall probability given in Equation (3). Let the random variable C_k denote the number of correct recalls in deck k. Due to the independence between the recall of words, $C_k \sim B(n_k, p_k)$. Define q_{n_k,c_k,p_k} as

$$q_{n_k,c,p_k} := P(C_k = c) = \binom{n_k}{c} p_k^c (1 - p_k)^{n_k - c}. \tag{4}$$

On most days, multiple decks need to be studied. Due to the assumed independence between the recall of different decks, the total transition probability for that day is just the product of the individual transitions of each deck.

3.4 Transition matrices

The transition matrix P_t depends on which decks have to be reviewed on day t. Due to the cyclic behaviour of the Markov chains, only a limited number of unique transition matrices are necessary. Moreover, even inside a cycle, certain combinations of decks to be reviewed appear more often, so the transition matrices for these days will be the same. The dummy variable $s \in \{0, 1, 2, \dots\}$, will be used in the subscript of the matrices to have a short-hand notation for when the matrix is applicable for multiple days. To prevent the notation to become too cumbersome for the realistic model, the transition matrices are explicitly given for $d = 5$.

Constructing the transition matrices is quite straightforward once you understand how the transition process works. Therefore, only the construction of the transition matrix for the days on which deck 1 and 2 are reviewed is explained thoroughly. The other transition matrices can be found in Appendix B.1.

When deck 1 and 2 are reviewed, only n_1, n_2 and n_3 can change. The words from deck 2 that are reviewed correctly will go to deck 3 and the rest will go to deck 1. That is, the number of words in deck 3 will increase by c_2 while the number of words in deck 1 will increase by $n_2 - c_2$. Words that are correctly reviewed in deck 1 will go to deck 2, while those that were incorrect will stay in deck 1. So, the number of words in deck 1 will decrease by c_1 and the number of words in deck 2 will become c_1. The latter is the case since all words that were initially in deck 2 already went to deck 1 or 3 due to the review of deck 2, so without the arriving c_1 words the deck would have been empty at the end of the day. Due to the assumption that no words are reviewed twice, the number of correctly reviewed words in a deck can not exceed the number of cards in that deck at the start of the day. In conclusion, on a day on which deck 1 and 2 are reviewed, only transitions from a state $(n_1, n_2, n_3, n_4, n_5, n_\text{m})$ to the state $(n_1 - c_1 + n_2 - c_2, c_1, n_3 + c_2, n_4, n_5, n_\text{m})$ are possible. The probability that such a transition takes place is given by $q_{n_1,c_1,1} * q_{n_2,c_2,2}$, where q is as defined in Equation (4). Deck 1 and 2 are studied on all odd days, except on day 13 when deck 4 is studied as well. That is, on days $1 + 2s\backslash 13$*, $s = 0, 1, 2, \dots$. This means that the transition matrix for all odd days except day 13 is given by

$$\begin{aligned}
[P_{1+2s\backslash 13^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} &= q_{n_1,c_1,p_1} * q_{n_2,c_2,p_2} \text{ for } \boldsymbol{\delta} = (-c_1 + n_2 - c_2, -n_2 + c_1, c_2, 0, 0, 0) \\
&\quad \text{if } c_1 \leq n_1,\ c_2 \leq n_2 \\
&= 0 \quad \text{else.}
\end{aligned}$$

3.5 Adjust the naive model to a realistic model for the Leitner system

In all decks except deck 1 words arrive on multiple days before that deck is reviewed again. Since some words have arrived to that deck in an earlier review session than others, the time since the last review is larger for these words. These words should therefore have a lower recall probability than the words that arrived on a more recent day. So, a distinction should be made between these sets of words in such a way that these sets have a different recall probability, and thus a more accurate transition probability for the total deck. The state description is changed: for $k \in \{2, \dots, d\}$ n_k is changed from a single number, the number of words in deck k, to a vector $(n_{k,\text{old}},\ n_{k,\text{new}})$. The first entry of the vector, $n_{k,\text{old}}$, contains the number of words that arrived first since the previous review of deck k. Naturally, the second entry, $n_{k,\text{new}}$, then contains the number of words that arrived in the second review session of one deck lower. The state description becomes
$(n_1,\ (n_{2,\text{old}},\ n_{2,\text{new}}),\dots,(\text{n}_{d,\text{old}},\ n_{d,\text{new}}),\ n_\text{m})$.

The transition matrices change as well. In order to determine the transition probability, it should be taken into account to which of the two subdecks of deck k the words should go when reviewing deck $k-1$ and in which of the two subdecks of deck k the words were when reviewing deck k. The realisation of these two different distinctions that have to be made, is described below and illustrated with a transition matrix. The other transition matrices can be found in Appendix B.2.

The latter of the two is achieved by not summing over the total number of words that can be done correctly in deck k, with as maximum n_k, but by distinguishing from which subdeck the words are. So, summing over each of the two subdecks with as maximum $n_{k,\text{old}}$ and $n_{k,\text{new}}$ respectively. This is important since the recall probability for these subdecks are different. As mentioned in Section 3.3, $\mathbf{t_k}$ is now a vector for $k \in \{2, \dots, d\}$, yielding a $p_{k,\text{old}}$ and a $p_{k,\text{new}}$ for the two respective subdecks of deck k. For example, the transition matrix for day 12*, on which deck 1 and 5 are reviewed, will now become

$$
\begin{aligned}
[P_{12^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = {} & q_{n_1,c_1,p_1} * q_{n_{5,\text{old}},c_{5,\text{old}},p_{5,\text{old}}} * q_{n_{5,\text{new}},c_{5,\text{new}},p_{5,\text{new}}} \quad \text{for} \\
& \boldsymbol{\delta} = (-c_1 + n_{5,\text{old}} - c_{5,\text{old}} + n_{5,\text{new}} - c_{5,\text{new}}, (0, c1), (0,0), (0,0), (0,0), \\
& \qquad c_{5,\text{old}} + c_{5,\text{new}}), \\
& \text{if } c_1 \le n_1,\ c_{5,\text{old}} \le n_{5,\text{old}},\ c_{5,\text{new}} \le n_{5,\text{new}} \\
= {} & 0 \quad \text{else.}
\end{aligned}
$$

To which of the two subdecks a correctly reviewed word should go, depends on the day and can be achieved by constructing more unique transition matrices. To illustrate this, look at day 3* and 5*. On both days deck 1 and 2 are reviewed. So, in the naive model the transitions of both days were represented by the transition matrix $P_{1+2s\backslash 13^*}$. However, after day 3* deck 3 is reviewed on day 6* for the first time, meaning that both the words of deck 2 that are reviewed correctly on day 3* and 5* have their next review on day 6*. The words of deck 2 that were done correctly on day 3* should therefore be stored in the 'old' subdeck of deck 3, while the words from day 5* should go to the 'new' subdeck of deck 3. So, instead of the same transition matrix, day 3* (and 7*, 11* and 15*) and 5* (and 1* and 9*) are now defined by different transition matrices

$$
\begin{aligned}
[P_{3+4s}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = {} & q_{n_1,c_1,p_1} * q_{n_{2,\text{old}},c_{2,\text{old}},p_{2,\text{old}}} * q_{n_{2,\text{new}},c_{2,\text{new}},p_{2,\text{new}}} \quad \text{for} \\
& \boldsymbol{\delta} = (-c_1 + n_{2,\text{old}} - c_{2,\text{old}} + n_{2,\text{new}} - c_{2,\text{new}}, (-n_{2,\text{old}} + c_1, -n_{2,\text{new}}), \\
& \qquad (c_{2,\text{old}} + c_{2,\text{new}}, 0), (0,0), (0,0), 0), \\
& \text{if } c_1 \le n_1,\ c_{2,\text{old}} \le n_{2,\text{old}},\ c_{2,\text{new}} \le n_{2,\text{new}} \\
= {} & 0 \quad \text{else}
\end{aligned}
$$

$$
\begin{aligned}
[P_{1+4s\backslash 13^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = {} & q_{n_1,c_1,p_1} * q_{n_{2,\text{old}},c_{2,\text{old}},p_{2,\text{old}}} * q_{n_{2,\text{new}},c_{2,\text{new}},p_{2,\text{new}}} \quad \text{for} \\
& \boldsymbol{\delta} = (-c_1 + n_{2,\text{old}} - c_{2,\text{old}} + n_{2,\text{new}} - c_{2,\text{new}}, (-n_{2,\text{old}} + c_1, -n_{2,\text{new}}), \\
& \qquad (0, c_{2,\text{old}} + c_{2,\text{new}}), (0,0), (0,0), 0), \\
& \text{if } c_1 \le n_1,\ c_{2,\text{old}} \le n_{2,\text{old}},\ c_{2,\text{new}} \le n_{2,\text{new}} \\
= {} & 0 \quad \text{else.}
\end{aligned}
$$

A small remark on the words that arrive in deck 4: they arrive either two and six days in advance or three and seven days in advance. The first is the case on day 4* while the second is the case on day 13*. An extra distinction in recall probability, in the form of $\mathbf{t}_4 = ((2,6),(3,7))$, is therefore necessary. Fortunately, no changes to the state description are necessary.

4 Analysis of the model

4.1 Performance measures

The performance of the Leitner system will be analysed by means of three performance measures:

1. Mean number of cycles until all words are mastered
2. Expected number of words mastered after t days
3. Expected number of words reviewed at day t

The first two measure the efficiency of the system. The third shows the distribution of the workload. Below each of these performance measures is outlined.

4.1.1 Mean number of cycles until all words are mastered

Students often would like to know how long it takes until they have mastered all their words. The state in which all words are mastered, $(0,(0,0),\ldots,(0,0),w)$, is the only absorbing state in the Markov process. Fortunately, absorption in $(0,(0,0),\ldots,(0,0),w)$ is thus certain. We are interested in the mean time to absorption. In Section 2.1, a common method to calculate the time to absorption for a time-homogeneous Markov chain is described. To be able to use this method, we use the time-homogeneous matrix $\mathbf{P} = P_1 P_2 \cdots P_T$. As a result of this, the mean time until all words are mastered is only calculated in cycles instead of individual days. However, it will still be a good indication of how efficient the system is, especially since there is only one fixed day per cycle on which the highest deck is reviewed and thus on which words can become mastered. The mean number of cycles until all words are mastered will be indicated by τ and is calculated with

$$\tau = \boldsymbol{\alpha_0} F \mathbf{1}.$$

4.1.2 Expected number of words mastered after t days

The expected number of words mastered after a certain day t, which will be indicated by m_t, does not say that much about a system on it self. However, it is a good measure to compare variants of the Leitner system to the original system and to each other. If system A has a higher m_t than system B, system A is preferred over system B when the rest is the same. One may wonder what the benefit of this measure in addition to the first measure is. For the first measure, the inverse of a matrix has to be taken. Moreover, contrary to the first measure, the second measure scales linearly with the number of words in the system as will be explained in Section 4.3.1. Both reasons make it feasible to analyse bigger systems with this second measure.

To calculate m_t, we need the matrix $P^{(t)}$, as defined in Section 2.1. Again, we are only interested in the row corresponding to the initial state $(w,(0,0),\ldots,(0,0),0)$. The vector $\boldsymbol{\alpha_0} P^{(t)}$ contains for each state the probability of being in that state after day t. Each state $\mathbf{n}$ contains the number of mastered words as the last entry of its state description, n_{m}. So, for each state the probability and the number of mastered words are known. Let the random variable M_t denote the number of words mastered after day t. The expected value of M_t can simply be calculated by multiplying the probability of being in a state $\mathbf{n}$ at time t, denoted by $[\boldsymbol{\alpha_0} P^{(t)}]_{\mathbf{n}}$, by n_{m} and summing over all states. That is,

$$m_t := E[M_t] = \textstyle\sum_{\mathbf{n}\in S} [\boldsymbol{\alpha_0} P^{(t)}]_{\mathbf{n}} n_{\mathrm{m}}.$$

Since m_t is an increasing function with t, the system and its adjustments will only be compared on the number of mastered words after one fixed day, which is chosen to be day 64. Day 64 is the last day of the fourth cycle, so possible phenomena specific to the start up of the system should have disappeared and the review of deck 5 of the fourth cycle has certainly taken place.

4.1.3 Expected number of words reviewed at day t

To determine the expected number of words reviewed at a certain day t, which will be indicated by r_t, we need to know in which state we are at the start of that day and which decks have to be reviewed on that day. Since all words in a deck, which has to be reviewed on that day, are reviewed once and only once on that day, no information about the correctness of the review of the words on day t itself is necessary. The probability of being in a certain state at the start of day t is the same as the probability of being in that state on day $t-1$, when started in the initial state $(w,(0,0),\dots,(0,0),0)$ on day 1. So, we need the vector $\boldsymbol{\alpha_0} P^{(t-1)}$. Let $\mathbf{d}_t$ be the vector containing the numbers of the decks that have to be reviewed on day t. For each state we can multiply the number of words initially in the decks $\mathbf{d}_t$, n_k for $k \in \mathbf{d}_t$, by the probability of being in that state at the start of day t, which can be found in $\boldsymbol{\alpha_0} P^{(t-1)}$. States that are not possible after day $t-1$ (and empty decks) will automatically have a zero contribution, so we can just sum over all states. Let the random variable $R_t(k)$ be the number of words in deck k at time t. So, the expected number of words reviewed at day t is calculated with

$$r_t := \sum_{k\in\mathbf{d}_t} E[R_t(k)] = \sum_{k\in\mathbf{d}_t} \sum_{\mathbf{n}\in S} [\boldsymbol{\alpha_0} P^{(t-1)}]_{\mathbf{n}} n_k.$$

This expected number r_t can be used to analyze the distribution of the workload by plotting the expected number of words reviewed per day during a cycle. Moreover, r_t can be used to calculate another useful measure for the distribution of workload: the ratio between the lowest and highest expected number of words reviewed during one cycle, which will be indicated by ρ and can calculated as follows

$$\rho := \frac{\max_{t=1+mT,\dots,T+mT} r_t}{\min_{t=1+mT,\dots,T+mT} r_t} \text{ for some } m \text{ large enough.}$$

4.2 Results for the Leitner system

All results are obtained by computations in the programming language Python. An interested reader can contact the author for the Python code used to obtain the results. Due to limitations in the size of a matrix, at most 5 words in the system could be examined with the current implementation. The analysis of the Leitner system below is given for both $w = 5$ and $w = 1$ in the system. The reason for the latter is explained in the section below. To obtain numerical results, a value for the global difficulty parameter θ in Equation (3) has to be chosen. All the results in this paper are obtained with $d = 5$ and $\theta = 0.3$, except when stated differently. This choice for θ is explained in Section 4.3.3.

The numerical outcomes for the three performance measures are given in Table 1. To make the reader familiar with the interpretation of these numerical values, the outcomes for the system with $w = 5$ will also be described verbally.

Table 1: The numerical values for the three performance measures for both five words and one word in the system.

# words in the system	τ	m_{64}	ρ
5	6.21	3.84	20.81 (day 8* and 12*)
1	3.59	0.77	20.81 (day 8* and 12*)

A student is expected to need 6.21 cycle to master all words in the system (τ). A word can only become mastered if it is reviewed correctly in deck 5. Since deck 5 is reviewed once in a cycle and only nonempty from cycle two on, the seemingly high number of cycles necessary to master all words can be put into context. After 64 days, the student is expected to have mastered 3.84 words of the total of 5 words (m_{64}). The biggest difference in the expected number of words reviewed is between day 8* and 12*. On day 12* the student is expected to review 20.81 times as many words as on day 8* of the same cycle m, m sufficiently large (ρ).

The number of words reviewed per day for the first seven learning cycles is plotted in Figure 3. For convenience, the first day of every learning cycle is marked on the x-axis. It is clear that there is a repeating pattern in the number of words reviewed per day in every learning cycle. This repeating pattern decreases in size, as was expected since every cycle some words leave the systems as they are mastered. The expected number of words reviewed per day fluctuates quite a lot. The least reviews per day take place on days on which only one deck is reviewed, while the most reviews happen on days on which higher decks are reviewed. This makes sense as there are multiple reviews of low decks before one higher deck is reviewed, so there are more 'chances' for words to arrive in the higher deck in the meantime.

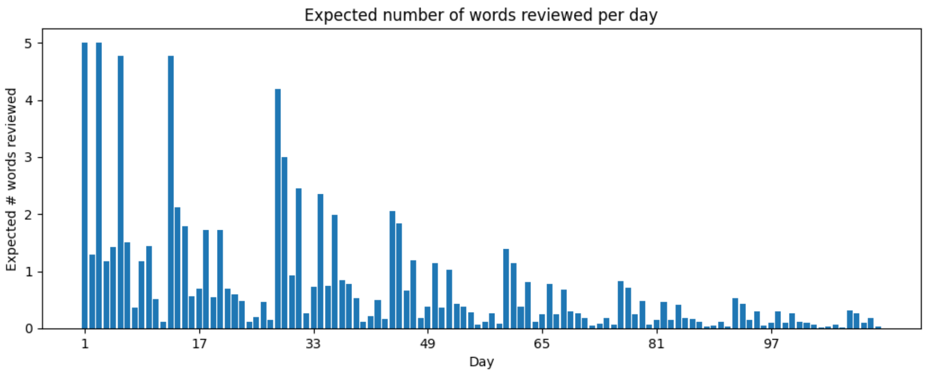


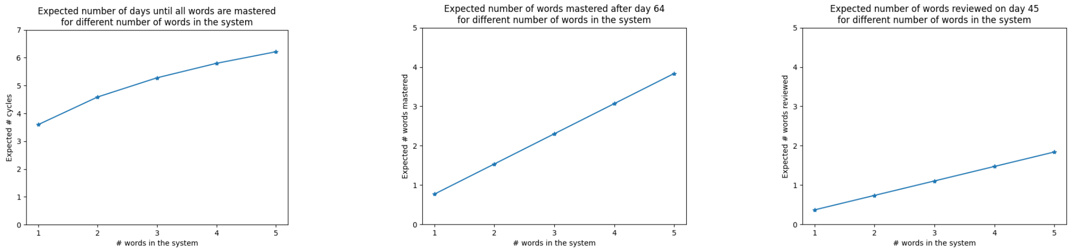
Figure 3: The expected number of words reviewed for each day of the first seven cycles.

4.3 Observations

4.3.1 m_t and r_t scale linearly with the number of words

Figure 4b and Figure 4c show that the second and third performance measures scale linearly with w. Intuitively this makes sense as the distribution of words among the days remains the same due to the assumption of independent recalls. So, a system with w words behaves for these two measures as w independent systems of one word.

Figure 4a shows that, unlike m_t and r_t, the first measure τ does not scale linearly with the number of words. This is explained by the fact that the day that all w words are mastered is the maximum of w days on which a single word in the system was mastered.



(A) The expected number of cycles until all words are mastered.

(B) The expected number of words mastered after day 64.

(C) The expected number of words reviewed on day 45.

FIGURE 4: The performances measures plotted against the number of words in the system.

The linearity of m_t and r_t ensures that $w = 1$ suffices for the computations for the comparison of proposed alternatives. From now on, all results are therefore obtained with $w = 1$, unless stated otherwise. This decreases the size of the state space, and thus of the transition matrices, drastically. Problems like limited memory storage for big matrices and long computing times are therefore avoided.

4.3.2 Fixed ratios for r_{y^*}

Figure 3 might raise two questions. Namely (i) if there is a factor in the decrease of the number of words reviewed on day t of every cycle and (ii) if the ratio between the number of words reviewed on two days of a single cycle is fixed. That is,

(i) $\frac{r_{y+mT}}{r_{y+(m+1)T}} \to \text{constant}$ as m grows large, $\forall 0 < y \leq T$

(ii) $\frac{r_{y+mT}}{r_{x+mT}} \to \text{constant}$ as m grows large, $\forall 0 < x, y \leq T$.

Table 2 and Table 3 show that both (i) and (ii) are the case for m big enough, so after the influence of the initial behaviour disappeared. (i) As the shape of Figure 3 already suggests and Table 2 confirms, r_{y^*} shows asymptotic geometric decay. (ii) As Table 3 indicates, the ratio between the r_{13*} and r_{14*} of the same cycle stabilizes around the value 2.91. The ratio between other combinations of days stabilizes at approximately the same rate. For example the ratio of day 5* and 6* is also given in Table 3.

If a margin of 0.01 between two consecutive cycles is taken, then one can say that this system stabilizes for (ii) after the sixth cycle. We will therefore call the sixth cycle and higher *steady cycles*. This enables us to study the distribution of workload over the days for only one cycle, since the ratios are almost fixed. This distribution can now be characterized by a plot of the expected number of words reviewed per day for only one steady cycle, like Figure 5. This fixed ratio also guarantees that the ratio between the lowest and highest expected number of words reviewed during one (steady) cycle, ρ, is a good measure.

Table 2: The ratio of the expected number of words reviewed on day 13 and on day 5 for the first 10 consecutive cycles.

$\mathbf{m}$	$\frac{\mathbf{r_{13+mT}}}{\mathbf{r_{13+(m+1)T}}}$	$\frac{\mathbf{r_{5+mT}}}{\mathbf{r_{5+(m+1)T}}}$
0	0.6270	0.4892
1	0.6149	1.2102
2	0.6186	0.5110
3	0.6174	0.6640
4	0.6178	0.6023
5	0.6177	0.6234
6	0.6177	0.6157
7	0.6177	0.6185
8	0.6177	0.6175
9	0.6177	0.6178

Table 3: The ratio of the expected number of words reviewed on day 13 and 14 of the same cycle and on day 5 and 6, for the first ten cycles.

$\mathbf{m}$	$\frac{\mathbf{r_{13+mT}}}{\mathbf{r_{14+mT}}}$	$\frac{\mathbf{r_{5+mT}}}{\mathbf{r_{6+mT}}}$
0	2.2580	0.2995
1	3.2396	1.1750
2	2.8128	1.0880
3	2.9515	1.1214
4	2.9003	1.1085
5	2.9184	1.1129
6	2.9119	1.1113
7	2.9142	1.1119
8	2.9134	1.1117
9	2.9137	1.1118

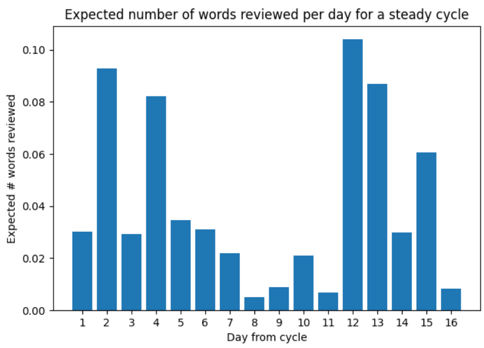


Figure 5: The expected number of words reviewed per day for a steady cycle (cycle 6).

4.3.3 Influence of the value of the global difficulty θ

As said before, the value of θ depends on several factors. Nevertheless, to generate numerical results it is necessary to assume a value for θ. A first idea would be to use the $\theta = 0.0077$ from the paper on which the recall probability in this research is based, [10].

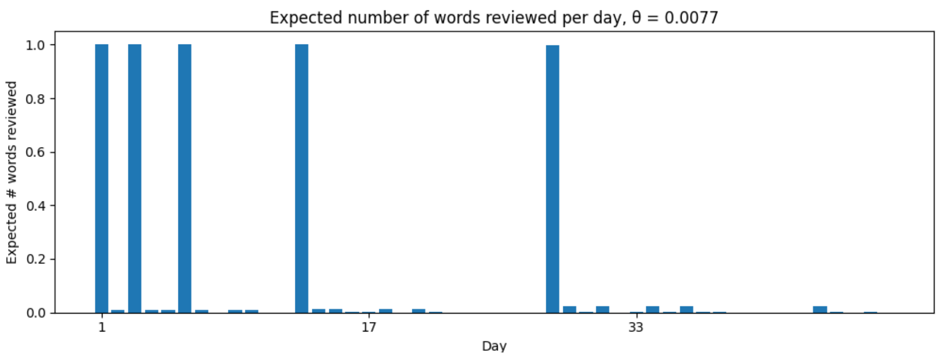
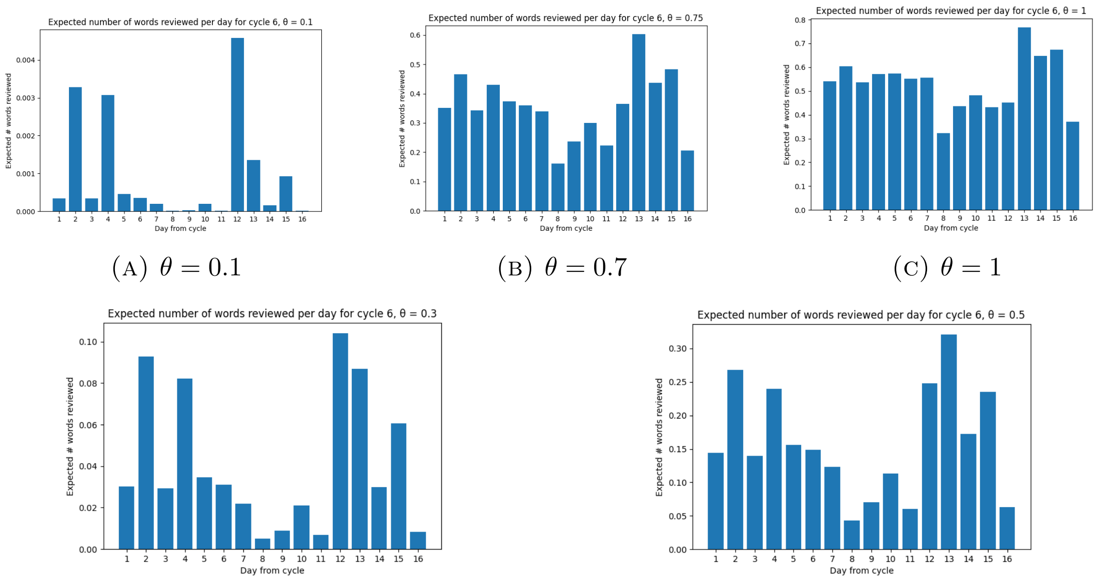


Figure 6: The expected number of words reviewed per day for $\theta = 0.0077$.

Figure 6 shows the expected number of words reviewed per day for $\theta = 0.0077$. This looks quite different from Figure 3 and not as an interesting system to examine. Moreover, the number of cycles to master all words τ is 2.02. Given that only after two cycles a word could be mastered, this is quite fast. These results are explained by the fact that such a low θ results in a very high recall probability, and consequently most words are reviewed correctly in the first (non-empty) review of each deck. These take place on day 1, 3, 6, 13 and 28 for deck 1 to 5 respectively, and therefore explain the five high peaks in Figure 6. The low peaks are caused by an incorrectly reviewed word which has to go back to deck 1 from where it follows a similar path to being mastered. So, with $\theta = 0.0077$, we are looking more at the limiting behaviour of the Leitner system than its use in a normal learning setting. This also explains the unrealistically high value of ρ.

It is therefore preferred to look at other values for θ. For different values of θ, the number of expected words reviewed per day in the sixth cycle is plotted in Figure 7 and the expected number m_{64} and the ratio ρ are given in Table 4.



(A) $\theta = 0.1$ (B) $\theta = 0.7$ (C) $\theta = 1$

(D) $\theta = 0.3$ (E) $\theta = 0.5$

FIGURE 7: The expected number of words reviewed per day in the sixth cycle for different values of θ.

In the graph for $\theta = 0.1$, the limiting behaviour observed with $\theta = 0.0077$ can easily be recognized. On the other hand, for $\theta = 1$ the limiting behaviour, to which the graph for $\theta = 0.7$ already tends, seems to occur as well. The graphs for $\theta = 0.3$ and $\theta = 0.5$, which are slightly bigger depicted in Figure 7, look more balanced. In the absence of reference values for the performance measures, we have to resort to logical reasoning to select a value for θ. The ratio ρ belonging to $\theta = 0.5$ feels more realistic than the one belonging to $\theta = 0.3$ when keeping in mind that Leitner seems to have thought carefully about the motivational aspect for the students. However, at the same time m_{64} for $\theta = 0.3$ feels more accurate, as one would imagine that after 64 days, a student should be able to learn three quarters of the words he has to learn instead of only half. Though this depends strongly on the language of the words as well. In this paper $\theta = 0.3$ was chosen. With this section, the reader is hopefully convinced that $\theta = 0.3$ is a decent choice, but more importantly he is now aware of what, and how big, the influence of θ is.

TABLE 4: m_{64} and ρ for different values of θ in the original system.

Value of θ	m_{64}	ρ
1	0.09	2.38 (day 8* and 13*)
0.75	0.22	3.75 (day 8* and 13*)
0.5	0.48	7.51 (day 8* and 13*)
0.3	0.77	20.81 (day 8* and 12*)
0.1	0.98	260.68 (day 8* and 12*)
0.0077	1.00	354772.23 (day 16* and 12*)

5 Adjustments to the Leitner system

Several possible adjustments to the Leitner system are proposed and analysed in this section. These adjustments can be divided into two categories. Some adjustments focus on achieving a more even distribution of workload. The objective of the other, more extreme, adjustments is to make the schedule easier to remember. The adjusted models are analysed using the second and third performance measures of Section 4.1 and compared to the original Leitner system. Since the second performance measure suffices to compare models on their efficiency and scales linearly with the number of words, as highlighted in Section 4.3.1, the first performance measure will not be taken into account due to computational difficulties. For the first two adjustments, the changes to the original model, as discussed in Section 3.5, are explained. The procedure for the other adjustments is similar and is left out to avoid focusing too much on technicalities. The reader can find all the results in Table 5 at the end of this section.

5.1 A more equal distribution of workload

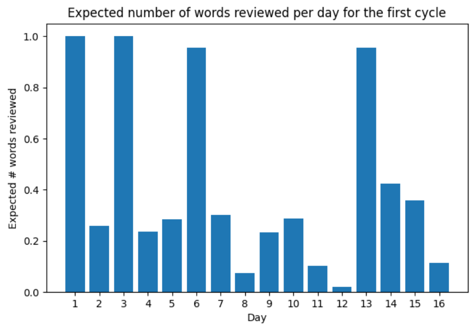
Figure 5 shows quite a difference in the expected number r_{y^*} for different values of y^*. While a student may have some extra motivation on the eighth day of each cycle since little words are expected to be reviewed, the student could be very discouraged on day 12* since he is expected to review a lot of words (20.81 times as many as on day 8*). It would therefore be convenient if r_{y^*} is approximately constant for y^*.

5.1.1 Introducing a day 0

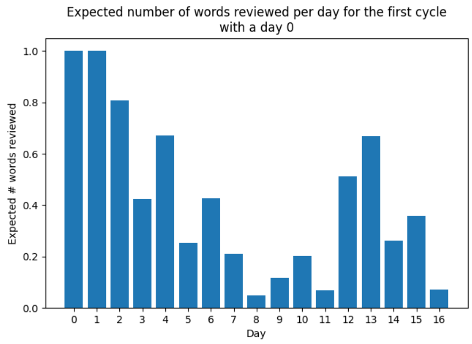
The number of words reviewed in the first cycle of the original Leitner system is given in Figure 8a. What immediately stands out are the peaks on day 1, 3 and 6, which are not present anymore in a steady cycle, shown in Figure 5. These extreme fluctuations in the first days are caused by the fact that the Leitner system 'just starts', as if you already have had a few review sessions. Deck 2 and 3 are certainly empty on day 1 and 2, respectively, so their scheduled review is useless, while all cards are still in deck 1 and 2 on day 3, causing a review of all words. The first non-empty review of deck 3 is only on day 6, causing another big peak in the number of reviews for that day. Only after these first extreme days, the words are spread out enough over the decks for the system to start working. A simple adjustment would therefore be to introduce a day 0 on which deck 1 is reviewed, which is only done when starting with the Leitner system. This would avoid scheduled decks being guaranteed empty.

The only change made to the original model is the introduction of the transition matrix P_0 which contains the transitions corresponding to reviewing deck 1. Since on day 8* and 16* also only deck 1 is reviewed, P_0 will equal P_{8s}. This means that instead of letting the Markov process start at $t = 1$, starting it at $t = 0$ would naturally turn out correctly.

With the introduction of a day 0 the extreme fluctuations during the first days disappeared, as can be seen in Figure 8b. Due to the smoother start up of the system, it is now also possible for words to be in deck 5 of the system after 12 days. This increased the efficiency of the system, as can be seen by the fact that now $m_{64} = 0.84$, compared to $m_{64} = 0.77$ when not having a day zero.



(A) Without the introduction of a day 0.



(B) When a day 0 is introduced.

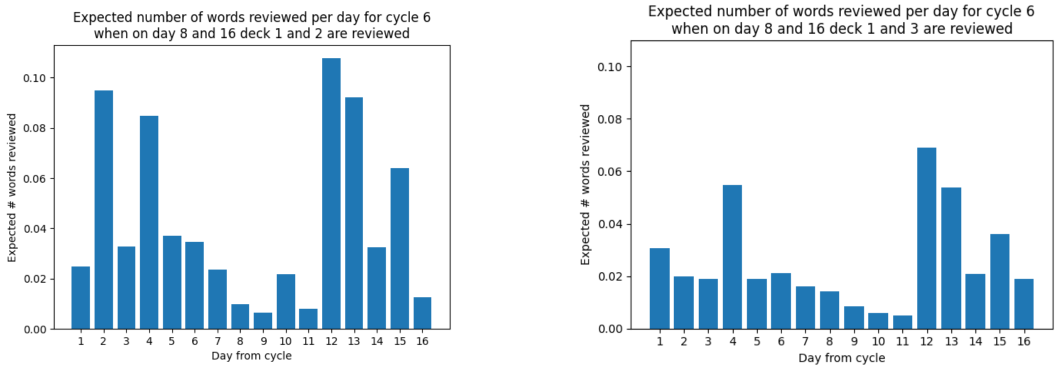
FIGURE 8: The expected number of words reviewed per day for the first cycle.

5.1.2 Adding an extra deck

Figure 5 shows that the days with the least expected number of words reviewed are mostly day 8* and 16*. Not surprisingly, these are the only two days on which only one deck, deck 1, is reviewed. To increase r_{8^*} and r_{16^*}, the review of an extra deck could be added. Both the individual adding of deck 2 and deck 3 are studied. Note that this adjustment causes the total workload to increase a bit, since per cycle an extra deck is reviewed, instead of flattening out the existing workload. If this would, however, pay off in having mastered the words notably sooner, it would be a reasonable trade off.

Due to the adding of deck 2 (deck 3), an extra arrival of words to deck 3 (deck 4) takes place. This means that an extra entry $n_{3,\text{middle}}$ ($n_{4,\text{middle}}$) should be added to the vector ($n_{3,\text{old}}$, $n_{3,\text{new}}$) (($n_{4,\text{old}}$, $n_{4,\text{new}}$)) in the state description. All transition matrices are adjusted accordingly, where only on day 8* and 16* words can be added to this middle state. Naturally $\mathbf{d}_{8^*}$ changes to [1,2] ([1,3]) instead of [1] and $\mathbf{t}_3$ ($\mathbf{t}_4$) gets an extra entry indicating the number of days from the batch of words arrived to the 'middle' subdeck until the review of deck 3 (deck 4).

The influence of adding deck 2 on day 8* and 16* on the distribution of words reviewed per day can be seen in Figure 9a. As expected, r_{8^*} and r_{16^*} increased, while r_{9^*} decreased a bit. However, r_{9^*} is still higher than r_{8^*} of the original system. Therefore, the ratio ρ decreased to 16.37 (between day 12* and 9*). Although an extra deck is reviewed per cycle, the expected number of mastered words m_{64} decreased slightly to 0.76. This could be explained by the fact that the words, which now arrive to deck 3 two days before their review, would in the original system only arrive one day before. This means that in deck 3 the recall probability for these words is now lower than in the original system, so fewer words are expected to go to deck 4.



(A) On day 8* and 16* deck 1 and 2 are reviewed instead of only deck 1.

(B) On day 8* and 16* deck 1 and 3 are reviewed instead of only deck 1.

FIGURE 9: The expected number of words reviewed per day for certain adjustments.

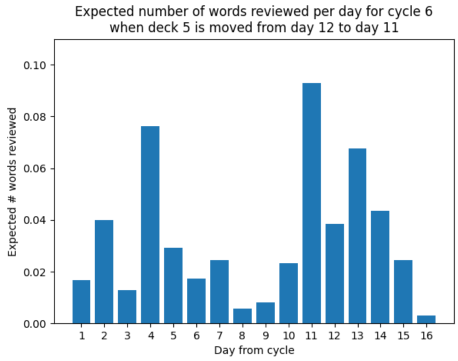
Adding deck 3 on day 8* and 16* altered the distribution of workload as shown in Figure 9b and decreased the ratio ρ to 13.95 (between day 11* and 12*). Moreover, m_{64} increased to 0.82. Although both performance measures seemed to improve from adding deck 3, an important side note is made in Section 6.1.

5.1.3 Moving deck 4 and 5

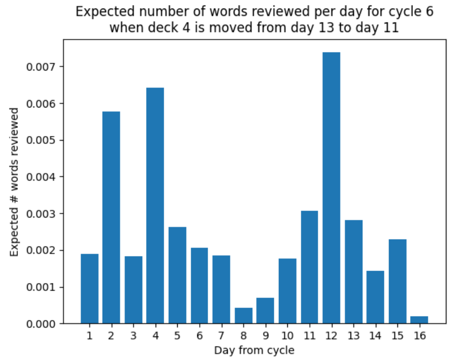
The previous adjustment focused on equalizing the workload by increasing r_t on the quietest days. Another approach to reach this goal would be to decrease r_t of one of the busiest days. Two possibilities are studied. The first one focuses on the peak of day 12*, while the second one focuses on the peak of day 13*.

Figure 5 shows a big difference between r_{11^*} and r_{12^*}. The influence of moving deck 5 from day 12* to day 11* is therefore examined. Figure 10a shows that the total distribution improves a bit, although the biggest contribution of the high peak is the review of deck 5 itself, which can not be flattened out with the current approach. The words that are now reviewed incorrectly in deck 5 will already be reviewed on day 12*, on which only deck 1 is reviewed, instead of contributing to a higher number of reviews on day 13* like in the original schedule. This causes the peak of day 13* to be lowered and the dip of original day 11* to be heightened, now on day 12*. This slight flattening of the distribution of the workload is not reflected in the ratio ρ though. Since the expected number of reviews on day 16* decreased, ρ increased to 31.09. The expected number of mastered word m_{64} increased slightly to 0.80.

Day 13* is the only day on which three decks are reviewed. This would be a logical explanation for the high peak on this day. During this research it was examined what happened when deck 4 is moved from day 13* to day 11*. Afterwards it does not seem like a logical choice: one would expect that it is like only switching day 11* and 13* instead of flattening out the workload. However, the results are quite remarkable and show an important pitfall of this research. Figure 10b shows that, as expected, the peak on day 12* did not change. The peak of day 13* did not move to day 11* but just disappeared, which was not as expected. A slight decrease on day 16* caused the ratio ρ to increase to 36.35. The most interesting observation is the enormous increase in which words are mastered. The number on the x-axis of Figure 10b already suggests it and the drastic increase of m_{64} to 0.97 confirms this. The reason for this increase is quite simple. By moving deck 4



(A) The review from deck 5 is moved from day 12* to day 11*.



(B) The review from deck 4 is moved from day 13* to day 11*.

FIGURE 10: The expected number of words reviewed per day for certain adjustments.

from day 13* to day 11*, deck 4 is now reviewed the day before deck 5 instead of the day after. This means that words arriving to deck 5 were last reviewed only one (and eight) day before instead of fifteen (and eight) days. This increases the recall probability for these words significantly, causing more words to be reviewed correctly in deck 5. Although this faster mastering of words sounds nice, the influence on the retention of the words in the long run is unknown.

5.1.4 Spread introduction of new word

On the first days the expected number r_t is the highest, since it is assumed that all words start in deck 1 on day 1. When the introduction of new words is spread (evenly) over the first four days, the graph of r_t for one steady cycle looks identical to Figure 5, except for the scale of the y-axis. Since the states have only integer entries and at least one word is introduced per day, $w = 4$ is needed to obtain results. The expected number m_{64} of 3.04 should therefore be divided by 4 before comparing it to the original Leitner system. Both ρ and the scaled m_{64} are almost the same as those of the original system, just like the graph of r_t. This illustrates that the behaviour in the long term is not influenced much.

5.2 An easier to remember schedule

Most people, and especially students, tend to think in weeks: they start a week on Monday and end with one or two days of weekend. For remembering the Leitner calendar, it would therefore be convenient to adjust the calendar such that the learning cycles are based on weeks. However, the previously described adjustments show that caution is necessary when deviating too much from the original Leitner system.

The original Leitner system has a learning cycle of 16 days for 5 decks, where the first and second 8 days are quite similar. So, we will focus on a variant that takes two weeks (14 days), where the first and second weeks are quite similar. Since the long-term learning effects of deviating too much from the original division of decks are unknown, the choice was made not to set up an own division of decks over the 14 days, but adjust the original Leitner system by removing day 8* and day 16*, on which only deck 1 would have been reviewed. For clarity this adjusted review calendar is given in Figure 11.

The new distribution of workload is shown in Figure 12. The distribution is different

Day	Decks					Day	Decks				
1*	1	2				8*	1	2			
2*	1		3			9*	1		3		
3*	1	2				10*	1	2			
4*	1					11*	1				5
5*	1	2		4		12*	1	2		4	
6*	1		3			13*	1		3		
7*	1	2				14*	1	2			

FIGURE 11: The review calendar for a biweekly system.

from the original system, as was shown in Figure 5, especially for the lower days, but does not appear to have worsened. Based on the ratio ρ, which decreased significantly to 12.28, the difference between the review sessions is less extreme, so one could say that the distribution of workload improved. The number of mastered words m_{64} did not change much.

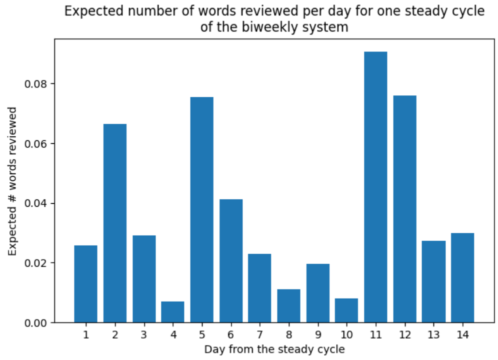


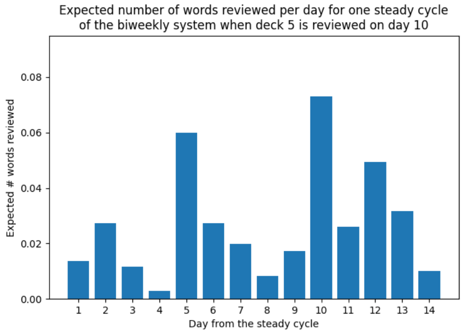
FIGURE 12: The expected number of words reviewed per day for a steady cycle (cycle 6) of the biweekly system.

5.3 Proposed alternative system

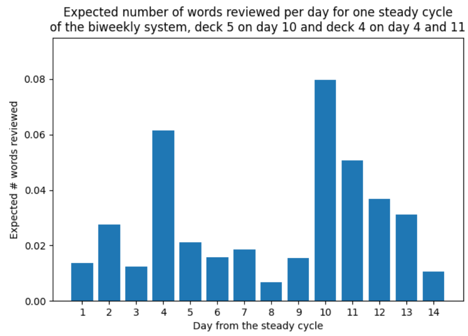
Several possible adjustments have been analysed in the previous subsections. In this subsection, the obtained knowledge is used to propose an alternative to the Leitner system that would be more user friendly, while at the same time maintaining the usefulness of spaced repetition. One of the biggest drawbacks of the Leitner system is that it is hard to remember. Therefore, the basis for the proposed alternative will be the biweekly system as discussed in Section 5.2. A first simple improvement will be the addition of a day 0. Although the adding of deck 3 on the original days 8* and 16*, the days on which only deck 1 is reviewed, showed an improvement in both performance measures, no deck will be added to the day on which only deck 1 is reviewed (day 4*). The reason for this is that the influence on the retention in the long run is unknown and feared to be worse, as will be explained in Section 6.1.

The biweekly system has its two largest expected review sessions on the day 11* and 12* which are similar to day 12* and 13* of the original system. Moving deck 5 to a day earlier showed a slight increase in the expected number of words mastered, as described in Section 5.1.3. Since deck 5 is reviewed only once per cycle, the moving of deck 5 does not affect the easy-to-remember structure of the biweekly system. This adjustment of the biweekly system, moving deck 5 from day 11* to day 10*, had indeed a similar effect as on the original Leitner system: m_{26} increased to 0.83 while also ρ increased strongly due to a decrease in expected number of reviews on the day on which only deck 1 is reviewed.

Nevertheless, Figure 13a shows that the distribution of workload over the other days seems to have slightly improved. The high ratio ρ is mostly caused by the low value of r_{4^*}. Since r_{5^*} is quite high, deck 4 is moved from day 5* to day 4*. Due to the aspiration for an easy-to-remember schedule, this means that deck 4 is moved from day 12* to day 11* as well. This adjustment caused a strong decrease in the ratio ρ from 26.22 to 11.54 while m_{64} did not change much. The total distribution of workload can be found in Figure 13b.



(A) Deck 5 is moved from day 11* to day 10*.



(B) Deck 5 is moved from day 11* to day 10* and deck 4 is moved from day 12* and 5* to day 11* and 4*, resp. (Proposed alternative)

FIGURE 13: The expected number of words reviewed per day for certain adjustments to the biweekly system.

The biggest contribution to the highest peaks in Figure 13b is due to the review of a single deck. These peaks seem therefore unavoidable with the chosen approach of only adjusting the original Leitner system. No improvements to the system adjusted as described above were found and therefore this is the proposed alternative to the Leitner system based on this research. The review calendar of this proposed alternative is given in Figure 14.

Day	Decks					Day	Decks				
1*	1	2				8*	1	2			
2*	1		3			9*	1		3		
3*	1	2				10*	1	2			5
4*	1			4		11*	1			4	
5*	1	2				12*	1	2			
6*	1		3			13*	1		3		
7*	1	2				14*	1	2			

FIGURE 14: The review calendar for the proposed alternative system.

The proposed alternative performs better on both performance measures and is easier to remember. However, the biweekly system (14 days instead of 16 days) and the moving of deck 4 decreased the time since last reviewed for the words, which makes it hard to judge the influence of the proposed alternative on the long term retention.

TABLE 5: The second and third performance measure for adjustments to the system (and the reference values of the original system).

Adjustment	**Graph of $\mathbf{r_{y^*}}$**	$\boldsymbol{m_{64}}$	$\boldsymbol{\rho}$
Original Leitner system	Figure 5	0.77	20.81 (day 8* and 12*)
Introduction of day 0	Figure 8b	0.84	20.80 (day 8* and 12*)
Adding deck 2 on day 8* and 16*	Figure 9a	0.76	16.37 (day 9* and 12*)
Adding deck 3 on day 8* and 16*	Figure 9b	0.82	13.95 (day 11* and 12*)
Deck 5 from day 12* to day 11*	Figure 10a	0.80	31.09 (day 16* and 11*)
Deck 4 from day 13* to day 11*	Figure 10b	0.98	36.35 (day 16* and 12*)
Spread introduction of words	-	~0.76	20.81 (day 8* en 12*)
Biweekly	Figure 12	0.79	12.28 (day 4* and 11*)
Biweekly: deck 5 to day 10*	Figure 13a	0.83	26.22 (day 4* and 10*)
Proposed alternative	Figure 13b	0.82	11.45 (day 8* and 10*)

6 Discussion

First, a small remark on the choice of state description has to be made. One may wonder if the time-inhomogeneity of the Markov process could be avoided by a different state description that also contains the decks that are reviewed 'today'. If the order of decks that have to be reviewed would have been the same, this would probably have worked. However, knowing which decks are reviewed 'today' does not uniquely determine which decks have to be reviewed 'tomorrow'. For example, both at day 3* and 5* deck 1 and 2 have to be reviewed, but the next day deck 1 and 4 or deck 1 and 3 have to be reviewed, respectively.

Another remark is about the choice to explicitly state the transition matrices relating to five decks while the original Leitner calendar, like in Figure 2, has seven decks. An attempt was made to state the transition matrices more generally by making use of the vector $\mathbf{d_t}$. Although this general formulation is possible for the naive model, the distinction between 'old' and 'new' subdecks made a general formulation of the transition matrix for the more realistic model, as described in Section 3.5, too cumbersome. Therefore the choice was made to facilitate a better understanding of the transition matrices by explicitly stating the transition matrices for $d = 5$, at the expense of a general formulation for d decks. Five decks sufficed to analyse the working of the Leitner system and to see the consequences of adjustments to the Leitner system. Adding the two 'missing' decks would probably result in a shift in days in the distribution of words reviewed (and mastered) per day, but would not help to gain more insight. Conversely, it would cause more problems with computing the transition matrices since the state space, and therefore the matrix size, increases drastically with d. Since there were now already inconveniences with the size of the matrices, as can be read below, $d = 5$ was preferred over $d = 7$ in this research. Moreover, our motivation for studying the Leitner systems was the use of physical flashcards by high school students. Since only one cycle of seven decks already takes 64 days, one may argue that a learning cycle of 16 days ($d = 5$) would more often be used by high school students than a 64 days learning cycle ($d = 7$).

In this research, no systematic approach was used to optimize the Leitner system.

Based on Figure 3 and Figure 5 flaws of the system were determined and ways to improve these by adjusting the system, as explained at the start of each subsection of Section 5.1, were examined. A different approach would be to choose one adjustment and determine the optimal execution of this adjustment or to select one parameter and determine the optimum for this parameter.

6.1 Limitations and further research

A problem encountered in this research was that the transition matrices quickly became too big to set up, let alone perform operations with, due to memory capacity. Due to this, only a system with at most five words could be examined. However, due to the linearity of the second and third performance measure with the number of words in the system and the fact that most analyses were based on comparison of systems, meaning that $w = 1$ could be used, the consequence of this computational problem was small. It did however cause the first performance measure to be less useful than initially thought. Since the transition matrices are sparse matrices, the problem could be circumvented by storing the nonzero indices differently. It was decided not to focus on this since the current method sufficed for this research as explained before. It might still be a topic for further study.

A Markov process is determined strongly by the chosen transition probabilities. This means that the choice for a different recall probability could give quite different results. For example, a recall probability that has a bigger contribution of the time since last reviewed or a word-specific complexity parameter. The influence of the choice of memory model [16] for recall probability could be examined further. Furthermore, Section 4.3.3 already showed the big impact of the choice of θ on the outcome of the model. As said, there were no good reference values available and an experimental study on which value of θ would be most realistic was not in the scope of this research. This would be an interesting topic for further research.

An important pitfall of this research is that it is tempting to conclude that it is always positive for the effectiveness of the system to repeat a word as soon as possible since the expected number of words mastered m_{64} would increase. This pitfall is caused by the inability to see the performance of the student in the long run. Psychological research, like [1], [2] and [13], shows that reviewing a word too soon means that it is still in the student's short-term memory, which means that the long-term retention will not improve. So, you have to almost 'forget' words to really learn them in the long run. Furthermore, the optimal time between reviews depends on how long you want to remember the words.

This psychological phenomenon makes it hard to really judge the proposed systems. An adjustment to the original system could improve the distribution of workload by moving the decks to different days, while at the same time also increasing the number of words mastered after a fixed day, but this does not mean that in the long run this is the best system. An extreme example of this was the shifting of deck 4 from day 13* to day 11*, as described in Section 5.1.3. This shifting caused words to arrive at deck 4 only one day before their review in deck 4, meaning that they had a significantly higher recall probability than in the original Leitner system, causing a higher number of words mastered after a fixed day. However, it would not seem unlikely that students using this adjusted system would perform less in retention in the long run. To a lesser extent, this could also be the case with the adding of deck 3 on day 8* and 16* and with the proposed alternative system, which showed higher effectiveness than the original Leitner schedule with the second performance measure and a better distribution of workload with the third performance measure. A learning cycle of 16 days of the original Leitner system is in the proposed alternative put into only 14 days and the number of days between the reviews of deck 3 and 4 is decreased by one. These on average shorter times between reviews can cause a better performance in effectiveness on the short term, while possibly performing worse on the retention in the long run, by the phenomenon described above. Further research into the effects of the proposed adjustments in the long run would therefore be necessary to be able to judge the suggested alternatives properly. An example of such research would be to have people studying by such a scheme and testing them a few months later.

7 Conclusion

In this paper, the Leitner system was modelled as a discrete-time inhomogeneous Markov process in Section 3, and analysed using three performance measures, introduced in Section 4.1. In Section 4.3, the relation between the number of words in the system and both the number of words mastered after as well as reviewed on day t was reasoned to be linear. This enabled us to compute the results for one word in the system, by which the problem of limited memory capacity was circumvented while the results remained useful. Moreover, the ratio between the number of words reviewed on two days of a cycle was shown to approach a fixed value, making the maximum ratio ρ a valid performance measure. Furthermore, the influence of the global difficulty θ on the results was examined. The choice of $\theta = 0.3$ was shown to be well-grounded, but more importantly the reader was made aware of what, and how big, the influence of θ is.

In Section 5 several adjustments to the Leitner system were proposed to make it more user friendly, both in equalizing the distribution of workload over days and in adjusting it to weekly cycles. Most adjustments to the review schedule showed an improvement in one or even all performance measures. However, the influence of these adaptations on the retention in the long run is unknown but feared to be worse.

The biweekly system was adapted using the adjustments of Section 5.1 that performed well, to serve as an alternative to the Leitner system. The proposed alternative, as given in Figure 11, is easier to remember than the original Leitner system due to its learning cycle of two weeks. It also showed a better distribution of workload, based on Figure 13b and a significantly lower ρ, than the Leitner system. At the same time, the efficiency as measured by m_{64} was maintained and even slightly improved. Therefore, the proposed alternative answers the research question posed in Section 1. However, Figure 13b still showed quite a difference in the expected number of words studied per day, which could not be equalized with the approach used in this research.

So, the suggested alternative performed better than the original Leitner system in the mathematical analysis. However, the change to a biweekly system (14 days instead of 16 days) and the moving of deck 4 decreased the time since last review of the words, which makes it hard to judge the influence of the proposed alternative on the long term retention. Further research into the effects of the proposed adjustments in the long run would therefore be necessary to be able to judge the suggested alternatives properly. Another important topic for further research would be the influence of the choice of the recall probability and its parameter values on the modelled behaviour of the Leitner system. A systematic approach to optimize (some part of) the Leitner system could also serve as a topic for further research.

References

[1] B. Carey. *How We Learn: The Surprising Truth About When, Where, and Why It Happens.* Random House, 2014.

[2] N. Cowan. What are the differences between long-term, short-term, and working memory? *Progress in Brain Research*, 169:323–338, 2008.

[3] H. Ebbinghaus. *Über das Gedächtnis.* Wissenschaftl. Buchgesell., 1885. trans Ruger HA, Bussenius CE (1913) [Memory: A Contribution to Experimental Psychology] (Columbia Univ Teachers College, New York).

[4] J. Karpicke and A. Bauernschmidt. Spaced retrieval: Absolute spacing enhances learning regardless of relative spacing. *Journal of experimental psychology. Learning, memory, and cognition*, 37:1250–7, 2011.

[5] John G. Kemeny and J.Lauri Snell. *Finite Markov Chains.* Springer-Verlag, 1976.

[6] S. Leitner. *So lernt man lernen: angewandte Lernpsychologie - ein Weg zum Erfolg.* Herder, 7th edition, 1972.

[7] E. Mettler, C. Massey, and P. Kellman. A comparison of adaptive and fixed schedules of practice. *Journal of experimental psychology. General*, 145, 2016.

[8] J. Murre and J. Dros. Replication and analysis of ebbinghaus' forgetting curve. *PloS one*, 10:e0120644, 2015.

[9] P. Pavlik Jr and J. Anderson. Using a model to compute the optimal schedule of practice. *Journal of experimental psychology. Applied*, 14:101–17, 2008.

[10] S. Reddy, I. Labutov, S. Banerjee, and T. Joachims. Unbounded human learning: Optimal scheduling for spaced repetition. In *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1815–1824, August 2016.

[11] S.M. Ross. *Introduction to Probability Models.* Academic Press, 9th edition, 2007.

[12] Teresa Scholz, Vitor Lopes, and Ana Estanqueiro. A cyclic time-dependent markov process to model daily patterns in wind turbine power production. *Energy*, 67:557–568, 04 2014.

[13] P. Smolen, Y. Zhang, and J. H. Byrne. The right time to learn: mechanisms and optimization of spaced learning. *Nature reviews. Neuroscience*, 17(2):77–88, 2016.

[14] B. Tabibian, U. Upadhyay, A. De, A. Zarezade, B. Schölkopf, and M. Gomez-Rodriguez. Enhancing human learning via spaced repetition optimization. In *Proceedings of the National Academy of Sciences.*

[15] W.L. Winston. *Operations Research, Applications and Algorithms.* Brooks/Cole - Thomson Learning, 4th edition, 2004.

[16] A. Zaidi, R. Caines, A.and Moore, P. Buttery, and A. Rice. Adaptive forgetting curves for spaced repetition language learning. In *Proceedings of Artificial Intelligence in Education: 21st International Conference Part II.*

A Overview of used symbols

TABLE 6: An overview of the important symbols and their description.

symbol	description
S	state space of Markov process
T	length of a learning cycle
$y*$	day y of a learning cycle, so all days t for which $t = y + mT, m = 0, 1, 2, 3, \ldots$
d	number of decks
w	number of words in the system
n_k	number of words in deck k of state $\mathbf{n}$
n_{m}	number of words mastered in state $\mathbf{n}$
c_k	number of correctly reviewed words in deck k
p_k	recall probability for deck k
$q_{n,c,p}$	probability of c correct recalls out of n words with recall probability p
θ	global difficulty
$\mathbf{t}_k$	vector containing the number of days since the batches of words in deck k are last reviewed
$\mathbf{d}_t$	vector containing the numbers of the decks that have to be reviewed on day t
τ	expected number of cycles until all words are mastered
m_t	expected number of words mastered after day t
r_t	expected number of words reviewed at day t
ρ	ratio between the lowest and highest expected number of words reviewed during one cycle

B Transition matrices

B.1 Naive Leitner model

$$\begin{aligned}[P_{2+4s}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} &= q_{n_1,c_1,p_1} * q_{n_3,c_3,p_3}, \quad \text{for } \boldsymbol{\delta} = (-c_1 + n_3 - c_3, c_1, -n_3, c_3, 0, 0) \\ &\quad \text{if } c_1 \leq n_1,\ c_3 \leq n_3 \\ &= 0 \quad \text{else}\end{aligned}$$

$$\begin{aligned}[P_{4^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} &= q_{n_1,c_1,p_1} * q_{n_4,c_4,p_4}, \quad \text{for } \boldsymbol{\delta} = (-c_1 + n_4 - c_4, c_1, 0, -n_4, c_4, 0) \\ &\quad \text{if } c_1 \leq n_1,\ c_4 \leq n_4 \\ &= 0 \quad \text{else}\end{aligned}$$

$$\begin{aligned}[P_{8s}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} &= q_{n_1,c_1,p_1}, \quad \text{for } \boldsymbol{\delta} = (-c_1, c_1, 0, 0, 0, 0) \ \text{ if } c_1 \leq n_1 \\ &= 0 \quad \text{else}\end{aligned}$$

$$\begin{aligned}[P_{12^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} &= q_{n_1,c_1,p_1} * q_{n_5,c_5,p_5}, \quad \text{for } \boldsymbol{\delta} = (-c_1 + n_5 - c_5, c_1, 0, 0, -n_5, c_5) \\ &\quad \text{if } c_1 \leq n_1,\ c_5 \leq n_5 \\ &= 0 \quad \text{else}\end{aligned}$$

$$[P_{13^*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} * q_{n_2,c_2,p_2} * q_{n_4,c_4,p_4}, \quad \text{for}$$
$$\boldsymbol{\delta} = (-c_1 + n_2 - c_2 + n_4 - c_4, -n_2 + c_1, c_2, -n_4, c_4, 0)$$
$$\text{if } c_1 \leq n_1,\ c_2 \leq n_2,\ c_4 \leq n_4$$
$$= 0 \quad \text{else}$$

B.2 Realistic Leitner model

$$[P_{2+8s}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} * q_{n_{3,\text{old}},c_{3,\text{old}},p_{3,\text{old}}} * q_{n_{3,\text{new}},c_{3,\text{new}},p_{3,\text{new}}} \quad \text{for}$$
$$\boldsymbol{\delta} = (-c_1 + n_{3,\text{old}} - c_{3,\text{old}} + n_{3,\text{new}} - c_{3,\text{new}}, (0, c_1), (-n_{3,\text{old}}, -n_{3,\text{new}}),$$
$$(0, c_{3,\text{old}} + c_{3,\text{new}}), (0, 0), 0),$$
$$\text{if } c_1 \leq n_1,\ c_{3,\text{old}} \leq n_{3,\text{old}},\ c_{3,\text{new}} \leq n_{3,\text{new}}$$
$$= 0 \quad \text{else.}$$

$$[P_{4*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} * q_{n_{4,\text{old}},c_{4,\text{old}},p_{4,\text{old}}} * q_{n_{4,\text{new}},c_{4,\text{new}},p_{4,\text{new}}} \quad \text{for}$$
$$\boldsymbol{\delta} = (-c_1 + n_{4,\text{old}} - c_{4,\text{old}} + n_{4,\text{new}} - c_{4,\text{new}}, (0, c_1), (0, 0),$$
$$(-n_{4,\text{old}}, -n_{4,\text{new}}), (0, c_{4,\text{old}} + c_{4,\text{new}}), 0),$$
$$\text{if } c_1 \leq n_1,\ c_{4,\text{old}} \leq n_{4,\text{old}},\ c_{4,\text{new}} \leq n_{4,\text{new}}$$
$$= 0 \quad \text{else.}$$

$$[P_{6+8s}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} * q_{n_{3,\text{old}},c_{3,\text{old}},p_{3,\text{old}}} * q_{n_{3,\text{new}},c_{3,\text{new}},p_{3,\text{new}}} \quad \text{for}$$
$$\boldsymbol{\delta} = (-c_1 + n_{3,\text{old}} - c_{3,\text{old}} + n_{3,\text{new}} - c_{3,\text{new}}, (0, c_1), (-n_{3,\text{old}}, -n_{3,\text{new}}),$$
$$(c_{3,\text{old}} + c_{3,\text{new}}, 0), (0, 0), 0),$$
$$\text{if } c_1 \leq n_1,\ c_{3,\text{old}} \leq n_{3,\text{old}},\ c_{3,\text{new}} \leq n_{3,\text{new}}$$
$$= 0 \quad \text{else.}$$

$$[P_{8*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} \quad \text{for } \boldsymbol{\delta} = (-c_1, (0, c_1), (0, 0), (0, 0), (0, 0), 0) \quad \text{if } c_1 \leq n_1$$
$$= 0 \quad \text{else.}$$

$$[P_{13*}]_{\mathbf{n},\mathbf{n}+\boldsymbol{\delta}} = q_{n_1,c_1,p_1} * q_{n_{2,\text{old}},c_{2,\text{old}},p_{2,\text{old}}} * q_{n_{2,\text{new}},c_{2,\text{new}},p_{2,\text{new}}} * q_{n_{4,\text{old}},c_{4,\text{old}},p_{4,\text{old}}} * q_{n_{4,\text{new}},c_{4,\text{new}},p_{4,\text{new}}}$$
$$\text{for } \boldsymbol{\delta} = (-c_1 + n_{2,\text{old}} - c_{2,\text{old}} + n_{2,\text{new}} - c_{2,\text{new}} + n_{4,\text{old}} - c_{4,\text{old}} + n_{4,\text{new}} - c_{4,\text{new}},$$
$$(-n_{2,\text{old}} + c_1, -n_{2,\text{new}}), (0, c_{2,\text{old}} + c_{2,\text{new}}), (-n_{4,\text{old}}, -n_{4,\text{new}}),$$
$$(c_{4,\text{old}} + c_{4,\text{new}}, 0), 0),$$
$$\text{if } c_1 \leq n_1,\ c_{2,\text{old}} \leq n_{2,\text{old}},\ c_{2,\text{new}} \leq n_{2,\text{new}},\ c_{4,\text{old}} \leq n_{4,\text{old}},\ c_{4,\text{new}} \leq n_{4,\text{new}}$$
$$= 0 \quad \text{else.}$$