Optimal Word Pair Learning in the Short Term: Using an Activation Based Spacing Model

Marnix van Woudenberg

s1313428

August 2008

Supervisors:

Dr. Hedderik van Rijn (Artificial Intelligence)

Dr. Fokie Cnossen (Artificial Intelligence)

Artificial Intelligence
University of Groningen

Abstract

What would you do if you only had 15 minutes to learn a list of word pairs for an exam tomorrow? What learning strategy would you use? And given the short amount of time to learn, does the strategy even matter? These are questions addressed in this thesis. The motivation for this project was to find an adaptive, optimal learning schedule, that is proven to work in a real-life setting.

Although a lot of work has been done on optimal learning paradigms, most of this research has focused on longer learning periods ($>$ 30 minutes), and longer retention intervals ($>$ 1 week). This is in stark contrast to informal reports on how students learn word pairs, which is more typically described as consisting of a single, shorter learning episode ($<$ 30 minutes) one day before a test.

To construct a optimal learning schedule, ACT-R's spacing model (Pavlik and Anderson, 2005) is used to assess the internal representation of presented word-pairs. On the basis of this model a Dynamic Spacing method is constructed that repeats word pairs just before they are forgotten.

We compared three variants of this method with a standard learning schedule. The three variants differed in the amount of adaptation to the individual's behavior. Students (selected from 3 HAVO/VWO) were presented with a learning session of 15 minutes on Day 1, and got an unexpected exam the next day. Analysis of the results shows that learning word pairs in the Dynamic Spacing condition results in better scores, given that the most sensitive adaptation method is chosen. This improvement is strongest in those students with below-average skills in the tested domain.

Although some issues remain, the work presented in this thesis shows that selecting an optimal Dynamic Spacing learning strategy improves the average results on the test by 10%, largely because students with below-average results score remarkably higher with the optimized learning schedule.

Acknowledgments

First of all I would like to thank my supervisor Hedderik van Rijn, who supported me throughout the entire project. Second of all I would like to thank Paul Inklaar (Belcampo College), Hanneke Hink (Werkman College) and Hilbert Oostland (Gomarus College) who enabled me to conduct my research experiments. Without their help I would not have been able to apply this research into a real-life setting.

Contents

Chapter 1

Introduction

What would you do if you only had 15 minutes to learn a list of word pairs for an exam tomorrow? Is there a certain learning strategy that you could use? For example, does it matter in which order you learn the word pairs? Or can it be more beneficial to learn only a set of these word pairs, but to rehearse this set more often? These are questions I would like to answer in this thesis. To get an idea about what strategy or learning schedule would be profitable, we have to take a look at memory research that has been done before. One of the key questions throughout the history on human learning is whether distributed practice is more beneficial than massed practice. After a lot of debate it is now recognized that a delay in rehearsal (in stead of repeating the same item twice or more) has a positive effect on the strength of the memory. This has become known as the spacing effect. By using the knowledge about this effect, an effective learning schedule can be derived. How to effectively learn word pairs in the short term is the question to be answered in this thesis.

1.1 History of the Spacing Effect

Ebbinghaus (1850-1909) has been one of the founding fathers of the scientific study of the mind. One of his main fields of research was human memory. By training himself on remembering lists of nonwords (Ebbinghaus, 1964, 1885), he was the first to identify what has become known as the spacing effect. Around the same time, Jost, inspired by Ebbinghaus, performed a series of experiments on the basis of which he introduced several laws of memory. One of Jost's laws which states the spacing effect is: *"if two associations are of equal strength but of different age, a new repetition has a greater value for the older one"* (McGeoch, 1943, p.140). This means that rehearsal of an item is more beneficial if the spacing between two rehearsals is large, because the memory of that item will become stronger than the memory of an item in which the spacing between two rehearsals is small.

It took however several decades before scientists were convinced the spacing effect was real and were able to identify under which conditions it occurred. Mostly to the frustration of Underwood (1970) who initially propagated the Total-Time law in free-recall learning. The Total-Time law stated that the amount of learning is a direct function of study time regardless of how that time is distributed. A breakthrough in favor of the *spacing effect* has been provided by Melton (1970). Not only did he provide an overview of the massed vs. distributed practice problem, but he also provided an overview of some new paradigms. The spacing effect states that distributed practice is more beneficial than massed practice. That is to say that repetitions that are separated by other items will be better remembered than repetitions that are adjacent (e.g., Glenberg, 1979). Furthermore, the *lag effect* states that the more items there are between two repetitions, the better memory for that item will be. This is sometimes known as the Melton effect.

Another couple of decades had to pass to fill the gap between theory and practice. Bahrick (1979) for example published a paper called *"the questions about memory we forgot to ask"*. In this paper he tried to put memory questions back onto the agenda and to explore the conditions under which information is maintained over a long time period. One decade later Dempster (1987, 1989) concluded that the spacing effect is robust and a ubiquitous phenomenon. This is because the spacing effect has been found in virtually all verbal learning tasks (e.g., paired-associate learning, free recall and recognition memory), but also in vocabulary learning and other classroom tasks like science and mathematical rule-learning. A second important conclusion by Dempster (1989) was that the theory about the spacing effect was underutilized in the classroom. Even though distributing study material is an easy thing to do, there has been a lack of application in the classroom setting. Reasons for this are the somewhat counterintuitive nature of the spacing effect and the lack of knowledge about this phenomenon by educators. Recent research by Seabrook et al. (2005) support these conclusions and also show that the spacing effect applies to a wide range of (child) ages.

Of course there is the question of how the spacing effect can occur. What are possible explanations for the spacing effect? There are several theories, and one of them is the Component-Levels theory by Glenberg (1979). This account suggests that the spacing effect is due to *variability encoding*. Variability encoding means that distributed practice increases the probability that a repeated item will be interpreted or analyzed differently at each occurrence and therefore strengthen the memory. Another account for the spacing effect is the *deficient-processing theory* (e.g., Hintzman (1974)). This approach states that the second occurrence of an item is not as thoroughly studied or rehearsed as the second occurrence of an spaced repetition. Therefore massed practice lacks some processing and isn't as efficient as distributed practice. Finally there is a *multiprocess account* in which the different

accounts apply under some circumstances, that is, the deficient-processing account for cued memory tasks and the variability encoding account for free recall (e.g., Greene (1989)).

Last but not least it is interesting if one could replicate the effects of spacing. Is it possible to predict human behavior based on some simple rules? There is a paper by Anderson and Schooler (1991) on this subject who describe a learning and retention function and even refer to data found by Ebbinghaus. The authors describe how human memory for specific items depend on frequency, recency and the pattern of prior exposes (spacing). Furthermore they try to figure out whether a power or exponential function fits the practice and retention function best. It was not until 2005 that Anderson together with Pavlik created a spacing model in ACT-R to predict the effects of spacing (Pavlik and Anderson, 2005).

1.2 Computational Models of the Spacing Effect

As mentioned before, one of the theories about the spacing effect is the encoding variability account, which was put forward by Glenberg (1979) in his Component-Levels theory. This theory states that the larger the time between two presentations, the greater the chance that the new presentation will be stored differently. Central notion here is that different traces for the same item result in a higher probability of that item being retrieved, rather than two repeated identical representations. According to Glenberg, each item can be traced via contextual, structural or descriptive components. The contextual component represents the context in which an item is learned. As the context does not change very quickly in a learning session, all trials during a single part or session are associated with the same context. This might be a reason why distributed practice is more beneficial than massed practice, as a different context creates an extra trace to the same memory item. The structural component can be translated as the categorization of an item, either by the learning material or by the subject's way of learning. Finally the descriptive component contains information about the articulation, meaning or other item related aspects and will be most useful in cued-recall tasks. Because the contextual component can change over time, this component explains best why distributing practice is a benefit. The latter two components are less relevant for the effect of spacing.

Search of Associative Memory (SAM) (Raaijmakers, 2003) is a mathematical model derived from the Component-Levels theory. Whereas the Component-Levels theory focuses on the encoding of a trace, the SAM model extends this theory by specifying how items are retrieved from memory. When an item is stored with little structural information, it will not be easily recalled when the system is solely provided with a structural cue. On the other hand, when both contextual and descriptive cues are provided, the retrieval of an associated item will be easier than the retrieval in which only the

contextual or descriptive cue is given. The basic framework of SAM consists of memory images that contain item, associative and contextual information. Each item is linked with the cues and stored images. On the basis of the strength of associations of these links, the memory strength of an item can be calculated by multiplying all individual strengths between the cues and the image. Finally the probability of retrieving an item (over another) is determined by the memory strength of the specified item divided by the sum of memory strengths for all the other items. Retrieving an item from memory when given the right cues is therefore always successful. To account for a basic forgetting phenomena, the contextual fluctuation model was introduced. The idea of this extension is that there is a context element that gradually changes over time. This implies that the probability of successful retrieval of an item decreases as the time between study and rehearsal increases. Furthermore it is important to note that when an item cannot be retrieved from memory, the retrieval of that item will also fail when the same cues are presented later on. Therefore, retrieval has to be successful in order to enhance the memory for that item with additional contextual information. The effect of spacing is thus modeled by successful repetitions in which the context, or another component, has changed enough.

Another computational model for the spacing effect was introduced by Pavlik and Anderson (2005). This model is based on ACT-R and contains a slight variation of standard ACT-R declarative memory equations in order to account for the spacing effect. In ACT-R each item in memory has the same speed of decay. The spacing model differs from standard ACT-R in that the speed of decay is not fixed anymore, but depends on the strength of the item in memory. This strength is known as the activation value of an item. As will be explained in more detail later, the activation value of a memory item in ACT-R depends on the amount of presentations (*frequency effect*) and the time of the presentations (*recency effect*). Items that are rehearsed often or recently will have a high activation value and will therefore be better remembered. The effect of spacing has been implemented in this model by providing a high decay to rehearsals of an item with a high activation value and a low decay to items with a low activation value. This means that items that are rehearsed often within a short time receive a high decay, while items that are rehearsed just as often but more spacious will receive a lower decay value.

There are some differences between the SAM and the ACT-R spacing model. One of the differences between the two models is about the strengthening of an item. In SAM a retrieval needs to be successful in order to store additional cues. When an item cannot be recalled from memory, there is no strengthening of additional cues and therefore no effect of spacing. The model of Pavlik and Anderson (2005) does not depend on correct retrieval to show an effect of spacing. In this model the height of the activation value determines the speed of decay and therefore the effects of spacing. This

means that a passive rehearsal with the same 'cues' also provides an effect of spacing, because each rehearsal has its own speed of decay. But before elaborating on this model, some basic understanding of ACT-R is needed. In the next section the ACT-R theory is explained in more detail to get a better understanding of how items are stored in and recalled from memory.

Chapter 2

The Spacing Effect in ACT-R

Because this thesis is about the learning word pairs, only a small part of the ACT-R theory will be explained. A general description about ACT-R will be given and a more in depth description about the declarative memory module. The declarative memory module takes care of storing and retrieving facts from memory. In standard ACT-R this memory module did not account for the spacing effect. After the adaption by Pavlik and Anderson (2005), the memory module could capture the effects of spacing. But first some knowledge about standard ACT-R is necessary to understand the spacing model in ACT-R.

2.1 ACT-R: a Model of Cognition

ACT-R (Adaptive Character of Thought - Rational) (Anderson, 2007; Anderson et al., 2004) is an architecture of cognition. It describes how human knowledge is acquired and produced. The basic components in ACT-R are *declarative* and *procedural knowledge*. The declarative knowledge refers to facts and explicit knowledge. This kind of knowledge is represented by chunks, which are memory items or facts. Procedural knowledge on the other hand is rule-like and refers to implicit knowledge. For example, production rules can specify how to retrieve and use declarative knowledge.

To get a better understanding of the ACT-R architecture, it is necessary to understand the following components. First, there are *modules*. For example, the visual module which visually perceives the "world" or the declarative module which takes care of the storage and retrieval of facts. Second, there are *buffers*. Each module is associated with one or more buffers. The processes in a module can be influenced by or influence the contents of a buffer. For example, if the model needs a certain declarative fact, a retrieval request is placed in the declarative (retrieval) buffer. The declarative module tries to retrieve this fact and - if found - places the result in the retrieval buffer. One of the

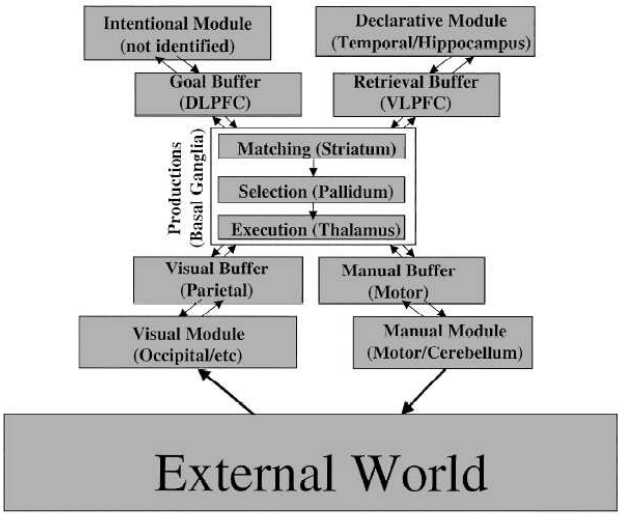


Figure 2.1: the basic architecture of ACT-R 5.0.

most important buffers is the goal buffer, which contains the current state of the model. Finally, there is the *central production system*. While different modules can run in parallel, the production system can only process one rule at a time and is therefore a bottleneck of the system. But one production rule can access and change different buffers at the same time. A module can only process one event at a time and is therefore another bottleneck. This process of matching rules with the content of the buffers, selecting and executing the production rules is what happens in the central production system. Figure 2.1 gives an overview of the modules, buffers and production system.

2.1.1 The Declarative Memory

The declarative memory module is the part of ACT-R that contains chunks or facts. To store and retrieve facts, each chunk receives a certain *activation level* which tells how strong that fact is in memory. The activation level depends on presentations from the past as well as the association strengths of the current context. This principle is reflected in the activation equation (see formula 2.1). This equation consists out of a base-level equation and an association part. The base-level equation depends on the number of prior presentations of a chunk as well as the age of all these presentations. The association strength depends on the relevance of the current context, but this part is not used in this thesis and will therefore not be further discussed here.

$$A_i = B_i + \sum_j W_j S_{ji} \tag{2.1}$$

Each fact in the declarative memory module has a certain activation level, which determines how easily and how quickly that memory item is returned. In this case only the base-level activation will be used to determine the activation value of a chunk. How the strength of a memory item is determined by the base-level activation can be seen in formula 2.2. This formula consists out of two important parts. The first part is the time t_j since the encounter j of an item. It is good to know that for each presentation of an item there is a time of encounter which is stored. The time t_j is calculated by subtracting the time of the j-th encounter from the current time. Thus, when the presentation of an item has been 10 minutes ago, the time t_j for that encounter will be 600 seconds. Older encounters will have a larger time since last encounter and more recent presentations will have a smaller t_j. This is important for the second notion about the base-level equation. There is a decay effect so that a learned fact will not be remembered forever. This decay effect is implemented by the parameter d. Note that when a large number is taken to the power of $-d$, this number will become small (e.g., $600^{-0.5} = 1/600^{0.5} \approx 0.04$). Therefore, according to the base-level equation, encounters that are old will have a small contribution and recent encounters will have a large contribution. Finally to determine the strength of one item in memory, the contributions of all the past encounters are summed up and the natural logarithm of this sum is calculated.

$$B_i = ln(\sum_{j=1}^{n} t_j^{-d}) \tag{2.2}$$

Two memory effects can now be explained by the use of the base-level equation. First of all there is the *frequency effect*. This effect states that the more an item is seen, the better the memory for that item. Using the above described formula it is easy to see that more encounters leads to a higher activation level and thus to a better memory for that item. Because when there are more encounters, there are more contributions (n) which make up the total sum of the base-level activation. Figure 2.2 shows that the activation level (the strength of an item in memory) increases after each encounter and drops slowly with the passage of time. This figure also shows that more encounters leads to a higher overall activation value.

The second memory effect that can be explained by the base-level equation, is the *recency effect*. This effect could already be seen in figure 2.2. But there is more to it than what is visible in the last figure. Because what for example happens when there are the same number of encounters, but at a later moment in time? Or when the same number of encounters are spaced more widely? The first question is easily answered, because the recency effect has a temporal effect, while the frequency effect has a permanent effect on the overall activation value. As can be seen in figure 2.3, the activation value

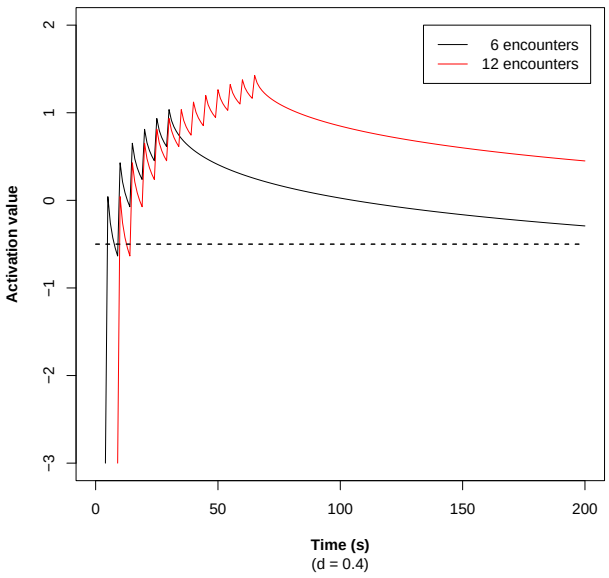


Figure 2.2: The frequency effect: the number of rehearsals determines the strength of an item in memory.

of the later presented item only has a benefit in the short term, but quickly becomes just a memorable as the earlier presented item. This should also answer the second question. Because when an item is spaced more widely, and the time of the last encounter remains the same, the earlier encounters will contribute less and less to the overall activation value. Figure 2.4 shows that the benefit of wide spacing is due to the recency effect, but that this advantage is only temporal. Therefore, the standard ACT-R model does not account for the effect of spacing. If it did account for the benefit of spaced repetitions, the advantage of the larger spacing would be permanent.

Other important parts of the declarative memory are the probability and latency of recall for chunks. To simply know the activation value of an item is not enough. What is the probability that an item can be retrieved from memory? Can the item be retrieved at all and if so, how quickly will it be retrieved? To start with the first question, a threshold τ has to be introduced which states at what point items can and cannot be remembered anymore. Activation values below this threshold cannot be recalled from memory anymore. Because in practice such a threshold is not very strict, but rather gradual, a noise parameter s can be used to account for the fluctuation each time an attempt is made to recall an item. The higher an activation value of an item, the better the memory for that item and thus the more easily it will be to recall the item. This concept is captured by the formula 2.3.

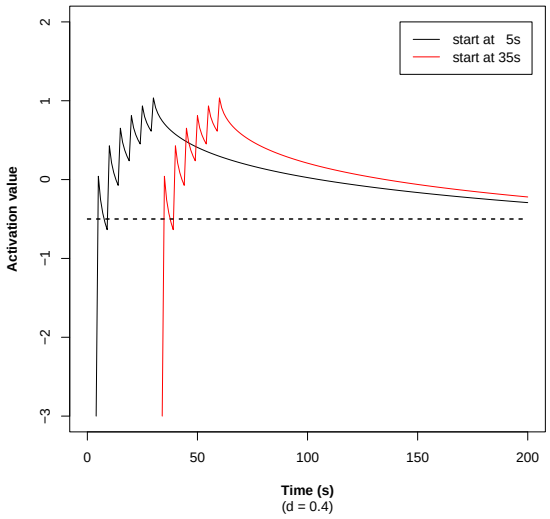


Figure 2.3: The recency effect: the strength of an item in memory decays over time.

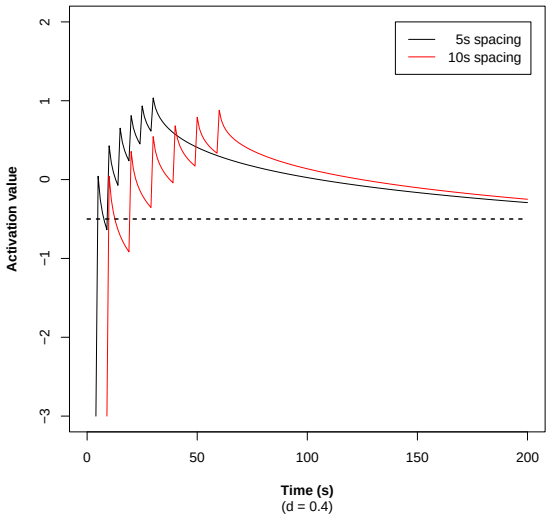


Figure 2.4: No effect of spacing in standard ACT-R: increasing the spacing between rehearsals does not have a lasting effect.

To answer the second question, it is necessary to know whether an item could be retrieved at all. Only when an item can be retrieved is it possible to calculate the speed of this retrieval. Like the probability of retrieval, the latency of retrieval depends on the activation value. Because items that are well remembered will be easily retrieved, while items that are hard to remember will also have a long retrieval time. The latency of retrieval of a chunk from memory is captured by formula 2.4.

$$P_i = \frac{1}{1 + e^{-(A_i - \tau)/s}} \tag{2.3}$$

$$T_i = Fe^{-A_i} \tag{2.4}$$

2.2 The Spacing Model in ACT-R

Because the declarative memory module of ACT-R could not explain the effects of spacing, Pavlik and Anderson (2005) made an adjustment to the base-level equation. In standard ACT-R the strength of an item in memory has no effect on the speed of decay. Therefore - concerning the speed of decay - it does not matter to the model whether presentations are very quickly after each other or spaced over some time. The spacing effect, however, predicts that the interval between presentations has an effect on the speed of decay and thus influencing the strength of activation in the long term.

Now then: how is the spacing effect implemented in ACT-R? To answer this question we need to have another look at the base-level equation. The base-level equation accounts for the *frequency effect* (the more often an item is seen, the better the memory for that item) and for the *recency effect* (the longer ago a presentation has been, the lower the contribution for the memory of this item). To also account for the *spacing effect*, the decay of an item needs to be dependent upon the spacing of the prior presentations. How can this be done? The answer is quite straightforward. Items that are largely spaced over time, will have a lower overall activation value than items that are rehearsed more quickly (assuming that the time of the last presentation is the same). Therefore, items with a low activation value should receive a smaller decay on a new encounter than items with a high activation value. This means that repeating items that are highly active in memory is - apart from the frequency effect - not as beneficial as repeating items with a low activation value. This concept is demonstrated in figure 2.5 where can be seen that a higher spacing of the same number of encounters is more beneficial than presenting an item shortly after each other. Figure 2.6 shows what would have happened according to standard ACT-R given the same situation.

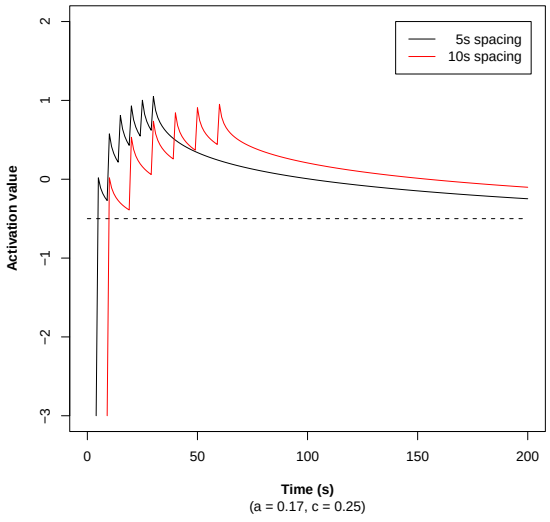


Figure 2.5: Spacing effect when using the spacing model: increasing the spacing between rehearsals has a positive effect in the long term.

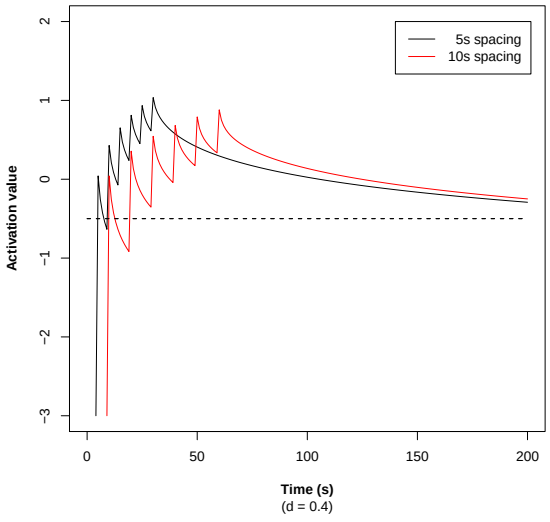


Figure 2.6: No effect of spacing in standard ACT-R: increasing the spacing between rehearsals does not have a lasting effect.

Formula 2.5 and 2.6 show the mathematics behind the spacing model. As can be seen in formula 2.5, the decay parameter is not fixed anymore, but depends on the j-th encounter of an item. Every time a new presentation or rehearsal occurs, the time of that encounter is stored. Since the decay

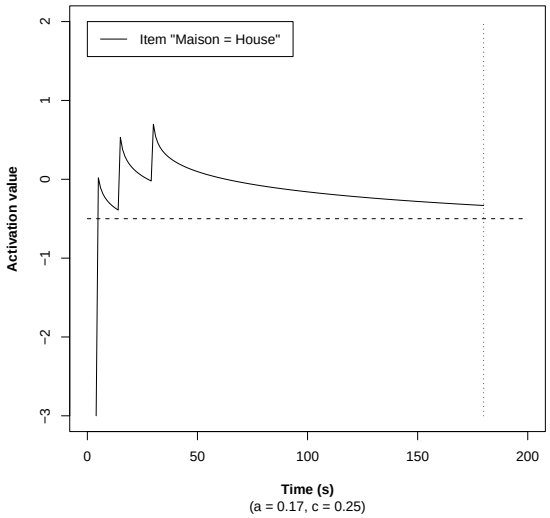
value in the spacing model is not fixed anymore, a unique decay value is calculated for each encounter of an item. How this encounter-specific decay value is calculated is shown in formula 2.6. The speed of decay now depends on the strength of the activation value (at the time of calculation). A high activation value will result in a high decay and a low activation value will result in a low decay. It is therefore in the long term more profitable to have a large spacing between presentations, so that the activation value remains small, which results in a low speed of decay for each new encounter. Of course the activation value should not be too small, because then retrieval of that item will become difficult. But this is something to be discussed later on. Furthermore the speed of decay depends on two parameters: a and c. The c parameter represents a scale factor for the impact of the prior presentations. Therefore, the c parameter is important for determining the strength of the spacing effect. The a parameter represents a constant value that is added to every decay value. This parameter thus holds the minimum decay for each encounter. Note that when $c = 0$ and $a = 0.5$, we are back at the fixed decay value of $d_j = 0.5$ for every encounter, which is the standard decay value in ACT-R.

$$m_n(t_{1..n}) = \ln(\sum_{j=1}^{n} t_j^{-d_j}) \tag{2.5}$$

$$d_j(m_{j-1}) = ce^{m_{j-1}} + a \tag{2.6}$$

Last but not least an example of how an item remains in memory, to get a better grip on the formula's. Suppose we want to learn a new fact like "Maison = House". After the first presentation (t=5) the item will be quickly forgotten, so it is repeated again after 15 and after 30 seconds. What will the activation value be after 3 minutes and will you remember the item by then? The spacing model will be able to provide an answer. For this, the activation value needs to be known after 3 minutes. At first the item "Maison = House" has no activation value (or -Infinity), because we assume it is a new fact. After the first encounter (t=5) the decay value d_1 will be equal to a (which is 0.17 in this example), because there is no effect of previous encounters ($e^{-Inf} = 0$). Just before the second encounter, however, there is an activation value at t=15, which is -0.391 ($m_1 = \ln(10^{-0.17}) = -0.391$). Therefore, the decay value d_2 for the second encounter will be $0.25 \cdot e^{-0.391} + 0.17 = 0.339$ (in which $c = 0.25$). The same calculation can be done for the third encounter which gives a decay value of $d_3 = 0.25 \cdot e^{-0.022} + 0.17 = 0.414$. Note that the activation value just before the third encounter is now a bit more difficult to calculate, because there are two previous encounters. That is, $m_2 = \ln(25^{-0.17} + 15^{-0.339}) = -0.022$. To answer the question whether the learned fact will be remembered after after 3 minutes (180 seconds), the activation value at t=180 needs to be calculated. This value

(m_3 at t=180) is equal to $\ln(175^{-0.17} + 165^{-0.339} + 150^{-0.414}) = -0.331$. Given a threshold τ of -0.5, the fact "Maison = House" will be remembered after 3 minutes, because the activation value is higher than the threshold at the time of test. The activation values can be seen in figure 2.7. The table next to this figure displays the activation and decay values at t = 5, 15, 30 and 180 seconds.



time (s)	activation value	decay value
5	$m_0 = -Inf$	$d_1 = 0.17$
15	$m_1 = -0.391$	$d_2 = 0.339$
30	$m_2 = -0.022$	$d_3 = 0.414$
180	$m_3 = -0.331$	-

Figure 2.7: An example of the activation and decay values for a chunk "Maison = House".

Note that the activation value depends on the time of calculation. For example, the activation value of the chunk "Maison = House" at time point t = 60 seconds, equals $m_3 = 0.025$. As long as there are no more new encounters, the activation value will decrease as time passes by. This has to do with the times t_j since last encounter which *increase* as time passes by. By simply changing the time t of calculation, the activation value can be derived at any point in time.

Another aspect to note in this example is that the decay value has increased after each rehearsal. Repeating a word pair on such a short term is not beneficial for the speed of decay. The three encounters are however useful, because of the frequency effect. But repeating the same item, for example, after t = 180 seconds in stead of t = 60 seconds, will result in a lower decay value ($d_4 = 0.35$ vs. $d_4 = 0.426$). This is therefore how the ACT-R spacing model accounts for the spacing effect.

2.2.1 The Bahrick Experiment

A very thorough research on learning word pairs has been done by Bahrick (1979). He examined the performance on vocabulary learning using 50 Spanish-English word pairs. The spacing of the word pairs in this experiment was in the long term. Instead of spacing the word pairs within a session, there were three groups which had either a 0, 1 or 30 days interval between rehearsal of the word pairs. Furthermore he used a technique for rehearsal that guarantees that each word pair is recalled correctly

exactly once during a rehearsal session. So, each word pair that is incorrectly recalled is shown again and stored in a queue. Then the queue of incorrect word pairs is rehearsed and word pairs that are recalled incorrectly again are stored in a new queue. This process repeats itself until each word pair has been recalled correctly once.

Table 2.1 shows the results of this experiment. The data shows the percentage correct of the words pairs at the beginning of each rehearsal session. Naturally the percentage correct increases with each new rehearsal session. Also the probability of recall is much higher for sessions with a short (or no) intersession interval. But when testing the word pairs again 30 days after the last session, the group with the highest intersession interval has the best performance. This experiment shows very well that a large spacing (using the same amount of study time) is more beneficial for remembering word pairs in the long term.

Inter-session interval (days)	Session 2	3	4	5	6	Following the 30-day interval
After three training sessions						
0	77	89				33
1	60	87				64
30	21	51				72
After six training sessions						
0	82	92	96	96	98	68
1	53	86	94	96	98	86
30	21	51	72	79	82	95

Table 2.1: Results of the Bahrick experiment.

In Pavlik and Anderson (2005) the spacing model is compared to several memory experiments, among which the above described experiment by Bahrick. Their fit using the spacing model came quite close to the results found by Bahrick. In order to replicate these results, they used a deterministic method to calculate the outcome directly. So in stead of running multiple trials and calculate the average score, the probability equation was used to calculate the average probability correct directly at the time of test. For this the assumption has to be made that each item is learned equally well. Furthermore an extra parameter was introduced to compensate for the rehearsals. So instead of rehearsing the incorrect recalled word pairs, there was only one encounter at the beginning of each

session and a multiplier b (set to 3.79) to increase the activation value at the start of each session. This means that there is no longer a spacing effect within a session, but only between session. Furthermore the b parameter should compensate for possible multiple encodings, because in the Bahrick experiment the first time each word pair was presented it was also pronounced verbally. Finally the duration of a learning session was estimated to last for 40 minutes. This means that for the 0 day interval the second session started after 2400 seconds. For the intersession intervals of 1 and 30 days a psychological time was used which means there is (in this case 40 times) more interference during a session than between sessions. When, for example, there is a 1 day interval without learning, this counts for the model as if only 86400 / 40 = 2160 seconds have passed. The start time of the second session for the spacing model should therefore be (2400 + (86400 - 2400) / 40 =) 4500 seconds. For reasons not clear to me, the start of the second day learning session in this simulation was after 6900 seconds.

I tried to replicate the experiment by performing a Monte Carlo simulation. This means that I ran a lot of trials, using noise on the activation values, and averaged the results at the time of test. I used the above described spacing model, but without the extra parameter b. Also, the word pairs that were incorrectly recalled are rehearsed until they are recalled correctly once, as in the experiment by Bahrick. In this simulation, each presentation or rehearsal took 5 seconds. Three encounters were stored (rather than one) for each successful presentation or rehearsal of a word pair, to account for a rehearsal effect. One session typically took 15 to 20 minutes for the first few sessions and 5 to 10 minutes for the last few sessions, depending on the spacing used between the sessions. I also used the psychological time for the model, so the intersession interval time between two sessions was reduced by a factor 40. This simulation should yield the same results as the one performed by Pavlik and Anderson, because it should not matter whether the result is calculated on average (over a large amount of trials) or directly by some derived formula. Unfortunately I was not able to replicate the data very well for the 1 and 30 days spacing intervals.

Wherein then lies the difference between my simulation and the one used by Pavlik and Anderson? I used the same parameters for my simulation, expect for the b parameter. I did, however, implement a rehearsal effect to increase the strength of one repetition, so that a rehearsals will not have been forgotten at the start of the next (spaced) session. The results on the 0-day intersession interval matched the data by Bahrick or Pavlik and Anderson very well. The results on the 1-day intersession interval were not the same as in the other experiments, but it did show the same trend. The results on the 30-day intersession interval were very poor. This has to do with either the small number of repetitions or the high decay values. But more about this after examining the extra b parameter. Because what does the multiplier b do? It increases the activation value at each moment in time

as to account for multiple presentations. Having multiple encounters within a very short time is not the same, because this would yield a high decay value. Using the b parameter can therefore be compared to having multiple encounters while using the decay value of the first encounter. Without the b parameter, the multiple encounters must have enough spacing to prevent a high decay value. I found that rehearsing the word pairs 6 or 7 times within each session, provides a much better fit of my simulation on the data. But when using the repetition method as described above (rehearse only incorrect word pairs), the simulation had only 1 or 2 repetitions on average, which is on the low side. This also meant that one of the first few rehearsal sessions typically took about 15 to 20 minutes, rather than the estimated 40 minutes. I also found that decreasing the decay parameters (especially the intercept parameter a) provides a better fit. This is due to the same reason that multiple encounters give a high activation and thus a high decay. Thus, in stead of using the multiplier b, either the decay parameters has to drop or their need to be enough spaced repetitions.

2.2.2 Positive Aspects

As mentioned before, the spacing model has been compared to several memory experiments. Another one worth mentioning here is the experiment by (Glenberg, 1976, Experiment I). In this spacing experiment the spacing of items is kept within one session. Participants had to learn four-letter word pairs consisting of unrelated common nouns. There were 500 events in a row, each consisting of either a presentation or test trial. Each word pair was presented twice at lags of 0, 1, 4, 8, 20, or 40 items. These word pairs were then tested at a retention interval of 2, 8, 32 or 64 items. For the long lag and retention interval a small variance of a few items was allowed.

The results of the Glenberg (1976) experiment show that there is a nonmonotonic performance for the short retention intervals (2 and 8 events), and a monotonic increase in performance for the long retention intervals (32 and 64 events). Therefore, short (but not too short) spacing is more beneficial when the time of test is quickly after the last presentation. In the long term increased spacing also increases performance on the test. This can be seen in figure 2.8, where the 2 and 8 item retention intervals perform better on the 4 and 8 presentation lag intervals, whereas the 32 and 64 retention intervals increase monotonic as the lag interval increases.

The model of Pavlik and Anderson (2005) fit the data quite well. For this the b parameter was used again in an identical way to the Bahrick (1979) model. This extra parameter was used to scale the presentation-test interval times, because these times were shorter in the Glenberg experiment than in the simulation of this experiment. Even though there is some deviation from the Glenberg experiment, the spacing model could account for the optimal spacing based on the retention interval.

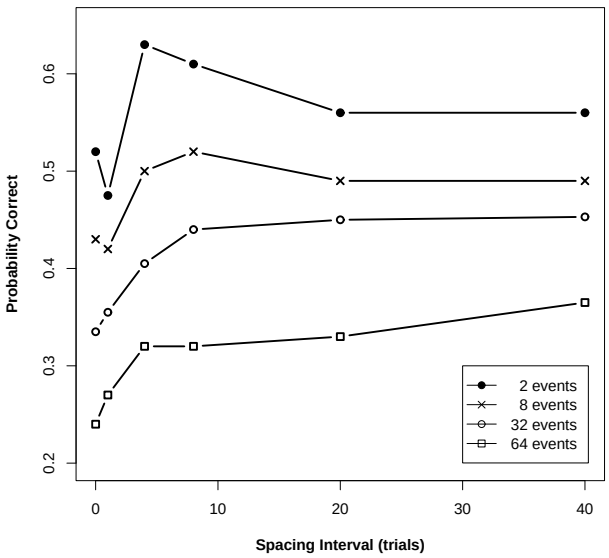


Figure 2.8: Results found by Glenberg (1976).

Further more, Pavlik and Anderson performed an experiment on their own. In this experiment participant were asked to learn 104 Japanese-English word pairs. On the first session, the items were studied once and then tested 1, 2, 4 or 8 times with 2, 14, or 98 intervening presentations. This is much like the Glenberg (1976) experiment, expect for the number of study and test events per item, which is in this case usually more than 2. On the second session, either 1 or 7 days after the first session, the participants were tested in much the same way as the first session. The results of this experiment show a crossover interaction (see figure 2.9). That is, word pairs learned with a short spacing are initially bettered remembered, but will eventually have a poorer performance than items learned with a longer spacing. Therefore, a large spacing between rehearsals of an item will result in less forgetting.

All in all the spacing model can account for the effects of retention and spacing on memory. For example, the crossover interaction (initial low performance on high spacing will eventually outperform small spaced rehearsals) is captured by this model. This effect is also shown well by the Bahrick (1979) experiment. Furthermore the model can capture the effect of optimal spacing for short lag and retention intervals and the monotonic increase for long spacing intervals. The model was also successfully fit to other experiments, like the Rumelhart (1967, Experiment 1) and Young (1971) experiment. The spacing model can therefore be used to create a smart learning schedule.

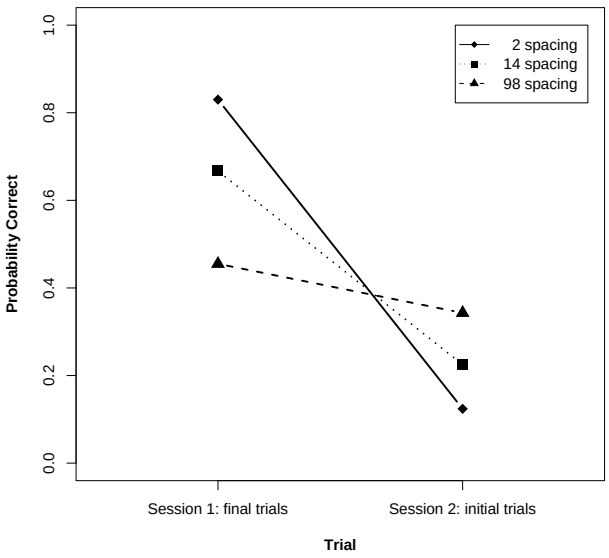


Figure 2.9: The crossover interaction as shown by Pavlik and Anderson (2005).

2.2.3 Negative Aspects

The spacing model as presented by Pavlik and Anderson (2005) also has some shortcomings. Although the solution of changing the decay parameter for each encounter is an elegant adaption to the ACT-R (declarative memory) model, several problems might occur in the short term. First of all, when a single word pair is presented continuously for say 5 minutes, the activation value - and therefore the decay rate - will become very large. This might result in a lower performance than when two or three word pairs are learned alternately for 5 minutes. Although it is true word pairs are better remembered when there is some spacing between each word pair, it is rather strange that remembering one word is more difficult than remembering two or three word pairs in the same amount of time. This example will be explained below in more detail.

The second more drastic flaw results from the same principle as the first and concerns the rehearsal effect. Suppose a word pair is shown for 10 seconds and the model is able the rehearse the word pair every second within these 10 seconds. This means that there are 10 encounters within a single learning event. Because the encounters are very quickly after each other, the activation value will become very high, which results in a rapid decay for the last encounter. Within such a short time, only the decay value of the last encounter has a great impact on the activation value in the long term. Now suppose the same item would have been rehearsed only 4 times within the 10 seconds of the same learning event. The decay value for the same item will be much lower and this might result in a better memory

in the long term, even though in the first case the word pair has more rehearsals. This is strange, because usually the frequency effect has a much stronger impact on the activation value than the spacing effect, especially when a word pair is rehearsed 2 times more often. However, according to this spacing model, many repetitions within such a short term can be a drawback. This is shown in figure 2.10.

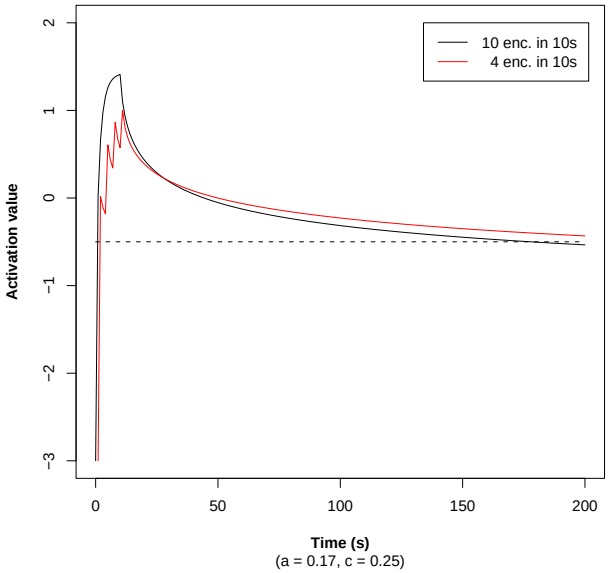


Figure 2.10: Drawback of the spacing model: a lot of repetitions in the short term can counter the benefit of the frequency effect.

The first flaw, which is derived from the same principle, is explained now in more detail. Suppose the model learns one word pair for one minute. Every five second this word pair is presented again so that there are 12 encounters within one minute. Now suppose there are two word pairs presented for five seconds, one after each other. In this case each of the two word pairs would have 6 encounters within one minute. Because the activation value of the single word pair would be much higher than the activation values of the two word pairs that are presented after each other, the decay value of the single word pair will be much higher. Therefore - in the long term - the single word pair would be easier forgotten than the two word pairs. This is somewhat strange, because remembering one word pair with 2 times the number of repetitions should be easier than remembering two word pairs with each half the number of repetitions. This could be because there is no variety in the case of learning only one word pair, but alternating 3 word pairs should not outperform the learning of only 2 word

pairs in the same amount of time, as is the case in in the example below.

I stumbled upon these drawbacks when trying to figure out whether there is some optimal amount of word pairs to learn in a given time. Table 2.2 displays the results of learning n word pairs for 5 minutes and then testing 30 minutes after learning. The word pairs are presented one by one with a spacing interval of 5 seconds. The first column of the table shows the number of words to be learned during the study phase. This is to find out which amount of word pairs is optimal to learn in the given time span. The second column shows the average probability of recall for all the n learned word pairs. One would expect to see that learning a single word pair and then testing on that single word pair would be easier than learning and testing three word pairs. But according to this example, learning three word pairs is more beneficial than learning one or two word pairs, which is rather strange. The third column shows the number of words that where correctly recalled at the time of test. Word pairs with an activation value below the threshold τ at the start of the the test are counted as incorrect. As can be seen here, learning more than 15 word pairs in 5 minutes will result in forgetting some of the word pairs. Also the number of words forgotten at the time of test drops rapidly as the number of word pairs to be learned increases beyond 15. So although learning three word pairs will result in the highest average probability of recall, learning more word pairs will be more beneficial because of the number of items remembered at the time of test. In this example learning 15 word pairs is the most optimal number of words to learn, because the activation values of these words are just high enough at the time of test to be remembered.

Another aspect of the presented spacing model also deserves some attention and concerns the psychological time. This issue has to do with the retention interval between the end of a learning session and the start of test. How well will an item be remembered after a week or a year? As time passes by the decay of an item seems to be slower than during a learning session. This can be explained by the interference effect, which states that during learning the memory of certain items is affected, while outside a learning session these memory items do not have to be altered. Because of the competition during learning, memory items are more easily forgotten within a learning session. ACT-R uses a *psychological time* to mimic this effect. This is done by a parameter h which is used to scale the amount of interference outside of a learning session. Suppose the interference parameter h is set to 0.025, then this means that interference during a session is 40 times greater than after the session. When looking at the short term, this psychological time is a bit awkward. Because what does the model predict 5 minutes or an hour after learning? This hardly has an effect on the activation values (1 hour * 0.025 = 1.5 minutes of extra decay time), while during learning every minute of decay seems to count as an hour of not learning. It is however true that there is a lot of interference

n words	average probability of recall	n words correct at test
1	0.684	1
2	0.758	2
3	optimal? **0.770**	3
4	0.765	4
5	0.754	5
6	0.740	6
7	0.722	7
8	0.703	8
9	0.683	9
10	0.665	10
11	0.642	11
12	0.623	12
13	0.598	13
14	0.577	14
15	0.559	optimal! **15**
16	0.424	12
17	0.302	9
18	0.192	6
19	0.092	3
20	0.000	0

Table 2.2: Number of word pairs learned related to score at test.

between items during learning and that the psychological time is meant for longer retention intervals. But concerning the spacing effect, there is a small problem. In standard ACT-R, the decay value for each encounter is the same, so each item decays at the same pace. This means that it does not really matter for the difference between items whether the psychological time was set to $h = 1/40$ or $h = 1/60$, because a shorter retention interval will be beneficial for all the word pairs. When using the spacing model, however, each item has its own decay and it therefore does matter what the amount of interference, and thus the time until test, is. For example, testing (1 day * 0.025 =) 36 'minutes' or (1 day * 0.017 =) 24 'minutes' after learning has a different impact on individual items, because of possible crossover interactions.

2.2.4 Summary

The ACT-R spacing model, as presented above, can predict the outcome of several memory experiments. This was done on the basis of the ACT-R declarative memory module, which can account for the frequency and recency effect. A shortcoming of standard ACT-R is that it could not account for the spacing effect. That is, the size of the interval between repetitions has no effect (or little effect due to recency) on the activation value of a memory item. To account for the spacing effect Pavlik and Anderson (2005) introduced the spacing model, which is a small adjustment in the declarative memory module of ACT-R. The only difference with standard ACT-R is that the decay value for each encounter of an item is now unique. This decay value depends on the strength of a memory item, so that repeating an item that is highly active is less beneficial than repeating an item with a low activation value. This principle results in the spacing effect, because an item that is repeated often in a short time will have a higher decay value (and will therefore be easier forgotten), than an item that is more widely spaced.

The spacing model has been fitted to several well known spacing experiments. Although some adjustments of the parameters were necessary, the spacing model can account for the different effects of spacing. For example, the data by Bahrick (1979) shows very well the effects of spacing. The spacing model fitted the data very well with the use of a b parameter to compensate for the number of rehearsals and the possibility of multiple encodings. Other models were also fitted well, among which the experiment by Glenberg (1976, Experiment I). This experiment showed that for short spacing intervals there is an optimum when the test trial is shortly after presentation, but for longer spacing intervals the performance on the test increases as the lag between items becomes larger. Another example shows that the crossover interaction is captured by the spacing model in the experiment performed by Pavlik and Anderson (2005) themselves. In their experiment word pairs with a short spacing scored best on the final trials of the first session, but had the lowest scores at the beginning of the second session (and vice versa for wide spacing).

There are, however, some shortcomings of the spacing model. Because of the adjustment of the decay parameter, there are some side-effects that might occur. This is especially true for items with a high activation value, usually in the short term. Take for example a phonological loop in which an item is repeated multiple times within one study event. Normally the frequency effect has a higher contribution to the overall activation value than the spacing effect, but when an item is repeated very often in a short time, the decay value for that item will become very large. It is therefore possible that an item is easier forgotten than the same item with less repetitions within one event. The same principle holds for repeating one, two or three items for a longer period, in which repeating

an item more often can become a disadvantage. Although large activation values can be a problem for the spacing model, this is rarely the case in practical applications. Another thing to note about the spacing model is the effect of crossover interaction in combination with psychological time. As explained above, there is more interference during learning that outside of a learning session. An interference factor h can compensate for this effect, but this means that the time until test depends on this factor h. Since each item has a unique decay in the spacing model, changing the interference factor has an impact on individual items in stead of on all the items in the same way.

Last but not least, the spacing model has been demonstrated to capture the effects of spacing. Also, a prediction can be made on optimal word pair learning. Because the activation value does not only depend on the frequency and recency of rehearsals anymore, but also on the spacing between rehearsals. As was shown by the crossover interaction in figure 2.9, the performance on a test depends on the spacing used and on the time of the test. When knowing the time of test in advance, an optimal learning schedule can be derived, based on the activation values of the model.

Chapter 3

An Experiment Using the ACT-R Spacing Model

The question to be answered is how to create a smart learning schedule for learning word pairs in the short term. We now have a little bit more information on how to create a smart learning schedule, based on the ACT-R spacing model. But there are some things to note about creating such a schedule, which is explained in the introduction of this section. For example, what is the impact of active rehearsal vs. passive rehearsal during learning? And should the amount of learning time be variable, or the amount of word pairs to learn? After an explanation is given of how to create a smart learning schedule, the implementation of the chosen strategy will be discussed in the method. This section describes what the different learning conditions are, how they are implemented and what parameter-values are used in this experiment. The results show an analysis of the data collected during the experiment. In this section it will be shown whether there is a relationship between the learning method used and the score on the test. Finally the conclusion gives an overview of whether some optimal learning scheme based on the spacing model is indeed more beneficial than some other standard learning method.

3.1 Introduction

In the following section the rationale of how to determine a smart learning schedule is explained. In order to determine a smart schedule for learning word pairs, there are some things that need to be clear. First of all, what is meant by a learning schedule? Second of all, what are the constrains in determining a smart learning schedule? And third of all, how can this be implemented in the ACT-R spacing model? Finally, the experiment is based on a computer program, so presenting and rehearsing word pairs is discussed in the context of a computer program.

First of all an explanation of what is meant by a learning schedule. The key variable for this is the order in which word pairs appear during study. To illustrate this, an example is given. Suppose that a word pair like "Maison = House" is presented for several seconds on the screen. Of course only one item will be processed at the same time. After this item a new word pair can be presented. After several items have been presented it may be time to rehearse the first word pair again - to recall the item from memory. At what moment an old word pair should be rehearsed or a new word pair presented, is the key question. For example, the order of word pairs can exist out of the following sequence, in which each number represents a word pair: 1, 2, 3, 4, 1, 2, 5, 3, 4, 6, 7, 5, ... As can be seen, the first word pair is presented at the first position and rehearsed at the fifth event. The order in which the word pairs appear is what is meant by a learning schedule. This is illustrated very clearly in figure 3.1.

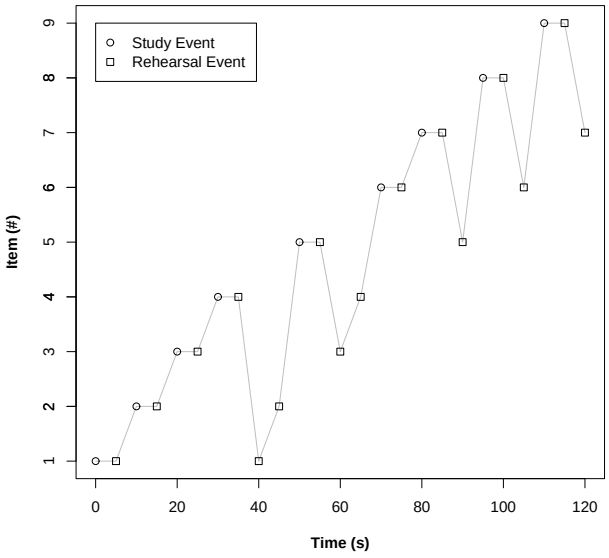


Figure 3.1: An example of the sequence of word pairs presented over time.

One of the ideas for creating a smart learning schedule was to determine the word order in advance. This could be done when provided with an amount of learning time and the start time of the test. Suppose you have a test tomorrow in which you have to remember the translation of 20 different words. You only have 15 minutes to learn for the test and then go off doing something else. What then is a smart learning schedule in which you profit the most from learning the word pairs? That is, what schedule gives you the highest score on the test tomorrow? There are three questions that can

be answered. First of all, is there an optimal amount of word pairs to be learned? For example, is it better to learn only 15 words, because you remember these words better than learning 20 words in the same amount of time? Second of all, is there a minimum amount of time in which it is possible to learn these 20 word pairs for the test of tomorrow? For example, is it possible to learn the 20 words in 12 minutes without any loss of performance? And third, is there an optimal schedule in which you always learn for the same amount of time, but only the order in which the words appear differs? The three questions are illustrated in figure 3.2. The last question is the one I will try to answer. But before elaborating on the third question, let me explain why the first two questions are more difficult to answer.

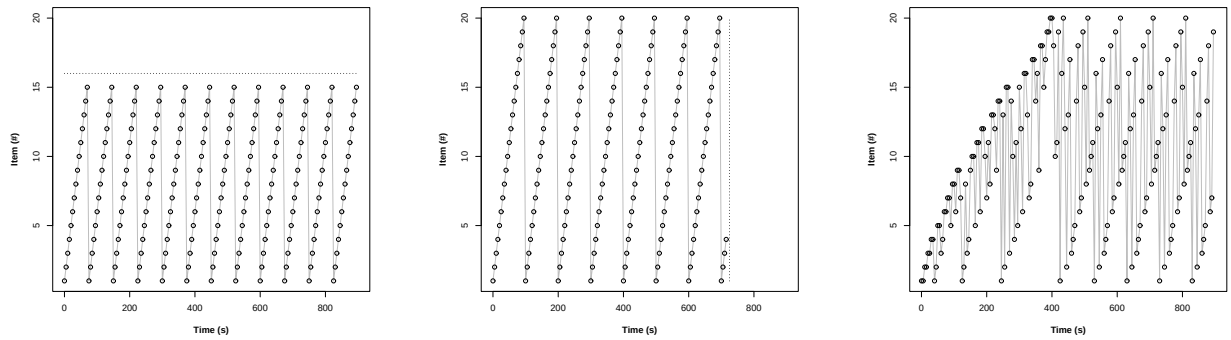


Figure 3.2: Finding an optimal learning schedule based on the number of word pairs to learn (left), the amount of learning time (middle) or the word order (right).

The first question out of three was about finding an optimal amount of word pairs to learn in a fixed learning time. It might for example be better to learn only 15 word pairs which you will remember very well the next day, instead of 20 word pairs which are all easily forgotten. This is because there are less repetitions of each word pair when you learn more words in the same amount of time. The spacing model can make a prediction for this situation, given some fixed schedule. Adapting the learning schedule (word order) as well as the number of words makes this situation complex to analyze. So let us assume a fixed learning schedule in which all the word pairs are learned one by one and then starts at the beginning again after the last word pair has been presented. Note that in this schedule the spacing between rehearsals of the same item now depends on the amount of word pairs to be learned. It is impossible to keep the same spacing while varying the amount of words to learn. But it is possible to take this variable into account when figuring out which amount of word pairs will give the best result on the test the next day. Figuring out what amount of word pairs would be optimal to learn was shown in the section about shortcomings of the spacing model. Still it is difficult to answer this question, because it requires a lot of experiments to test the hypothesis and

there are two dependent variables. Last but not least there is the possibility that some users will require more practice than others depending on their reaction times and amount of incorrect answers. Therefore, the number of rehearsal events within a fixed amount of learning time will depend upon the performance of the user.

The second of the three questions was about an optimal amount of time to learn 20 word pairs for the next day. This means that the number of words is fixed, but the amount of time is variable. Here too arises the question of what learning schedule to use, but for simplicity it is good to use only one (fixed) schedule. Besides, the spacing model predicts a schedule that is optimal, namely that large spacing is better for decay. In this case we use the same schedule as before which means that all the words pairs are presented after each other before starting at the beginning again. To predict when it is time to stop learning, the activation values of all the word pairs need to be above a certain threshold at the time of the test. The question, however, is how high the activation values need to be at the time of test. Since there is always some noise and a probability of recall, the model cannot guaranty a perfect score, even when you learn for a long time. Therefore there is an extra question to be answered in this condition: at what point are the word pairs learned 'good enough'? Although we are free to choose a safe amount of learning time, there is still the problem that the reaction time of the user and the number of incorrect answers influences the time between rehearsals. This last problem has an impact on both the number of rehearsals as well as the speed of decay. That is, the amount of learning time depends on (the reaction of) the user.

The third question is the one which I want to answer in this thesis. As was shown in the previous two research questions, the question of what learning schedule to use came up despite the original question about the number of words or the amount of learning time. Also the need for a dynamic schedule was brought to attention. Therefore a good research question would be: what is an optimal learning schedule when the number of words and the amount of learning time are fixed? This question is easier to answer, because now the word order is the only dependent variable in the experiment. A solution to the previous questions concerning the learning schedule was to present all the word pairs after each other and then repeat the word pairs from the beginning. This guarantees the maximum amount of spacing for each word pair. But on the other hand is it very likely that word pairs will have been forgotten by the time they are repeated again. We know that active rehearsal (correct remembrance) is more beneficial for a memory item than passive rehearsal (to show the answer again after incorrect or no recall). Therefore it is better to rehearse word pairs before they are forgotten. Besides that, it is motivating if you remember a lot of word pairs correctly instead of being confronted with word pairs you do not remember anymore because of the large spacing.

Based on the above reasoning, the following dynamic schedule will be useful for a computer program that assists in learning word pairs. This dynamic schedule determines during training what the next word pair to be presented or rehearsed will be. According to the spacing model a word pair has a certain activation value as soon as it has been presented. This model also contains a threshold for determining at what point a word pair can be recalled and when it is forgotten. The activation values of the items in memory can be compared to a fixed threshold to determine whether a word pair should be rehearsed or whether a new word pair can be presented. But finding the next word pair like this is not as simple as it seems. Because once the activation value has passed the threshold, it is already too late for rehearsal. Therefore a word pair should be rehearsed slightly earlier, before its activation is below the threshold. But at what point above the threshold should a word pair be rehearsed? The idea of an extra threshold is not a good option. This is because each word pair has its own decay, and some word pairs will reach the threshold faster than other word pairs. A better question is: at what point *before* the threshold is *reached*, should a word pair be rehearsed? The activation value of a word pair might be above the threshold now, but can be below the threshold after the 5 or 10 seconds that another word pair is presented or rehearsed. This means we need to look ahead to predict whether a word pair will be below the threshold after several seconds from now. This needs to be done for all the word pairs that have been presented so far, in order to keep up the memory for these items.

How to keep up the memory of all the earlier presented items in a dynamic learning schedule will be explained here. First of all, all the word pairs are assumed to be unknown. Their activation value is therefore unknown (or -Infinity if you want to calculate an activation value with zero encounters). After the first word pair is presented, it has an activation value that will be above the threshold for some amount of time. Then we have to look ahead for several seconds to see whether the activation value is still above the threshold after the next presentation or rehearsal of another item. If this is the case, a new word pair can be presented to the subject. If this is not the case, the current word pair needs to be rehearsed, so that the activation value is high enough to have some time to present other word pairs. Note that this is still the easy case, because there is only one word pair in memory. But suppose there are several items in memory that need to keep their activation values above the threshold for some time. In this case it is smart to look ahead for several events, because there is the risk of more than one item falling below the threshold at the same time. It is impossible to foresee this perfectly, because there is always the risk that recently rehearsed word pairs need to be rehearsed again, while there are still other word pairs that need to be rehearsed as well. Furthermore there is the question of how far the program should look ahead, especially since we do not know the reaction time of the user in advance. Besides, it is not a problem if the program makes a small error every now

and then if this prevents a lot of calculations for such a small issue. It is however a good policy to look ahead for more than one presentation so that the program will not be too late for several events in a row. This is demonstrated in figure 3.3.

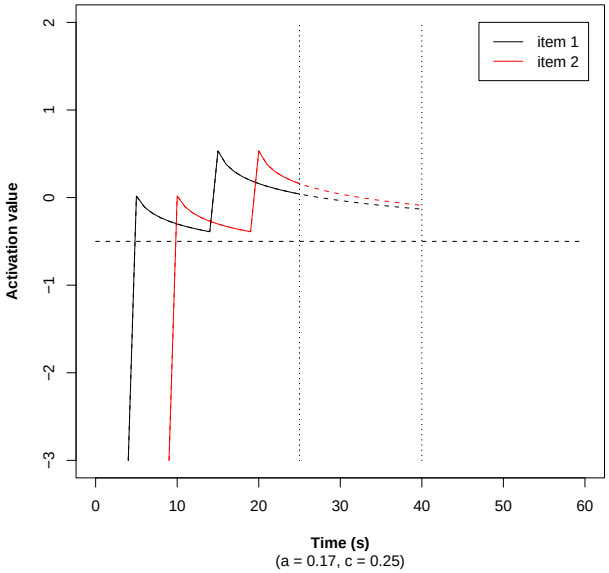


Figure 3.3: Look ahead time: how many seconds should the learning program look ahead to determine whether an old word pair should be rehearsed or a new one presented?

Here follows a summary about the dynamic learning schedule for a better understanding. The first step is to present a new word pair. Then there is some fixed time the program looks ahead to see whether the activation value of this item will still be above the threshold while presenting other items. If this is the case, then it is no problem to present a new item. If this is not the case, it is necessary to repeat the first word pair, until the activation value is high enough to support other items being presented on the screen. When there are several items in memory, all the previously presented word pairs need to be checked to see whether it is time to rehearse one of these word pairs or whether there is time to present a new word pair. When the activation values of all the items currently in memory are above the threshold and will remain so for several presentations, then a new item can be presented.

Finally there is the question what to do when all the word pairs have been presented at least once? Which item should be rehearsed when there are no more new items to present? An easy answer is to rehearse the word pair with the lowest activation value (even though it is above the threshold),

because we want the activation values of all the word pairs to be as high as possible. Another method to implement is to look for the word pair with the largest last encounter. This method results in the largest possible spacing for each word pair. Although the first implementation is more elegant, I opted for the latter one. The thought might arise to stop learning after the activation values of all the word pairs are high enough. But, of course, the activation values need to be high enough at the moment of test, not at the end of the learning period. Furthermore the dependent variable is the order of the word pairs and not the learning time. Therefore the learning time will be fixed and rehearsal continues despite the height of the activation values.

3.2 Method

In this paragraph the implementation of the experiment will be explained. The dynamic learning schedule, as explained in the introduction of this section, is used for the implementation of the computer program. The four different learning conditions, of which three are based on the spacing approach, are explained. And finally the parameters used for the spacing model and by the program are described.

Participants and design

Four different groups from three different high schools attended to this research project. The different learning conditions were spread among the subjects within each group. That is, each group had subjects that used one of the four conditions described below. This is important in order to minimize the effect that one class might be better than another class and thus influence the conditions used. The subjects were all 3^{rd} year students from either *havo* or *vwo* who attended French lessons. A list of 20 French-Dutch word pairs was used for each class, provided by their teacher. Each teacher was asked to provide a word list that the students had not seen so far (see Appendix B). During the program and the test the student only had to recall the Dutch word and not the other way around.

There are four learning conditions in this experiment. Three of them are Dynamic Spacing conditions (DS), of which two adjust the parameters to the performance of the subject (DS-R and DS-RT). There is one control condition (C) which is based on the flashcard method. More about the control condition later on. First a short review of the **Dynamic Spacing condition (DS)**. This condition is as described in the introduction of this section. The Dynamic Spacing condition applies to all of the three spacing conditions. This method works in the following way. Each word pair is presented to the subject one by one. Word pairs that are about the drop below the threshold will be rehearsed. When the activation values of the currently presented word pairs are above the threshold for some amount of time, a new word pair is presented. Finally, when there are no more new word pairs to be

presented, the word pair with the oldest last encounter is rehearsed.

The second condition is like the first, but adjust the spacing parameters to the (correct/incorrect) response of the subject. This is called the **Dynamic Spacing - Response condition (DS-R)**. According to the Dynamic Spacing condition, word pairs should be rehearsed before they are forgotten. But depending on the response of the subject, an answer during rehearsal could be incorrect. This means that the model predicted that the subject should have been able to recall the current word pair, but in practice the subject failed to respond correctly. In other words: the activation value of the current word pair should have been lower, namely below the threshold. This can be done by increasing the decay parameters for that item so that from now on the word pair will be rehearsed more quickly than was originally planned. Therefore, in case of an incorrect answer, the spacing parameter a (decay intercept) is increased by 0.01 for this specific word pair. The spacing parameter c (decay scale) and the threshold τ remain the same, because there parameters are more global. Furthermore the choice has been made to adjust at word pair-level and not at subject-level, because during such a short learning session the feedback contains more information about the word pair than about the subject. The reverse of the above mentioned adjustment also applies: when the activation value of an item has dropped below the threshold, but the subject responds correct nonetheless, the decay intercept a is decreased by 0.01 for that word pair. Although this shouldn't happen it is impossible to rehearse all the word pairs on time, especially when there is a variable response time (up to 15 seconds) depending on the subject. When the subject in such a case provides an correct answer anyway, then the decay parameters must have been too high. In either case only the decay intercept a is adjusted and not the decay scale parameter c. See figure 3.4 for an example of what happens after an incorrect response.

The third of the Dynamic Spacing conditions is like the first two, but adjusts the decay parameters depending on the reaction time of the subject. Therefore this method is called the **Dynamic Spacing - Reaction Time condition (DS-RT)**. As explained in the section about the ACT-R declarative module, the latency of retrieval can be calculated based on the current activation value of an item (see formula 2.4). This latency can be used to make a prediction about the reaction time of the subject. A difference can be measured by comparing the reaction time of the subject on a rehearsal and the calculated reaction time based on the activation value of an item in the model. If this difference is small, nothing should be done. But if this difference is large, there has apparently been an incorrect estimation of the activation value. When the subject responds faster than the predicted reaction time, the real activation value for that item should have been higher. This means that the decay value must have been too high, because the activation value by the model has been too low. This can be adjusted by decreasing the decay parameters (again, only the decay intercept a), so that the activation value

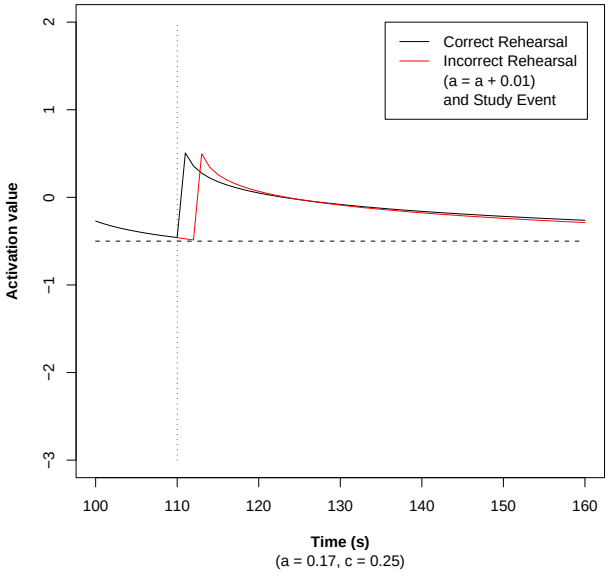


Figure 3.4: Adjustment of the a parameter in condition DS-R after incorrect response.

for that item will be higher from now on. In this case the decay intercept is decreased by the difference in seconds divided by 1000, but only when the difference is greater than a $\frac{1}{2}$ second. Furthermore the difference in reaction time has a limit of 10 seconds, so that changes to the decay intercept cannot be greater than 0.01. The reverse case also leads to adjustment of the decay intercept. When the reaction time predicted by the model is greater than the reaction time of the subject (and differs more than a $\frac{1}{2}$ second), the decay intercept is increased by the difference / 1000. Also, when an answer is incorrect an increase of 0.01 is given to the decay intercept of that word pair, as if the subject had a reaction time of 10 seconds.

The last of the four conditions is the **Control condition (C)**. This condition is like a flashcard method in which only the incorrect word pairs are repeated again. Therefore the spacing model is *not* used in this method. But to get a fair comparison between this condition and the spacing conditions, the word pairs are presented in sets of 5. The word order in this condition is very simple. First present 5 word pairs (and immediately rehearse these word pairs after the first presentation). Then rehearse only the word pairs that have been recalled incorrectly. After this rehearse the word pairs that have been recalled incorrect for a second time. Repeat this process until all the word pairs are recalled correctly exactly once. Then present a new set of 5 word pairs and rehearse until every word pair has been rehearsed correctly. When the entire set of words has been rehearsed, restart at the beginning

and rehearse the word pairs in sets of 5 again. This is basically the method used in the experiment by Bahrick (1979), but now the word pairs are rehearsed in sets of 5 instead of repeating the entire word list at once. The different learning conditions are summarized in table 3.1.

Condition	**Description**
DS	A dynamic learning schedule using the spacing model of ACT-R.
DS-R	The same as DS, but adjust the spacing decay (intercept) parameter upon an incorrect answer of the subject.
DS-RT	The same as DS, but adjust the spacing decay (intercept) parameter upon the reaction time of the subject.
C	Control condition in which incorrect word pairs are rehearsed (per 5 word pairs).

Table 3.1: The different learning conditions.

Procedure

The learning program displays either a study or rehearsal trial. Each new word pair is presented for 5 seconds on the screen. For each rehearsal a subject has up to 15 seconds to reply. As soon as the subject presses Enter (or when the 15 seconds have passed) the user gets a response whether the answer is correct or not. This response is shown for 2 seconds on the screen and shows either a "correct", "incorrect" or an "almost correct" message. The "almost correct" message is shown when the answer has a Levenshtein distance smaller than 3 (and greater than 0; otherwise it would be correct). The Levenshtein distance calculates the difference between two strings based on the insertion, deletion or substitution of a single character. For example, the strings "house" and "hows" have a Levenshtein distance of 2 (one substitute and one delete operation). Almost correct answers are counted as incorrect answers, but it provides the user feedback on small typing errors. After an incorrect message, there is always a study trial of 5 seconds which contains the correct answer. After a correct message, the next word pair to be presented is determined by the learning condition as described above.

For determining the next word pair in the Dynamic Spacing conditions, there is a look ahead time of 15 seconds. As explained above there needs to be some time to see whether a word pair is about to be forgotten. It is a good policy to look ahead for several presentations, because several word pairs might drop below the threshold at the same time. Although most users will either type an answer or press Enter before the 15 seconds response time, it is good to assume the worst case scenario and support the possibility that a user might wait for 15 seconds. Furthermore having a look ahead time of

15 seconds creates space for 3 presentations in a row, which seems to be enough - and not too much - for determining whether a word pair should be rehearsed or a new word pair can be presented.

Choosing the model parameters for the Dynamic Spacing conditions, was a bit more difficult. Because of the short learning time and the preference for small initial spacing, the following parameters were derived: decay intercept $a = 0.25$, decay scale $c = 0.25$ and threshold τ = -0.5. Choosing smaller decay parameters will result in a sequence in which a lot of word pairs (up to 10 or 20!) are presented before rehearsal of earlier word pairs start. It is, however, unlikely that someone will remember more than 5 word pairs before starting rehearsal. After several test, these parameters provided an acceptable word order in which rehearsal of the first word pair started before the presentation of the fifth word pair. Such a limit is important, because the program should rehearse word pairs before they are forgotten. A learning schedule in which 10 or 20 word pairs will be presented before rehearsal starts is not very realistic when learning new word pairs.

The program has a fixed learning time of 15 minutes, after which the learning session is ended by the program. A total of 20 unique word pairs are rehearsed within these 15 minutes. The list of word pairs is randomized before the start of a session in order to prevent a learning effect based on the order of the word list. The next day a test was taken on paper too find out how many word pairs where correctly remembered. The score on this test is the dependent variable of this experiment.

Program parameters	study time = 5s, rehearsal time = 0 to 15s, feedback time = 2s, look ahead time = 15s
Model parameters	$a = 0.25$, $c = 0.25$, τ = -0.5, noise $s = 0.255$, latency $F = 1$

Table 3.2: The parameters used in this experiment.

3.3 Results

For this experiment 4 groups of students from three different high schools were tested. Each group was from the 3^{rd} year of *havo* or *vwo* and attended French lessons. The list of word pairs for each group was selected by their teacher and consisted of word pairs from a chapter that the students had not seen before. Within each group the four conditions were spread among the students, so that each group had about 5 students for each condition. The learning program allowed the students to learn for 15 minutes, after which the program ended. Within these 15 minutes the students rehearsed up to 20 word pairs in a word order depending on the learning condition. Some students did not see all of the 20 word pairs, but more about this later. One day after the learning session the students

performed a written test in which they had to recall the Dutch translations of the French word pairs. Below is a table which shows per condition the number of subjects, the mean scores on the test, the mean grades on French and the average number of unique word pairs seen.

	DS	**DS-R**	**DS-RT**	**C**	(mean)
Number of subjects	21	21	20	20	(20.5)
Mean score on test	**7.40**	**8.24**	**9.05**	**8.07**	**(8.19)**
Mean grade on French	6.46	5.95	6.46	6.67	(6.39)
Number of unique WPs seen	19.33	19.57	19.55	20.00	(19.61)

Table 3.3: The results of the experiment.

In total 82 subjects attended the experiment correctly. This is after rejection of 9 subjects. Three of the rejected subjects were very slow based on the criterion that they either waited 15 seconds or pressed Enter without an answer in more than 10% of the rehearsal events. Another three subjects were removed, because they had a score that was lower than the mean minus 2 times the standard deviation of that condition. Finally there were three subjects that attended the learning session, but were not present at the time of the test.

As can be seen in the table above, the number of subjects was equally divided among the different conditions. Furthermore the mean grades on French of the subjects within each condition is also fairly equal. Note that the grade on French is not a true indicator of how well a student can learn French word pairs, because the grade consists, for example, also out of learning grammar. The fact that grade is not very representative for the score on learning French word pairs, is supported by the low correlation of 0.18 between the two variables. The grade however does give an indication of how good students are on French and whether these students are equally divided among the conditions. Therefore the grade can be used to correct for variance in the ANOVA due to this variable.

Unfortunately the number of unique word pairs seen by the subjects was not the same. Table 3.3 shows that in the Dynamic Spacing conditions, the subjects saw on average 19.5 word pairs. This was due to the short learning time of 15 minutes (see Appendix A for more detail). While some students from the DS conditions had seen 20 unique word pairs within these 15 minutes (with an average RT of 4.7 seconds), other students had seen only 18 or 19 unique word pairs (with an average RT of 5.9 seconds). Subjects in the control condition always saw 20 unique word pairs (with an average RT of 5.1), because the end of the word list is reached quicker. The average reaction times per condition can be seen in table 3.4. These are the reaction times between the start of a rehearsal event and the end of a rehearsal event. A rehearsal event ends either by pressing the Enter key or after 15 seconds.

	DS	**DS-R**	**DS-RT**	**C**	(mean)
RT for subjects who saw < 20 WPs	5.95	6.15	5.69	-	(5.93)
RT for subjects who saw 20 WPs	4.79	4.53	4.68	5.14	(4.79)
Average reaction times	**5.23**	**4.92**	**4.94**	**5.14**	**(5.06)**

Table 3.4: The average reaction times per condition.

The question to be answered is whether there is a difference between the scores based on the learning condition. The appropriate test for this question is a between-subjects ANOVA. An ANOVA shows whether there is a significant difference between learning conditions, based on the means and standard deviations. If the p-value of this test is very small, than the chance that the conditions are equal can be rejected. If there is a difference between the conditions, further tests have to be performed to see wherein this difference lies. In this case when looking for a difference between the learning conditions, the ANOVA results is $F(3, 78) = 3.83$ with $p = 0.013$*. This means that the score on the test depends on the condition used the day before. Figure 3.5 shows the average scores per learning condition.

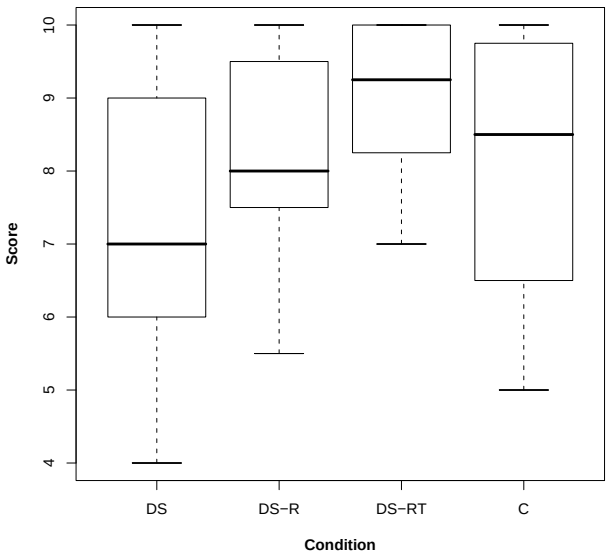


Figure 3.5: The average scores per condition.

The ANOVA shows that there is a difference between the learning conditions. The hypothesis that the learning conditions are equal can be rejected. The next question is wherein the difference lies? But before this question is answered, it is good to look at the effect of the grades on the score. This

is because good students are likely to score higher at this test than other students. Therefore grade can be used as an extra independent variable, so that the variability due to the difference in quality of learning is reduced. When performing an ANOVA with correction for grade, the results are $F(3, 77) = 3.95$ with $p = 0.011$* for the learning condition used and $F(1, 77) = 3.35$ with $p = 0.071$ for the grade of the subjects on French. As can be seen, there is still a difference between the learning conditions. The grade, however, does not predict the score on the test. This is not so unexpected, because a grade on French depends on more than learning word pairs alone. Furthermore the correlation (0.18) between grade and score was small as well (see table 3.5). It is nice to see that the correlation between grade and score is by far the highest for the tradition control condition. To get an idea of what the performance on the test is when compensating for the grade on French, another plot can be made. Figure 3.6 shows the residuals of an ANOVA after correction for grade. This means that part of the noise is explained by how well a subject performs on French. What is left is the difference between the conditions without the variance explained by grade.

	DS	**DS-R**	**DS-RT**	**C**	(mean)
correlation between grade and score	0.18	0.26	-0.08	0.45	(0.18)

Table 3.5: How well does the grade on French predict the score on the test?

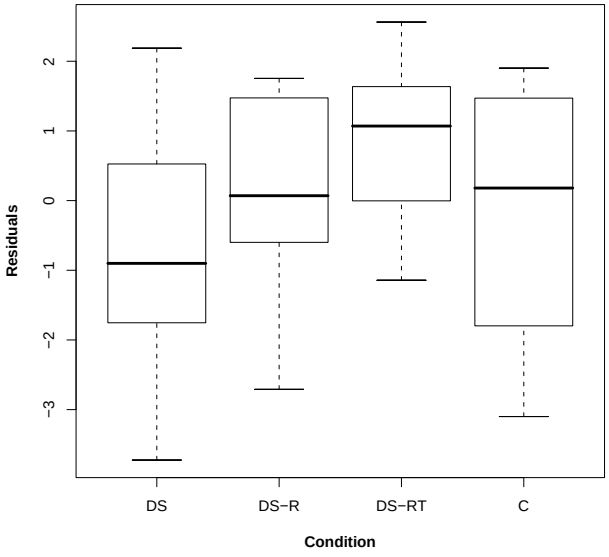


Figure 3.6: The residuals of the scores after correction for grade.

To find out wherein the difference between the learning conditions lies, further tests have to be taken. In this case several t-tests are performed. For these t-tests, the residuals are used after correction for grade. Furthermore I want to compare the difference between the three Dynamic Spacing conditions, as well as the difference between the Dynamic Spacing conditions and the control condition. This is to find out whether some of the Dynamic Spacing conditions perform better than the others and to find out which of the spacing conditions are better than the control condition. When looking at figure 3.6 it is obvious that if a difference between a spacing and the control condition is to be found, that it has to be between DS-RT and C.

First of all I am interested in the difference between the Dynamic Spacing conditions. Since two of these methods (DS-R and DS-RT) adjust the decay intercept parameter a to the subjects, an increase in performance might be expected. As can be easily seen in figure 3.6, there is an increase in performance from DS to DS-R and from DS-R to DS-RT. This is not very surprising, since method DS-R repeats word pairs that are incorrectly recalled more often, and method DS-RT is even more fine tuned, because it acts on the reaction time of the subject, even when an answer is correct. The question remains whether the difference between these three methods is significant. When performing a t-test between DS and DS-R this results in a t-value of -1.92 ($p = 0.063$, $df = 36.17$). Therefore, there is no difference between the first two Dynamic Spacing conditions. The t-test between DS and DS-RT provides t = -3.53 ($p = 0.001$*, $df = 32.40$). This means that subjects in condition DS-RT score higher than subjects using condition DS. Finally the difference between DS-R and DS-RT which has a t-value of -1.88 ($p = 0.068$, $df = 38.1$). Therefore, the adjustable methods DS-R and DS-RT do not differ enough to say that one method is better than the other, when comparing the score on the test.

Second of all I am interested in the difference between the Dynamic Spacing conditions and the control condition. Using the graphs, it is quite obvious that if there is a difference to be found, it should be between the best spacing condition (DS-RT) and the control condition (C). When testing on this difference, the following t-statistic is found: $t = 2.22$ ($p = 0.033$*, $df = 31.31$). This difference is significant, thus the spacing method DS-RT provides students a higher result on the test than when using the control condition C. When looking for further difference between the Dynamic Spacing methods and the control condition, this results in t = -1.11 ($p = 0.275$, $df = 39$) for the t-test between DS and C, and $t = 0.66$ ($p = 0.511$, $df = 35.08$) for the difference between DS-R and C. That there is no difference between DS-R and C is not surprising, since these conditions show a similar result. The control condition might have been better than the Dynamic Spacing method DS, but this also is not the case.

In short, there are differences to be found on the scores when using a certain learning conditions. The differences found in this experiment are between the Dynamic Spacing condition DS-RT, the Dynamic Spacing condition DS and the control condition C. The Dynamic Spacing condition DS-RT adjusts the decay value depending on the reaction time of the user. This adjustment seems to provide a better result than when using Dynamic Spacing alone. Furthermore, using the spacing method in combination with the adjustment on reaction time provides a better result than learning the word pairs in sets of 5 (and repeating only the incorrect word pairs).

To understand the mechanism behind the DS-RT method a little bit better, a short analysis of the decay intercept parameter (a) has been performed. Because the control condition does not use the spacing model, this condition is left out of the analysis. In the standard Dynamic Spacing condition, the decay parameters (a and c) do not change at all, so there is also no point at looking at this data. What is, however, interesting to look at is the development of the a parameter for the DS-R and DS-RT condition. For this I had to retrieve the a-values of each word pair at the end of the training. This will be called the final a-values, as it refers to the parameter value at the last encounter of a word pair. Figure 3.7 shows the final a-values of the word pairs per condition. Figure 3.8 shows also the final a-values, but this time averaged per subject. As can be seen, the adaptive Dynamic Spacing conditions have the tendency to increase the a-value, but when averaging per subject the mean a-value in the DS-R condition ends up a bit lower than 0.25.

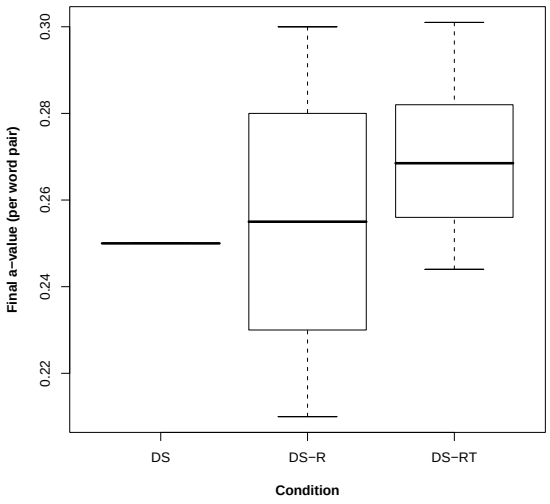


Figure 3.7: The final a-value of each word pair at the last encounter during study.

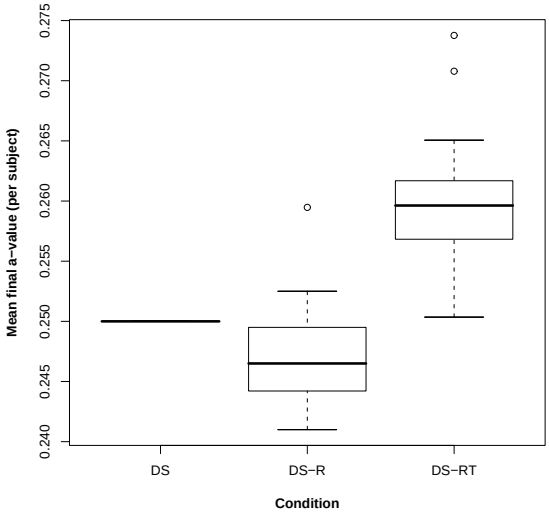


Figure 3.8: The final a-values averaged per subject.

Finally some more elaboration about the number of word pairs that a subject has seen. As mentioned earlier, when using one of the Dynamic Spacing conditions, not all subjects saw 20 unique word pairs, but rather 18 or 19 word pairs. This is especially true for method DS, in which 9 out of the 21 subject did not see 20 unique word pairs. In condition DS-R and DS-RT there where both 6 subject who saw less than 20 unique word pairs. To use the number of unique word pairs as an extra independent variable, would not be correct, because correcting for the number of word pairs will be beneficial for the Dynamic Spacing conditions and a disadvantage for the control condition. Furthermore, not seeing all the word pairs is a side-effect of the Dynamic Spacing conditions and should be taken into account. What can be done, however, is to show if the method used determines how many word pairs a subject has seen. This can be done by another between-subject ANOVA in which the number of unique words is used as the dependent variable. Such an ANOVA results in $F(3, 77) = 3.05$ with $p = 0.034$*, which means that the number of word pairs seen does depend on the condition used during study. In this case there was also a correction for grade: $F(1, 77) = 1.43$ with $p = 0.235$, but this is simply to correct for some variability.

Further t-tests can be performed to find out which learning conditions differs when predicting the number of word pairs seen during learning. But it is quite obvious that the difference will be between the Dynamic Spacing conditions and the control condition. This is indeed the case, because only the difference between DS and C (t = -3.33, $p = 0.003$*, $df = 20.29$), DS-R and C (t = -2.22, $p = 0.038$*, $df = 20.4$), and DS-RT and C (t = -2.35, $p = 0.03$*, $df = 19.31$) show a significant effect. Thus when using one of the learning conditions, you will probably see less unique word pairs than when using the control condition. This is therefore a disadvantage of the Dynamic Spacing conditions.

3.4 Conclusion

As was shown in the results, the score on the test depends on the learning condition used. In this case method DS-RT differs from DS, and DS-RT differs from C. This means that Dynamic Spacing provides a higher score than the control condition, but only when the program adjusts the decay parameters to the subjects based on their reaction times. Also, when using Dynamic Spacing, it is better to adjust the decay parameters upon reaction time than to keep the decay parameters the same.

Note that although the Dynamic Spacing condition DS-RT provides a higher score, the spacing methods, as used in this experiment, have a disadvantage. This has to do with the fact the new word pairs are only presented when the old word pairs are remembered good enough. Given the short learning time of 15 minutes, it is - in the Dynamic Spacing conditions - possible that some students do not see all of the 20 unique word pairs. This is usually due to high reaction times or the number of

incorrect responses of the subject. Note that this is a side-effect of the Dynamic Spacing conditions. However, despite this disadvantage, method DS-RT still performs better than the control condition. This is quite amazing!

Furthermore, the performance of subjects in learning condition DS-RT seems to be very stable. Not only does this method have the smallest standard deviation, but there are also some other reasons why this method is stable. When looking at the grades, for example, as a prediction of the score, this method had the lowest correlation. The control condition on the other hand had the highest correlation between grade and score and would therefore suggest that this is a good comparison as to how students learn at school.

Another important issue is that, when looking at the data of 3 different high schools, there seems to be no abnormalities. Only a few subject where removed from the data, the average grades where almost equal in each condition and the groups where all from the same level and year of study. The word lists used were taken from the study material of the group and from a chapter that the students had not worked with so far. The experiment has therefore been run in a very real-life situation.

Finally it can be noted that the score on the DS-RT method is, an average, one point higher than the score on the control condition. This is also quite amazing, because the subjects only studied for 15 minutes and the word order within these 15 minutes does seem to matter a lot! Of course further experiments can be done to get more stable results and to fine-tune the program, but the condition DS-RT scores very well nonetheless. And last but not least, when using this learning program for 15 minutes the day before a test provides an average score of 8*, which is not bad either.

*if you are a 3^{rd} year *havo/vwo* student who attends French lessons.

Chapter 4

Discussion

The question of this thesis was how to effectively learn word pairs in the short term, in this case with the aid of a computer program. To answer this question one has to look at the frequency and recency of the presented word pairs. Also the spacing (number of intervening items) between presentations seems to have an impact on the score at the time of test. In a wide variety of memory tasks it is true that distributing learning material over time has a positive effect on memory in the long term. Therefore, the spacing effect is another component to keep in mind when creating an effective learning schedule.

In the introduction of this experiment it was explained that finding an optimal learning schedule is not as easy as it first seems. For example, the number of rehearsals of each word pair varies, depending on the learning time and the number of word pairs to be learned. It is therefore a good policy to have these last two variables fixed when looking for an optimal learning schedule. Another issue is that active rehearsal is more beneficial than passive learning. Thus, using the maximum amount of spacing between two rehearsals is not a good option, because then the first word pair will be forgotten by the time the last word pair has been presented. This means that finding an optimal learning schedule has to do with repeating word pairs before they are forgotten.

The results of the experiment show that learning in an incremental Dynamic Spacing way is indeed more beneficial than a standard flashcard method. This is only true if the learning program adjusts the decay intercept parameter a (per item) to the reaction times of the user. The grade of a subject or the difficulty of a word pair does not seem to have a strong effect on the DS-RT learning method. There is, however, a disadvantage of using one of the Dynamic Spacing methods. The problem is that when learning for such a short time, some of the users will not be able to see all of the word pairs. This is because a slow reaction time or a lot of incorrect rehearsals will lead to postponing the presentation of new items. The control condition has this problem too, because repeating the same mistake will

prevent a user from advancing to the next set of 5 word pairs. In this experiment, however, subjects in the control condition did see all of the 20 word pairs, even though some participants did not have many repetitions.

A thing to note for future work is to make sure every participant sees the same number of unique word pairs. Keeping the number of rehearsals the same for each user is a very hard thing to do, because this depends on the answers and the reaction times of the user. For example, a long reaction time will result in less repetitions given a fixed learning time. An incorrect answer will result in an extra study event, which also decreases the number of rehearsals. What can be done, however, is to shorten the maximum time of a rehearsal event. In this case students had up to 15 seconds to respond, which resulted for some students in a low reaction time. Decreasing the maximum response time of a rehearsal event will increase the number of repetitions within a learning session, and this might help seeing all the word pairs on time.

Another method to make sure all the word pairs are seen on time is simply to increase the learning time of a session. But this solution is not very convenient. First of all, if you want to test memory using a short learning session, you do not want to increase the session time. Second of all, there might arise a ceiling effect in the results on the test, because some students have time to learn the word pairs even better than they already did. Decreasing the number of word pairs to be learned will have the same effect, because students in the DS-RT condition already scored a 9 out of 10 an average. It is therefore better to limit the response time, so that students will continue to the next word pair in stead of waiting for a long time. This does not solve the problem that incorrect answers will lead to less repetitions, but mistakes cannot be avoided. Repeating the same mistake is especially a problem for the control condition, because this will prevent the program from advancing to other word pairs.

Interesting recent work has been done by Pavlik and Anderson (2008). Their research is also about finding an optimal schedule of practice. Their optimized learning schedule was also a dynamic method which repeated word pairs with a low activation value. One of the differences with the experiment in this thesis is the decision boundary of when to start rehearsing. In stead of using a look ahead time, they used an extra threshold to determine when a word pair should be rehearsed. Furthermore, they used another threshold to distinguish between study rehearsals (very low activation) and drill rehearsals (low activation). Using an extra threshold in stead of a look ahead time can be a disadvantage when using the spacing model, as explained in the introduction of this experiment. Presenting a study event in stead of a rehearsal event for items that are very near the threshold, is a smart mechanism for preventing incorrect rehearsals. This will increase the number of correct rehearsals, and therefore reap the benefit of active rehearsal. This mechanism is thus a recommendation for future work.

An important difference between this research and the one by Pavlik and Anderson (2008) concerns the learning and retention time. Their experiment had three (relatively long) learning sessions and a retention time of one week. In this experiment the participants only had a very short time to learn (i.e. 15 minutes) and were tested the next day. Because of the short learning time in this experiment, there was a need for small initial spacing. This was done by using higher initial parameters to get a more realistic word sequence. As was shown in the results, the a-parameter might even have been a little bit higher. Pavlik and Anderson (2008) also mention that some (like Jost) advocate smaller initial spacing, but that there is still a debate going on about this issue. It will, however, be good to look at this issue, especially when learning for only a short time period.

Another important difference between the two research projects concerns the adaption at item and participant level. In this research the decay intercept a changed in the DS-R and DS-RT condition. This change was only at the item level. The model in Pavlik and Anderson (2008) has three extra parameters to correct for the difference in participant, the item and the participant/item interaction. It is indeed a good idea to correct for participant as well. But this means we need to have some information about the participant to estimate an initial parameter value. Given the model from this experiment, it would be possible to start a new learning session with the average a-value of the word pairs at the end of the previous session. This way the model can also correct for differences in participant. But when having only one short learning session, it is difficult to adjust for the participant during the learning session.

One of the recommendations for future work concerns the difference between active and passive rehearsal. In this case I used the knowledge about the benefit of active rehearsal by repeating the word pairs (in the Dynamic Spacing conditions) before they are forgotten. The implemented spacing model, however, did not distinguish between active rehearsals (correct response) and passive rehearsals (study event). This is definitely something to have in mind when creating a smart learning schedule, because active rehearsals are more beneficial and this will separate the difficult from the easy items. Increasing the activation of a correct rehearsed item more than a studied item, for example, will result in remembering this item for a longer time period. This does not only increase the spacing of that item, but it will also help to present new word pairs earlier in the sequence of repetitions. Furthermore, it is more realistic to distinguish between active and passive rehearsals, but implementation of this feature will have to wait for future work.

Finally some more general recommendations. As was shown in this experiment, when learning word pairs in the short term, using the Dynamic Spacing DS-RT condition has a benefit over the traditional learning condition. This experiment was done in a typical real-life situation using data

from three different high schools. Learning programs on rehearsing word pairs could therefore benefit from the method as described in this thesis. Although some fine tuning of the parameters will be needed, the Dynamic Spacing method can be an excellent tool for increasing the effectiveness of word pair learning. Furthermore the spacing model is quite general when it comes to learning facts. This method can thus also be used for other facts and domains, like learning the chemical elements or geographical places. Educational programs on learning facts can therefore benefit from this spacing method, and mostly those programs that help to study word pairs just the day before the exam.

Bibliography

Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press.

Anderson, J. R., Bothell, D., Byrne, M., Douglass, D., Lebiere, C., and Qin, Y. (2004). An integrated theory of mind. *Psychological Review*, 111(4):1036–1060.

Anderson, J. R. and Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science*, 2(6):396–408.

Bahrick, H. P. (1979). Maintenance of knowledge: Questions about memory we forgot to ask. *Journal of Experimental Psychology: General*, 108(3):296–308.

Dempster, F. N. (1987). Effects of variable encoding and spaced presentations on vocabulary learning. *Journal of Educational Psychology*, 79(2):162–170.

Dempster, F. N. (1989). Spacing effects and their implications for theory and practice. *Educational Psychology Review*, 1(4):309–330.

Ebbinghaus, H. (1885/1964). *Memory: A Contribution to Experimental Psychology (Über das Gedächtnis)*. New York: Dover Publications. (Original work published in 1885. Translated by Henry A. Ruger and Clara E. Bussenius in 1913).

Glenberg, A. M. (1976). Monotonic and nonmonotonic lag effects in paired-associate and recognition memory paradigms. *Journal of Verbal Learning and Verbal Behavior*, 15(1):1–16.

Glenberg, A. M. (1979). Component-levels theory of the effects of spacing of repetitions on recall and recognition. *Memory and Cognition*, 7(2):95–112.

Greene, R. L. (1989). Spacing effects in memory: Evidence for a two-process account. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15(3):371–377.

Hintzman, D. L. (1974). Theoretical implications of the spacing effect. In Solso, R., editor, *Theories in cognitive psychology: the Loyola symposium*, pages 77–99. Lawrence Erlbaum.

McGeoch, J. A. (1943). *The Psychology of Human Learning*. New York: Longmands Green.

Melton, A. W. (1970). The situation with respect to the spacing of repetitions and memory. *Journal of Verbal Learning and Verbal Behavior*, 9(5):596–606.

Pavlik, P. I. and Anderson, J. R. (2005). Practice and forgetting effects on vocabulary memory: An activation-based model of the spacing effect. *Cognitive Science*, 29(4):559–586.

Pavlik, P. I. and Anderson, J. R. (2008). Using a model to compute the optimal schedule in practice. *Journal of Experimental Psychology: Applied*, 14(2):101–117.

Raaijmakers, J. G. W. (2003). Spacing and repetition effects in human memory: Application of the SAM model. *Cognitive Science*, 27(3):431–452.

Rumelhart, D. E. (1967). The effects of interpresentation intervals on performance in a continuous paired-associate task. Technical Report 16, Institute for mathematical studies in social sciences, Stanford University.

Seabrook, R., Brown, G. D., and Solity, J. E. (2005). Distributed and massed practice: From laboratory to classroom. *Applied Cognitive Psychology*, 19(1):107–122.

Underwood, B. J. (1970). A breakdown of the Total-Time law in free-recall learning. *Journal of Verbal Learning and Verbal Behavior*, 9(5):573–580.

Young, J. L. (1971). Reinforcement-test intervals in paired-associate learning. *Journal of Mathematical Psychology*, 8:58–81.

Appendix A

Performance of Participants

As can be seen in this section, some of the students in the Dynamic Spacing condition do not see all of the 20 word pairs (see figure A.1). Students in the control condition do see all of the word pairs, but on bad performance can repeat the same set of word pairs over and over again (see figure A.2).

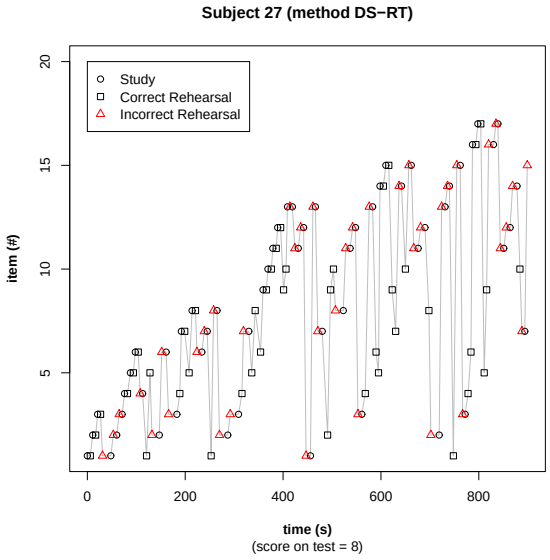


Figure A.1: The learning schedule of a student who performs bad in the DS-RT condition.

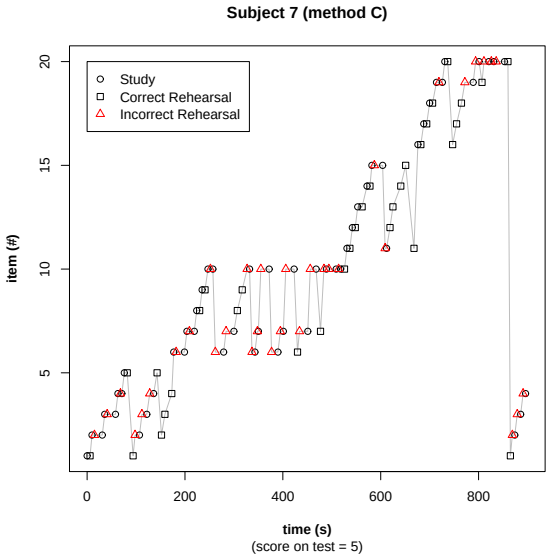


Figure A.2: The learning schedule of a student who performs bad in the control condition.

When a student performs well, he or she will see all of the 20 word pairs in the Dynamic Spacing condition (see figure A.3). A student who performs well in the control condition will see the 20th word pair earlier in the sequence (around 500 seconds), but on the other hands starts repeating correct word pairs at a later point in time (see figure A.4).

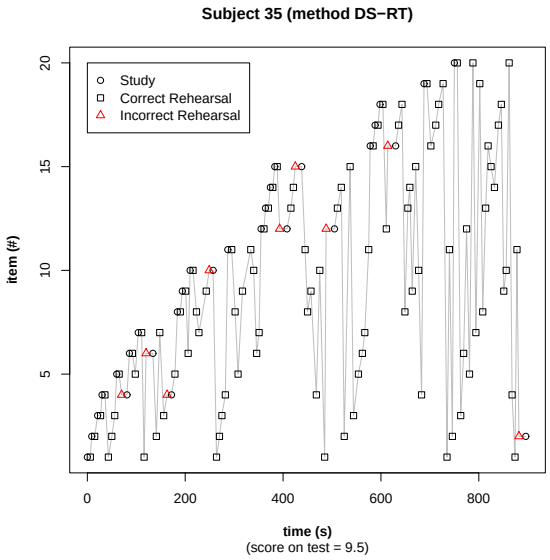


Figure A.3: The learning schedule of a student who performs well in the DS-RT condition.

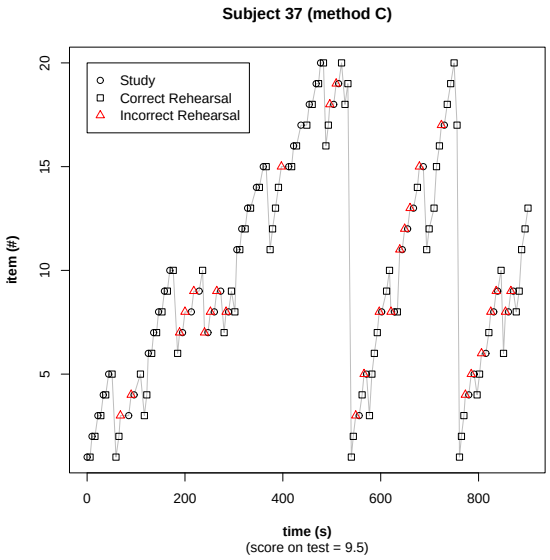


Figure A.4: The learning schedule of a student who performs well in the control condition.

What is interesting to see is that when a student performs very well, the control condition also provides an optimal spacing (see figure A.6), although some of the word pairs are still forgotten. In the DS-RT condition (figure A.5) the same sequence is repeated after about 500 seconds, but this is a bit more difficult to see.

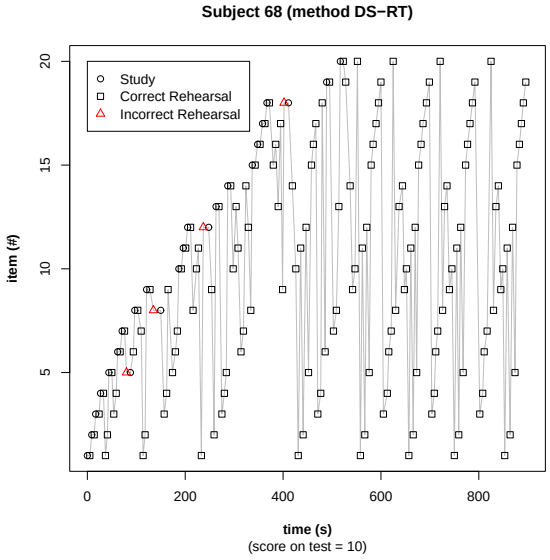


Figure A.5: The learning schedule of a student who performs extremely well in the DS-RT condition.

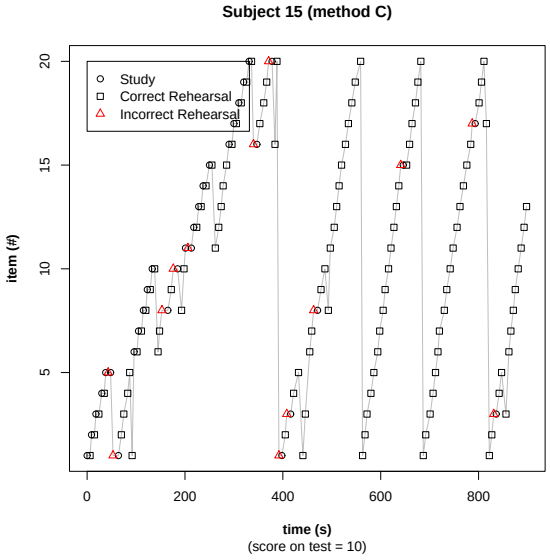


Figure A.6: The learning schedule of a student who performs extremely well in the control condition.

Appendix B

Word Lists

French	**Dutch**
age de ... ans	... jaar oud
la course	de wedstrijd
permettre	toestaan
decouvrir	ontdekken
la facon	de manier
la curiosite	de bezienswaardigheid
durant	tijdens
se composer de	bestaan uit
l'institution	de instelling
les transports en commun	het openbaar vervoer
se situer	zich bevinden
la decision	het besluit
developper	ontwikkelen
lointain	verafgelegen
le passager	de passagier
obliger	verplichten
afin de	om te
le bilan	het eindresultaat
le VTT	de mountainbike
le voyou	de schooier

Table B.1: Word list 1.

French	**Dutch**
la guerre	de oorlog
nuisible	schadelijk
dsagrable	onaangenaam
la mouche	de vlieg
cultiver	kweken
efficace	doeltreffend
la fourmi	de mier
l'avance	de voorsprong
le vignoble	de wijngaard
superflu	overbodig
menacer	bedreigen
traiter	behandelen
en apparence	schijnbaar
souffler	blazen
la croissance	de groei
la hausse	de stijging
soutenir	steunen
agir	handelen
ignorer	niet kennen
la disparition	de verdwijning

Table B.2: Word list 2.

French	**Dutch**
la direction	de richting
bricoler	knutselen
l'expérience	de ervaring
l'apprentissage	de scholing
parfois	soms
la preuve	het bewijs
se situer	zich bevinden
regulièrement	regelmatig
se distraire	zich ontspannen
déranger	storen
rendre heureux	gelukkig maken
l'étude	de studie
suffisant	voldoende
étonné	verbaasd
redoubler	blijven zitten
mettre de côté	opzijleggen
l'employeur	de werkgever
embaucher	aannemen
prendre l'air	een luchtje scheppen
sécher les cours	spijbelen

Table B.3: Word list 3.