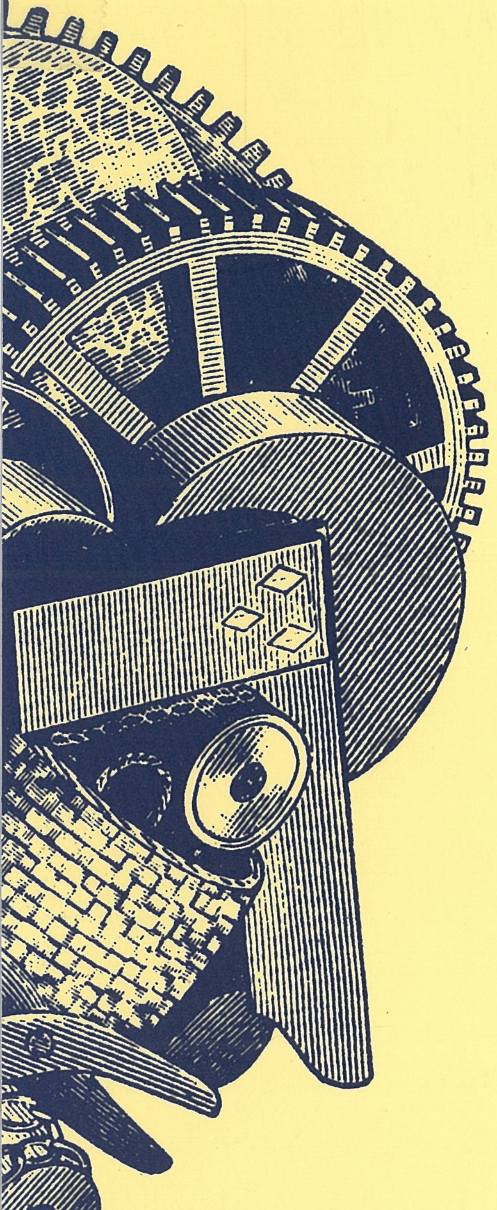
An agent-based interactive instruction system

DOI:
10.6100/IR475538

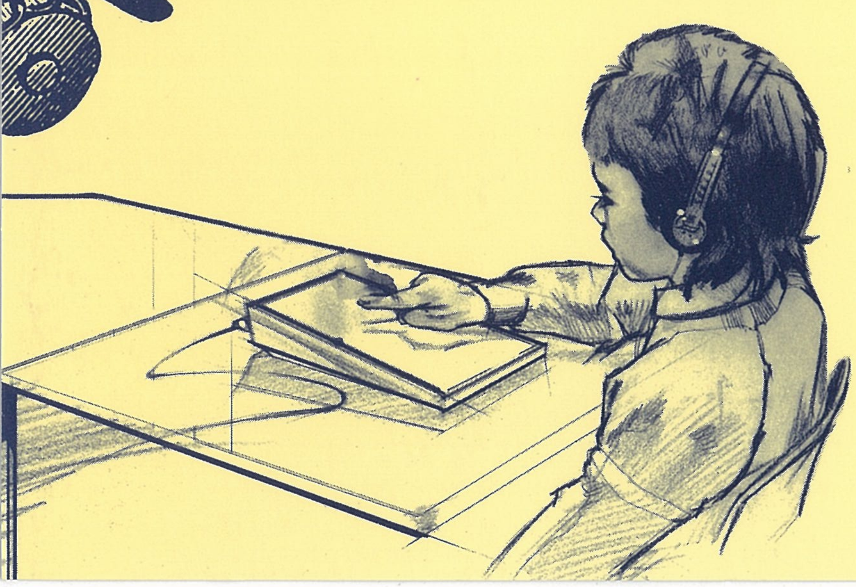
Document status and date:
Published: 01/01/1997

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)



An Agent-Based Interactive Instruction System

Judith Masthoff



An Agent-Based Interactive Instruction System

Masthoff, Judith Françoise Maria

An Agent-Based Interactive Instruction System /
Judith Françoise Maria Masthoff.
Thesis Eindhoven University of Technology.
ISBN 90-386-0319-3
Subject headings: agents / interactive instruction

Cover: The head originates from 'the inventor', a 19th century woodcut by Poyet. The learner is an artist impression by Gus Rodriguez.

An Agent-Based Interactive Instruction System

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Eindhoven,
op gezag van de Rector Magnificus, prof.dr. M. Rem,
voor een commissie aangewezen door het College van Dekanen
in het openbaar te verdedigen op
woensdag 15 januari 1997 om 16.00 uur

door

Judith Françoise Maria Masthoff

Geboren te Geleen

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr. D.G. Bouwhuis
en
prof.dr. A. Vandierendonck

Copromotor: dr. R.R.G. van Hoe

for Corry

Acknowledgments

I would like to thank the undergraduate students Annemie Melis, Barbara van de Par, and Mona Claessens for their help in running the experiments, and Birgit Albertmeier and Joseé Zielman for their help in the design and implementation of the Square Dance course. I thank Herman ter Horst, Paul van Loon, and Jaap Aeilkema for their various contributions to other aspects of my work.

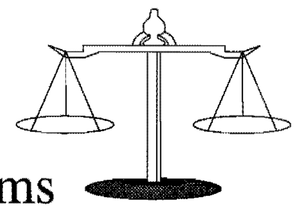
The help and cooperation of the staff of the Christiaan Huygens College in Eindhoven, and particularly Mr. van der Horst and Mr. de Graaf, has been essential for carrying out some of the experiments described in this thesis.

Finally, I would like to thank all the people who have made IPO such a stimulating environment for me.

Table of contents

Chapter 1

An evaluation of current interactive instruction systems



Abstract
An overview is given of the requirements to be met by an interactive instruction system. These vary from the basic requirements of any software system, like robustness and speed of response, to the requirements of any dialogue system, like transparency and learnability, to more specific requirements like a mixed locus of control and adaptation to the learner. Current approaches to the design of interactive instruction systems, namely computer-assisted instruction, intelligent tutoring systems, microworlds, and interactive learning environments, are evaluated in terms of these requirements. Among other things, it turns out that (1) striving for adaptivity tends to be at the expense of the more basic requirements, that (2) generativity tends to be poor, and that (3) a mixed locus of control is lacking.

1.1 Why do we need interactive instruction systems?

Learners prefer tailor-made instruction which is adapted to their interests and capabilities, and which allows them to learn at their own pace, whenever and wherever they want to. As far as the effectiveness of learning is concerned, one-to-one tutoring seems to be better than classroom teaching. In Bloom (1984), it is reported that in various experiments the average performance of learners under tutoring turned out to be about two standard deviations above the average of the control class. Unfortunately, however, it is often too costly to provide every learner with a good human tutor. Moreover, there is currently a tendency for classrooms to become larger and for the variability within classrooms to grow.[1]

These considerations have led to attempts to construct interactive instruction systems (IIS's) that can tutor as well as humans. In a classroom environment, IIS's could be used to give poor learners extra, and specialized attention. Another possibility is to use IIS's for the average and good learners, providing the teacher with more time for the truly problematic cases. Learners could also use such a system at home.

Another reason to build an IIS is the fact that for some skills it would be too dangerous, difficult, or expensive to teach them in real life. This applies, for instance, to the skills of pilots, astronauts, and controllers of chemical and nuclear plants. The IIS can then be used to simulate the training domain.

Over the last decades, many researchers have tried to construct interactive instruction systems (see, e.g., Wenger (1987) and the NATO series on Educational Technology). In general, their goal has been more humble than to make a real tutor (even though they sometimes called their system a tutoring system nevertheless); instead they typically intended their system to assist in classroom instruction. However, despite all the money and efforts expended, education inside and outside schools has hardly been affected (Bouwhuis, van Hoe, & Bouma, 1996).

In this chapter, we will analyse this problem, by identifying weaknesses in current approaches to the development of IIS's. First, the requirements to be met by an IIS are described. Next, some current approaches to the design of such systems are discussed and evaluated in terms of these requirements. Finally, some conclusions and recommendations are offered.

1. In the Netherlands, this is a natural consequence of government programs like "Weer samen naar school" (Back to school together) which are intended to stimulate integration.

1.2 Requirements of an interactive instruction system

What requirements should an interactive instruction system meet? As argued in Bouwhuis & Bunt (1993), an interactive instruction system is a special kind of dialogue system. In turn, a dialogue system is a special kind of software system. The most important requirements of any software system are therefore discussed first, followed by the additional requirements of a dialogue system, and finally the requirements specific for an interactive instruction system. As it is hard to determine when a software system becomes a dialogue system or a dialogue system becomes an instruction system, there is possible disagreement about whether a certain requirement should be mentioned in one category or the other. However, this is not very important, because we are interested in all the relevant requirements of an interactive instruction system.

1.2.1 What requirements should any software system meet?

In Somerville (1985), the following requirements are mentioned for a software system:

1. *Robustness.* Unexpected behaviour of the user should not lead to a total collapse of the system. This is even more important in an IIS, because a student will not (and should not) have as much knowledge about the learning system as, for instance, a professional user of a software system. Hang-ups and bugs will therefore have a fairly negative effect on the student's attitude.

2. *Maintainability and Generativity.* In general, a software platform should be sufficiently flexible and maintainable to tolerate modifications and extensions at no great expense as regards programming. Preferably, the system or parts of the system are easily reusable in other systems.This is one of the reasons for the emergence of Object Oriented Programming (Meyer, 1988). The architecture should be as simple as possible. In an IIS, maintainability implies, for instance, that the course can easily be extended and modified. Generativity implies that the modules of an existing IIS can be reused when designing a system for another learning domain, or another group of students. This reduces the cost and time needed for the design of a new system.

3. *Economy.* To ensure speed of response, it is desirable for a system to operate in real time. However, the application should use as few resources (like memory) as possible. There should be a good trade-off between performance and use of resources.

In an instruction situation, these requirements are often even more imperative than usual.

1.2.2 What are the additional requirements of a dialogue system?

A dialogue system is an interactive software system, where interactive means that both the system and the user perform actions that are meaningful in combination, i.e., they can be interpreted as serving one or more common goals. The main difference as compared with an ordinary software system is, therefore, the interleaving of the actions of the system and the user which is actually part of the functionality, with each action of the one being in some way a response to the action of the other. This leads to the following additional requirements:

1. *Transparency of dialogue.* According to Bouwhuis & Bunt (1993), users should know at any point in time where they are in the dialogue, how they got there, where they came from, how they can go back, where they can go from there, how to correct an error and what result will ensue from an action.In addition the number of choice points should be limited.

2. *Speed of response.* A delay in response may reduce the perceived interactivity considerably. However, a delay that is intolerable in one condition may not be noticeable in another (Taylor, 1988a). As mentioned by Taylor, delays of a few tens or hundreds of msec in echoing characters may badly disrupt the work of a typist, but similar delays in response to a completed command are accepted as normal. Students are not as motivated to use a computer as users who have no alternatives. A student will not sit still while a system muddles for minutes trying to figure out what the student is doing (Anderson, Boyle, Corbett, & Lewis, 1990). A way to avoid undesirable waiting time is to mimic human conversation. Humans try to avoid silent periods as far as possible and frequently intersperse these periods with vapid remarks like "uh"(Beun, 1989).

3. *Efficiency of communication.* There should be a good trade-off between the speed and accuracy of communication. A possible way to increase the efficiency is by using layered protocols (Taylor, 1988a, 1988b). In the process of human-machine communication many layers of abstraction can be distinguished (similar to Tanenbaum (1988) for machine-machine communication). For instance, at a high level of abstraction a sentence is communicated, at a lower level of abstraction the words the sentence contains are communicated, at a very low level pixels of the screen are put on in order to form letters. To make communication successful, the levels of abstraction in the communication of both partners should be the same. An important property of a layered protocol is that feedback is provided at different levels of abstraction. This enables the speaker to verify at an early stage of communication whether his or her intentions are being accurately perceived. Of course, the contents of the feedback should depend on (the history of) the

user's actions. For a more detailed explanation of layered protocols see Taylor (1988a,1988b) and Engel & Haakma (1993).

Another possibility to increase efficiency is the use of layered expectations (Engel & Haakma, 1993). In a dialogue, a recipient can generate expectations about what will follow. Such expectations reduce the number of possible decodings of the last part of the message. In human conversations, it often occurs that the listener is so quick that the reaction to a message takes place even before the speaker has finished the sentence.

4. *Cognitive validity.* In many systems the designers are trying to model users in terms of their beliefs and knowledge (e.g., Kobsa & Wahlster, 1989). It is assumed that such models can be used to anticipate or explain the reactions of the user and thereby make communication more efficient. However, a system that uses an invalid cognitive model has a high probability of taking incorrect decisions, which are incomprehensible for the user. This affects both the robustness and the transparency of the system. Therefore, it is important that the models used are cognitively valid. Moreover, in order for a dialogue system to be effective, the assumptions on which it is based regarding the way people interact in general, or, in the case of an IIS, the way students learn, should be cognitively valid as well.

5. *Learnability.* When highly motivated, a user may learn to work with a terrible interface, just as children learn in no time to use the interface of a game. However, users are sometimes hesitant to learn something new, especially something totally unfamiliar to them. Because dialogue systems, and in particular IIS's, are not only intended for computer experts, it is important for ordinary people to overcome their reluctance to the new technology. Even when a system is intended for children, the teachers and parents still decide whether or not to adopt it. One way to make a system more acceptable is to make its interface so transparent that it is easy to learn. As argued by Nielsen (1993), learnability is a very important prerequisite for usability.

6. *Evaluation research: user-centred approach.* At all stages, the design process should be based on continuous user testing with the prototype. This is what Norman & Draper (1986) call the user-centred design. Nevertheless, it remains necessary to evaluate the system in actual and prolonged use. This is especially true for IIS's, because progress is usually so slow that sufficiently reliable data concerning educational success only become available after a considerable time. In particular, one should be aware of the Hawthorne effect (Mayo, 1933): the phenomenon that non-experimental variables rather than the intended variables determine the experimental results. For instance, in the evaluation of an IIS, motivation and attention are not the experimental variables, but they can influence the effectiveness of learning enormously. According to Bouwhuis (1992), the mere fact that the instruction program differs so

much from conventional forms of instruction causes motivation and attention to increase in the initial stage.

Snow & Swanson (1992) describe two ways to evaluate an IIS. It can be judged either by how well it approximates the effects of expert tutoring and small-group teaching directly, or by how well it complements human classroom teaching to attain such effects. Moreover, as mentioned in Clancey & Soloway (1990), systems need to be evaluated by applying them in different domains to determine the generality of the methods.

1.2.3 What requirements are specific for an interactive instruction system?

What distinguishes an IIS from a more general dialogue system? The goal of an IIS is to promote learning, preferably just as well as a human tutor does. Learning can be defined as a change of the learner's behaviour in reaction to a certain situation (Gagné, 1985). In this case, a situation should be viewed as a combination of events external to the learner (as perceived by the learner) and the internal state of the learner (e.g., memory, previously learned capabilities).

Teaching can then be defined as generating events external to the learner in order to promote learning. However, as, according to the definition of learning, the effect of events depends on the internal state of the learner, it is logical that these events should be adapted to the characteristics of the individual learner, such as motivation, knowledge, and skills. Moreover, the kind of events needed may also depend on the kind of learning involved.

Different kinds of learning can be distinguished, depending on the kind of capabilities learned. Gagné (1985) distinguishes between procedural knowledge ("knowing how", for instance, knowing how to multiply two numbers), declarative knowledge ("knowing that", for instance, knowing that the capital of France is Paris), cognitive strategies, motor skills, and attitudes. IIS's are mostly intended for the teaching of procedural and declarative knowledge. Within these categories, it possible to make new distinctions. For instance, Gagné distinguishes within procedural knowledge between rules, concepts, discriminations (like the difference between the sounds of /e:/ and /i:/), higher-order rules (as in problem solving), and procedures. Associations, as mental links between two events, can be seen as the building blocks of these higher order capabilities.We will not discuss all these kinds of learning in detail here (see Gagné, 1985), but association learning and concept learning will be discussed more deeply in Chapter 3 and 5, respectively. For the moment, we are more concerned with the properties of learning in general, which result in the following additional requirements for an IIS:

1. *Adaptivity with respect to teaching order.* Different kinds of learning are often interdependent, which means that certain capabilities are prerequisite to the learning of other capabilities (Gagné, 1985). For instance, learning the names of two persons has as a prerequisite the capability of discriminating between the two persons. This kind of interdependence shows that an important task of a teacher is to determine the order in which capabilities are taught.

According to Wood, Wood, & Middleton (1978), a student should continually be confronted with tasks of controlled complexity: confronting the student with exercises that lie beyond the student's current level of competence, but not so far beyond that he or she is unable to master the problem presented. This approach implies that the student never succeeds too easily nor fails too often, which is good for motivation. Moreover, they demonstrated that this teaching strategy is more effective than other strategies.

Apart from the fact that each task must be at the adequate cognitive level for the student –neither too simple nor too difficult– the task sequence as a whole must be coherent: successive tasks should deal with the same or related concept sets (McArthur, Statx, Hotta, Peter, & Burdorf, 1988). In addition, to enhance motivation, the concepts to be taught should be adjusted to the interests and needs of the student.

2. *Adaptivity with respect to repetition.* Repetition is needed for skill training and to prevent forgetting that may be due to the interference between different capabilities. For instance, in stimulus-response learning, interference may be caused by a similarity between stimuli or responses. The issue of repetition will be discussed in more detail in Chapter 3, 4, and 5, for association learning and concept learning, respectively. As learning depends on the individual learner, the amount of repetition and the decision as to what to repeat at what time also have to be adapted to the student.

3. *Adaptivity with respect to feedback.* The preceding discussion about the efficiency of communication mentioned the importance of feedback at different levels of the communication. In an IIS, feedback plays a very important role, because it can be expected that misunderstandings and errors frequently occur. Moreover, according to Gagné (1985), in all the types of learning mentioned above, confirmation (a terminating event that provides satisfaction, also called praise or reinforcement) is crucial and should be provided contingent on the required behaviour. Therefore, adaptivity of feedback is dealt with here in more detail as a requirement of an IIS.

There are many studies concerned with the timing of feedback. Some of those report that feedback should be provided immediately after the student's response (Sullivan, Schutz & Baker, 1971; Kulik & Kulik, 1988). Others argue that feedback should be delayed, i.e., presented some time after the learning event (More, 1969; Sassenrath & Yonge, 1968, 1969). According to Anderson et al. (1990), more experienced students can find immediate feedback annoying. So, the ability of the student should probably influence the timing and nature of feedback.

The correct answer to a question can be provided after one response (single-try feedback), or after several responses (multiple-try feedback). Clariana (1993) compared the effectiveness of these two kinds of feedback. He found that single-try feedback was more effective for low prior-knowledge students, while for high prior-knowledge students multiple-try feedback was more effective.

There are also studies that investigate the effectiveness of different amounts of information in the feedback. A distinction can be made between knowledge of response feedback (like "right", "wrong"), knowledge of correct response feedback (like "the right answer is..."), and more elaborate forms of feedback. A lot of research suggests that the amount of feedback should depend upon the prior knowledge or ability of the student (see, e.g., Tobias, 1973; 1976; McGowen & Clark, 1985; Spaai, 1994). There is also evidence that there is a correspondence between the confidence a student had in his response and the correctness of that response, on the one hand, and the amount of feedback a student wants, on the other hand (Kulhavy & Stock, 1989).

All of this suggests that the amount and timing of feedback should be adapted to the individual student. In the educational literature, feedback is mostly viewed as a way to correct errors during instruction (Anderson, Kulhavy, & Andre, 1972), but feedback is also an important factor for motivation. Adaptation of the degree of enthusiasm and disappointment can be important for keeping a student motivated. If responses are repeatedly correct, feedback may be suspended altogether, using the absence of explicit feedback as a default option for positive feedback (Bouwhuis & Bunt, 1993).

4. *Adaptivity with respect to presentation.* At each moment in the training session, the most appropriate representation of a task has to be presented to the student. In multimedia systems in general, a selection has to be made of the available media, like audio, print, pictures, animations, stills, and video. From ten different media selection models discussed in Reiser & Gagné (1982), seven have student characteristics as a prominent factor. This therefore means that the way a task is presented should also be adapted to the student. Factors frequently considered in media selection models are reading ability and age (e.g., Briggs & Wager, 1981). In Dale (1969), a "Cone of Experience" is presented that lists 12 categories of media and experiences in a hierarchical fashion. The media range from manipulation of real objects (direct, purposeful experiences), through increasingly simulated experiences,

to the use of symbols (reading). Briggs & Wager (1981) suggest as rule of thumb, when using Dale's cone, that one should go as low on the scale as necessary to ensure learning by the student, but as high on the scale as possible for efficient learning.

5. *Adaptivity with respect to explanation.* Verbal instruction should be given to state the performance to be expected when learning is complete, to provide guidance, to explain. The issue of instruction will be discussed in more detail in Chapter 7.

According to Cawsey (1993), explanations should not be totally determined beforehand. She argues that for complex explanations, interactions with the user should take place as the explanation progresses and should influence how that explanation continues. This is, of course, closely related to the notion of layered protocols, as discussed above as a means to increase the efficiency of communication.

As described in Entwistle (1981), Pask (1976) has argued for the existence of distinct styles of learning (i.e., the different strategies students may have for learning something). In experiments, he found that a mismatch between the learning style of a student and the teaching style (i.e., the way an explanation is given) results in bad student performance. Experimental findings from Kyllonen, Lohman, and Snow (1984) on the effects of aptitude and instruction on the performance of subjects in a spatial visualization task also suggest an interaction of aptitude and the most effective way to instruct. So, the explanation of the material should also be adapted to the learning style of a particular student.

6. *Mixed locus of control.* When discussing dialogue systems, we have defined interaction as a situation in which both the system and the user perform actions, where the action of the one is a kind of response to the actions of the other. This does not, however, necessarily imply that both interaction partners play an equal role. One partner may be more in control of the interaction than the other one. For instance, in a school situation, the teacher is generally more in control than the student. Two aspects seem to be important when trying to define what control means.

In the first place, the decision as regards turn taking is crucial: in order to be in control, an interaction partner has to be able to decide for himself when to act. An interaction partner who can only act when the other one explicitly gives the opportunity to act does not have control.

In the second place, the decision of the action is important: in order to be in control, an interaction partner has to be able to decide for himself what action to perform. When the set of actions of the system and user can be interpreted as together serving one or more goals, the partner who determines these goals can be said to have more control.

Of course, it need not necessarily be the case that only one of the two partners is in control. In most natural dialogues this is certainly not the case. There is usually a mixed locus of control: during the interaction both partners can decide upon turn taking, and both can determine, under some constraints, what kind of action to perform. According to Gentner (1992), a mixed locus of control is almost essential for keeping p a student motivated.

In the interaction between two human partners, there is not an explicit decision about the locus of control. Instead, the locus of control follows quite naturally from the interaction of the partners. A mixed locus of control can be created dynamically on the basis of the instruction model of Wood, Wood, and Middleton (1978). In this model, instruction is considered as a layered model of intervention varying between the "general verbal encouragement" level and the "demonstration" level, whereby each layer represents a different level of control from the side of the teacher. The behaviour of the teacher should then be contingent on the performance of the student: the most appropriate intervention is chosen according to the success of the student. This strategy demands that when the student starts to fail the IIS takes over control immediately, to the point where the student finds himself successful, after which the IIS attempts to progressively relinquish control to the student.

Indirect evidence for the effectiveness of this strategy has been found by Swanson (as reported in Snow & Swanson, 1992). Swanson found that total student control (as in discovery learning) was good for the most able students, but was particularly ineffective with low-ability students, who benefited most from contingent tutoring. Total teacher control (as in lectures) produced intermediate results. Experiments in Ellerman (1991) show that children are unable to monitor their own learning completely.

1.2.4 Summary of requirements

The requirements discussed above can be summarized as follows. The first three are general requirements that should be met by any software system, the next six are additional requirements of a dialogue system, and the last two are requirements specific for an IIS.

1. Robustness
2. Maintainability and Generativity
3. Economy

4. Transparency of dialogue
5. Speed of response

6. Efficiency of communication
7. Cognitive validity
8. Learnability of the interface
9. User-centred evaluation research
10. Adaptivity with respect to teaching order, repetition, feedback, presentation, and explanation
11. Mixed locus of control.

As far as the relative importance of the different requirements is concerned, it is my opinion that the more general the requirement, the more important it is. So, the requirements of a software system are more important than the additional requirements of a dialogue system, and in their turn they are more important than the additional requirements of an IIS. For instance, adaptivity is pointless in a system which has frequent breakdowns.

1.3 How well do current approaches meet these requirements?

In this section, current approaches to the design of IIS's are evaluated in terms of the requirements discussed above. Four types of learning systems are distinguished. There is, however, no clear boundary between the different categories; depending on the criterion used, some systems could be classified as belonging to different categories. The evaluation in terms of the requirements does not necessarily hold for all systems belonging to a certain category; the probable consequences of the use of a certain approach are described. For instance, if it is stated that the robustness of a certain category of systems is good because they are simple systems, that does not mean that every system of that kind is robust.

1.3.1 Computer Assisted Instruction

A Computer Assisted Instruction (CAI) system can be compared with a sophisticated learning-book. Lessons prepared by a human expert are encoded in prestored instructional units, often called frames. These frames contain small portions of the curriculum material and are successively displayed on the screen for presentation and for questioning. Their sequence is determined by fixed branching decisions based on a predefined set of possible answers expected from the student. Most CAI systems are drill-and-practice systems, in which a student has to answer questions on a specific topic until a certain criterion has been reached. Only simple information about the student is registered in what could be called a student model, as overall measures of performance and how far the student has progressed in the curriculum. An example of CAI is the drill-and-practice system for elementary mathematics developed at Stanford (Suppes & Morningstar, 1972).

How well do CAI systems meet the requirements mentioned in the previous section?

1. *Robustness.* CAI systems have a simple architecture, use simple rules, and do not make assumptions about the student's evolving knowledge. Therefore, their robustness tends to be very good.

2. *Maintainability and Generativity.* Because they are very simple systems, CAI systems can be easily maintained. Generativity is good as the branching mechanisms and multiple choice exercises, which are commonly used, can be reused for various kinds of domains. A problem of the branching mechanism, however, is that the addition of new material may be not straightforward, because all possible paths through the course material have to be determined beforehand.

3. *Economy.* CAI systems do not use complex computations and, except for the course material, require only a limited amount of storage space. So, their economy tends to be very good.

4. *Transparency of dialogue.* The number of ways to proceed through the course material is often rather limited, making these systems very deterministic and predictable. Transparency can therefore be readily attained.

5. *Speed of response.* CAI systems tend to react very fast, because not many computations are needed, and those which are required are very simple.

6. *Efficiency of communication.* The structure of the communication between a CAI system and its users is typically very simple. Consequently, efficiency of communication is hardly an issue.

7. *Cognitive validity.* Drill-and-practice systems are based on the work of the behaviourist Skinner (see, e.g., Skinner, 1960). For Skinner, the goal of learning was for students to exhibit appropriate behaviours; it is not important what students think as long as they behave correctly. He views teaching as the process of conditioning students to respond correctly to a given stimulus. This theory of learning does not take account of such factors as motivation, intention, and understanding.

8. *Learnability of the interface.* As CAI systems are relatively simple systems, their user interface tends to be simple and easy to learn.

9. *User-centred evaluation research.* User-centred evaluation research is lacking in most CAI systems, though the Reading Board (Ellerman, 1991) has been developed by using it.

10. *Adaptivity.* The Adaptivity of CAI systems is limited to the amount of branching and is therefore very limited.

11. *Mixed locus of control.* The locus of control is fixed: the system completely controls the interaction, which is not very motivating.

1.3.2 Intelligent Tutoring Systems

An Intelligent Tutoring System (ITS) contains explicit representations of the knowledge to be taught (domain expert) and of the way this knowledge has to be transferred to the students (tutoring expert). Its student model is much more fine-grained than that of a CAI system: it is intended to represent the knowledge the student has acquired at any phase of the learning process. The student is modelled in terms of the domain expert: the system records what knowledge of the expert the student has already acquired.

The student model should be executable, in that it simulates the cognitive processes by which the student solves the task. To enable the replication of errors made by the student, the incorrect or so-called 'buggy'-rules that the student uses (from a predefined set) are also stored. Exhaustive testing is used to diagnose which combination of buggy-rules reproduces the incorrect answers of the student. Examples of ITS's are Anderson's LISP, geometry, and algebra tutors (Anderson, Boyle, Corbett, and Lewis, 1990). An overview of ITS's can be found in Wenger (1987).

Evaluation in terms of the requirements:

1. *Robustness.* As mentioned in Bouwhuis et al. (1996), ITS's are not quite robust. In the first place, ITS's try to interpret the student's errors in terms of a predefined set of bugs, and it has been observed that unexpected errors occur frequently with which the systems cannot cope. In an empirical study of elementary algebra errors, Payne & Squibb (1990) found many infrequent errors and few frequent ones, and a high degree of inconsistency between students: different errors had different explanatory power in different student groups. This illustrates that it is hard to make a sufficiently complete library of buggy rules.

Secondly, as instruction is based on the assumed state of the student's knowledge and not on the student's behaviour itself, problems can arise because the student's performance is misinterpreted within the expert module. There are various reasons why it is very difficult to diagnose student errors. First, students make errors in an inconsistent way (Payne & Squibb, 1990; Sleeman, Kelly, Martinak, Ward, & Moore, 1989). Second, there are errors which are not readily described as buggy-rules (Payne & Squibb, 1990). Third, a single error can sometimes be explained by alternative (combinations of) buggy-rules (Payne & Squibb, 1990). Fourth, it is often difficult to identify whether an error is the result of applying incorrect knowledge –the subject has made a mistake– or whether the student has intended to perform the appropriate action, but failed to do so - the subject "slipped" (Norman, 1981).

2. *Maintainability and Generativity.* ITS tend to be complex systems, which makes them hard to maintain. As far as generativity is concerned: it is not easy to change the domain of an ITS, as the tedious work of determining a library of buggy-rules has to start all over again for each new domain. On the other hand, the tutoring expert could be reusable.

3. *Economy.* ITS's need a lot of complex computations (for the diagnosis of errors) and a large amount of storage space (for the library of buggy rules and the student model), making economy rather poor.

4. *Transparency of dialogue.* The actions of the ITS are determined on the basis of the assumed state of the student's knowledge. So, when the student's performance is misinterpreted within the expert module, the decisions of the ITS are no longer transparent to the student.

5. *Speed of response.* Exhaustive diagnosis cannot be used in complex domains, because it would make response delay unacceptable (Pijls, Daelemans, & Kempen, 1987). It is infeasible to keep the search time within acceptable limits in a more complex domain, except by making considerable sacrifices with respect to the robustness of the system.

6. *Efficiency of communication.* The use of the student model is supposed to make the communication more efficient. For example, the student model can allow the system to make use of knowledge previously acquired by the student when explaining new topics. But, of course, the efficiency depends on the correctness of the model.

7. *Cognitive validity.* ITS designers claim that the student model allows feedback that can correct any misconception, and helps to determine what to teach next and how to teach it (VanLehn, 1988). Error-specific remediation is assumed to be superior to reteaching. Putnam (1987) found, however, that detailed diagnosis was not an important goal of teachers as they tutored students. Teachers generally do not adopt the role of a diagnostician, even when in a tutorial situation. Studies reported in Sleeman et al. (1989) show that both model-based remediation and reteaching were effective in remedying errors, but there was no discernible difference between them. There are various possible reasons for this (see also Bouwhuis et al., 1996).

In the first place, student modelling is not based on explicit knowledge of the learning process; the history of the student's performance is rarely taken into account. Hence, the student's learning process is only monitored indirectly.

Secondly, it is not clear that the representation of the expert module corresponds to the cognitive representation of the student. There is indeed empirical evidence that expert knowledge structures do not provide the most useful models for teaching (Roschelle, 1990). In many ITS's, a rule-based representation formalism is used (e.g., Anderson et al., 1990). There is, however, increasing evidence against the idea that production rules form the units of knowledge (e.g., Gluck & Bower, 1988a; 1988b).

8. *Learnability of the interface.* Depends on the application.

9. *User-centred evaluation research.* Lacking; most systems built so far have been laboratory experiments which have not been evaluated in actual practice (Wenger, 1987).

10. *Adaptivity.* Even though the main goal of ITS designers is to make adaptive systems, the adaptivity they provide is limited. In most ITS's (e.g., Anderson et al., 1990), the student model is used only to determine what feedback to give a student when he or she has made a mistake. In a review by Ohlsson (1986) of a wide variety of intelligent tutors, it is noted that few use their expertise to influence the global structure of lessons. McArthur et al. (1988) have made an attempt to use student modelling to adapt task sequencing to the individual student. However, most ITS's are certainly not adaptive with respect to, for instance, task sequencing, presentation, and locus of control.

11. *Mixed locus of control.* The locus of control is fixed: the system completely controls the interaction, which is supposedly not very motivating for the students and, consequently, reduces the effectiveness of learning.

The problems caused by the use of a fixed library of buggy-rules are addressed by the makers of a system called PIXIE (Sleeman et al., 1989; Wenger, 1987). PIXIE does not exclusively use a predefined set of buggy-rules, or mal-rules as they are called there, but generates new mal-rules if necessary. When a step cannot be accounted for by the current model of rules and mal-rules, the student's step is posited as a new mal-rule in a form of data-driven diagnosis. Such an approach makes the system more robust and saves the time and space otherwise needed to make a large library of mal-rules. To become truly useful, however, mal-rules of a more general kind have to be generated, which can account for an entire class of errors. But in principle, a given mistake can be explained by any number of mal-rules, some of which will be completely implausible. Preventing the generation of implausible mal-rules is basically an unsolved problem.

1.3.3 Microworlds

Microworlds are exploratory simulation environments in which a student can learn by acting on the virtual environment. Current topics of microworlds include Euclidian mathematics (geometry Turtles, see Papert, 1980), Newtonian physics (physics Turtles, see Papert, 1980), chemical titration, and kinematics (DiSessa, 1995). Students are supposed to acquire knowledge by experimenting with the simulation and drawing conclusions from the results. They can set parameters initially and see what happens (e.g., in a chemical titration microworld) or even adjust parameters continuously (e.g., in a flight simulator). This constructive learning should lead to a more qualitative understanding of the domain and should correct persistent misunderstandings. Moreover, it should help students to learn how to develop and debug their own theories, which is, as suggested by Papert (1980), more important than teaching them theories we consider correct.

Evaluation in terms of the requirements:

1. *Robustness.* Microworlds use no assumptions about the student's knowledge. So, the robustness of these systems tends to be good, though this depends of course on the complexity of the simulations.

2. *Maintainability and generativity.* Microworlds are very domain-dependent systems. It may be easy to modify a simulation a little, but it is often hard to reuse components for other domains. Moreover, for some domains (like physics, complex machines) it is a lot easier to construct microworlds than for other domains (like language).

3. *Economy.* The economy depends on the complexity of the simulations, but, as there is no overhead of student modelling, it tends to be good.

4. *Transparency of dialogue.* The interaction between a microworld and its user tends to have an extremely simple structure ('user acts, system reacts'). Consequently, transparency is automatically guaranteed.

Speed of response. The speed of response depends very much on the complexity of the simulation. When the simulation does not require very complex computations, it may be good. However in more complex domains the speed of response may become unacceptable.

6. *Efficiency of communication.* The structure of the communication between a microworld and its users is always very simple. Consequently, efficiency of communication is hardly an issue.

7. *Cognitive validity.* It is assumed that learning by discovery will lead to a durable and deep understanding by the students (e.g., Wittrock, 1966). As stated in Bouwhuis et al. (1996), there is, however, no reliable and systematic evidence for this; the incidental nature of the discovery seems to lead to frag-

mentary knowledge at most, and not to a generalizable body of knowledge. Moreover, as described in VanJoolingen (1993), there is extensive evidence that students are often unable to approach a simulation in a scientific way: they do not formulate an adequate hypothesis, are unable to design experiments suitable for testing the hypothesis, or to interpret the results correctly. They often use trial and error rather than insight. Furthermore, Klahr and Dunbar (1988) show that students seek evidence that will confirm their hypothesis rather than disconfirm it, and that they tend to maintain the hypothesis even when confronted with evidence that it is incorrect. Finally, as already mentioned above (in the discussion of the requirement of an adaptive locus of control in Section 1.2.3), there is evidence that learning by discovery is ineffective with low-ability students.

8. *Learnability of the interface.* The learnability depends on how complex the simulation gets. The interface of a flight simulator can be difficult to learn, while the geometry Turtles can be handled even by small children (Papert, 1980).

9. *User-centred evaluation research.* Except perhaps for the military applications, evaluations are mostly lacking.

10. *Adaptivity.* There is no adaptivity, except that the parameters set by the student determine the course of the simulation.

11. *Mixed locus of control.* Though the student is mostly in control, many microworld systems have a certain degree of autonomy. For instance, in a flight simulator the student can initiate a range of actions, but also has to respond to actions initiated by the system (Gentner, 1992).

1.3.4 Interactive Learning Environments

Interactive learning environments (ILE's) provide a number of tools to support the learning process. They are independent of the domain to be learned and aim at collaborative and constructive learning. Often, they are network systems like, for example, MediaText (see Soloway, 1991). With MediaText, students can write a document (e.g., about physics) and incorporate video clips, sound clips, graphics and animations into it. They can share their documents with each other and get remarks on them by using e-mail. Other examples of ILE's are CSILE (Bereiter & Scardamalia, 1992) and LOGO (Papert, 1980).

Evaluation in terms of the requirements:

1. *Robustness.* ILE's tend to be quite robust, as no assumptions about the student's knowledge are made and no domain knowledge is used.

2. *Maintainability and Generativity.* ILE's are domain-independent, so they can be used indiscriminately for all kinds of domains.

3. *Economy.* There are no student and tutoring models and no complex calculations are needed, so that the economy tends to be very good.

4. *Transparency of dialogue.* The dialogue between an ILE and its user is restricted to domain-independent issues, so that transparency of the dialogue is easy to attain.

5. *Speed of response.* ILE's are mere tools, which require no complex computations. However, the speed of response also depends on the technical limitations implied by working on a network.

6. *Efficiency of communication.* The structure of the communication between an ILE and its user tends to be very simple. Efficiency of communication is therefore easy to attain.

7. *Cognitive validity.* It is assumed that collaboration between students increases learning efficiency. There is renewed interest in collaborative learning with studies that show positive effects (Bielaczyc, Pirolli, & Brown, 1994). However, Elshout (1992) concludes from a review of 22 experiments that address the question of whether collaborative learning is more effective than individual learning that there is no evidence for that. As mentioned in Bouwhuis et al. (1996), there are indeed studies that show notable improvements for poor learners when using an ILE, but in those studies, the ILE tends to be used with an active supervising teacher. So, it could well be that an ILE keeps the more able students busy, providing the teacher with an opportunity to give the poor learners extra support.

8. *Learnability of the interface.* Tends to be good, especially due to their similarity to tools people are already familiar with. For example, teachers and parents can use MediaText almost immediately, because of its similarity to a word processor (Soloway, 1991).

9. *User-centred evaluation research.* Evaluations are mostly performed afterwards and are not an integral part of the design process.

10. *Adaptivity.* There is no adaptivity, except that the ILE can be used for whatever the student wants to do.

11. *Mixed locus of control.* The locus of control is fixed: the student is totally in control. However, though it may be very suited for high-ability students, complete control of the interaction by the students does not lead to effective learning for all students, as has already been argued above (when discussing the requirement of an adaptive locus of control).

1.4 Conclusions and recommendations

In Table 1 a summary is presented of the evaluation of various approaches to the design of IIS's as discussed in the previous section. As can be seen, each approach has a number of positive features:

CAI systems score very well with respect to basic requirements such as robustness, economy, maintainability, generativity, transparency, and speed of response. The main reason for this is that they have a very simple architecture and do not involve complex computations. However, they lack adaptivity and do not lead to long-term motivation.

The design of ITS's is aimed at obtaining adaptivity. ITS designers try to do this by modelling the student in terms of the expert. Though this may result in a certain degree of adaptivity, it also tends to lead to poor robustness, economy, generativity, transparency, and speed of response. So, all basic requirements are violated. Moreover, the amount of adaptivity achieved is often limited: there is some adaptation of the content of feedback, but hardly any with respect to, for instance, the locus of control, presentation, and task sequencing.

Microworlds have a simple architecture, but nevertheless provide an appealing environment for the student, as they offer a considerable amount of variation.

A positive feature of ILE's is the very good learnability of the interface, which results from the use of a familiar metaphor, like a word processor. However, the goal with ILE's is not very ambitious, as they are only intended as tools.

Table 1: *Summary of the evaluation of approaches to IIS design in terms of the requirements (+=tends to be good, ++=tends to be very good, -=tends to be bad, --=tends to be very bad, ?=depends).*

Requirement	**CAI**	**ITS**	**Microworld**	**ILE**
Robustness	++	--	+	++
Maintainability & generativity	+	-	-	+
Economy	++	--	+	++
Transparency of dialogue	+	-	+	+
Speed of response	++	--	?	?
Efficiency of communication	+	?	+	+
Cognitive validity	-	-	-	-
Learnability of interface	+	?	?	++
User-centred evaluation research	-	-	-	-
Adaptivity	--	+	--	--
Mixed locus of control	--	--	+	--

The main shortcomings of the earlier approaches seem to be a lack of:

• *Adaptivity.* Only ITS's offer some adaptivity, but at the expense of more general requirements

• *Generativity.* Only CAI systems and ILE's are reusable in different domains. A common problem for ITS's and microworlds is that the design process is very much domain-specific. As a consequence, for each new domain, the design of such a system must start all over again.

• *Mixed locus of control.* Interactive instruction systems of the type of CAI and ITS allow the user almost no control. Supposedly such systems are not very motivative and are therefore not quite effective. On the other hand, in ILE's the control is almost entirely in the hands of the student. As argued above, complete student control does not result in optimal learning for every student either. Only microworlds sometimes provide sometimes a limited form of a mixed locus of control. However, monitoring of the student's learning process is mostly lacking.

• *User-centred design.* For all the approaches reviewed, there are almost never evaluations at an early stage in the design which can be used to improve the design. Only very rarely are complete systems tested in a real-life, non-experimental setting.

Our approach, which is the topic of the remainder of this dissertation, will be specifically aimed at addressing these requirements.

Our realization of adaptivity, which will be discussed in Chapter 2, is based on a situated theory of cognition. No complex student modelling is used, but the student's learning process is monitored in a more direct way, using the learner's (history of) behaviours instead of a model inferred from these. We will attempt to demonstrate that this method has several advantages as compared with the conventional approach taken in ITS's. Moreover, we will show how adaptivity can be attained not only for feedback, but also for the other aspects of teaching discussed above, like instruction and task sequencing.

Generativity is addressed by using an architecture which contains several generic modules the behaviour of which is derived from general instruction principles. To make it probable that the basic requirements are met, as in CAI systems, our architecture of an IIS will be simple, in addition to which the IIS will not involve complex computations.

A mixed locus of control will be obtained as a natural consequence of the interaction between an adaptive IIS and the student.

In the next chapter, the role of experiments in the design process will be briefly discussed and this will be illustrated in subsequent chapters.

References

Anderson, J.R., Boyle, C.F., Corbett, A.T., & Lewis, M.W. (1990). Cognitive modelling and intelligent tutoring. *Artificial Intelligence, 12,* 7-49.

Anderson, R.C., Kulhavy, R.W., & Andre, T. (1972). Conditions under which feedback facilitates learning from programmed lessons. *Journal of Educational Psychology, 63,* 186-188.

Bereiter, C. & Scardamalia, M. (1992). Two models of classroom learning using a communal database. In S. Dijkstra, H.P.M. Krammer, and J.J.G. van Merriënboer (Eds.), *Instructional models in computer-based learning environments* (Volume F104 of NATO ASI Series). Berlin: Springer.

Beun, R.J. (1989). *The recognition of declarative questions.* Ph.D. thesis, University of Tilburg, The Netherlands.

Bielaczyc, K., Pirolli, P.L., Brown, A.L. (1994). *Collaborative explanations and metacognition: identifying successful learning activities in the acquisition of cognitive skills.* Report no. CSM-8.

Bloom, B.S. (1984). The 2 sigma problem: the search for methods of group instruction as effective as one-to-one-tutoring. *Educational Researcher, 13,* 4-16.

Bouwhuis, D.G. (1992). Cognitive modelling and learning. In F.L. Engel, D.G. Bouwhuis, T. Bösser, and G. d'Ydewalle (Eds.) *Cognitive modelling and interactive environments in language learning* (Volume F87 of NATO ASI Series). Berlin: Springer.

Bouwhuis, D.G., & Bunt, H. (1993). Interactive instructional systems as dialogue systems. In L. Verhoeven (Ed.) *Training for literacy.* Dordrecht: Foris.

Bouwhuis, D.G., VanHoe, R., & Bouma, H. (1996). Advancing education by proper technology. In T. Liao (Ed.), *Advanced educational technology: Research issues and future potential* (Volume F145 of NATO ASI Series). Berlin: Springer.

Briggs, L.J., & Wager, W.W. (1981). *The handbook of procedures for the design of instruction* (2nd ed.). Englewood Cliffs, N.J.: Educational technology publications.

Cawsey, A. (1993). Planning interactive explanations. *International Journal of Man-Machine Studies, 38,* 169-199.

Clancey, W.J., & Soloway, E. (1990). Artificial intelligence and learning environments:preface. *Artificial Intelligence,* 2, 1-6.

Clariana, R.B. (1993). A review of multiple-try feedback in traditional and computer-based instruction. *Journal of Computer-Based Instruction, 20,* 67-74.

Dale, E. (1969). *Audiovisual methods in teaching* (3rd ed.). New York: Holt, Rinehart, & Winston.

DiSessa, A.A. (1995). Designing Newton's laws: Patterns of social and representational feedback in a learning task. In R.-J. Beun, M. Baker, and M. Reiner (Eds.), *Dialogue and instruction: Modelling interaction in intelligent tutoring systems* (Volume F142 of NATO ASI Series). Berlin: Springer.

Ellerman, H. (1991). *MIR, a monitor for initial reading.* Ph.D. thesis, Eindhoven University of Technology, The Netherlands.

Elshout, J.J. (1992). Formal education versus everyday learning. In E. de Corte, M.C. Linn, H.Mandl, and L. Verschaffel (Eds.), *Computer-based learning environments and problem solving* (Volume F84 of NATO ASI Series). Berlin: Springer.

Engel, F.L., & Haakma, R. (1993). Expectations and feedback in user-system communication. *International Journal of Man-Machine Studies, 39,* 427-452.

Entwistle, N. (1981). *Styles of learning and teaching: an integrated outline of educational psychology for students, teachers and lecturers.* New York: Wiley.

Gagné, R.M. (1985). *The conditions of learning and theory of instruction* (4rd ed.). London: Holt, Rinehart & Winston.

Gentner, D. (1992). Interfaces for learning: Motivation and the locus of control. In F.L. Engel, D.G: Bouwhuis, T. Bösser, and G. d'Ydewalle (Eds.), *Cognitive modelling and interactive environments in language learning* (Volume F87 of NATO ASI Series). Berlin: Springer.

Gluck, M., & Bower, G. (1988a). Evaluating an adaptive network model of human learning. *Journal of Memory and Language, 27,* 166-195.

Gluck, M., & Bower, G. (1988b). From conditioning to category learning: an adaptive network model. *Journal of Experimental Psychology: General, 117,* 227-247.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12,* 1-48.

Kobsa, A., & Wahlster, W. (Eds.) (1989). *User models in dialogue systems.* Berlin: Springer.

Kulhavy, R.W., & Stock, W.A. (1989). Feedback in written instruction: the place of response certitude. *Educational Psychology Review, 1,* 279-308.

Kulik, J.A., & Kulik, C.C. (1988). Timing of feedback and verbal learning. *Review of Educational Research, 58*, 79-97.

Kyllonen, P.C., Lohman, D.F., & Snow, R.E. (1984). Effects of aptitudes, strategy training, and task facets on spatial task.performance. *Journal of Educational Psychology, 76*, 130-145.

Mayo, E. (1933). *The human problem of an industrial civilization.* New York: The Macmillan company.

McArthur, D., Statz, C., Hotta, J., Peter, O., & Burdorf, C. (1988). Skill oriented task sequencing in an intelligent tutor for basic algebra. *Instructional Science, 17*, 281-307.

McGowen, J. & Clark, R.E. (1985). Instructional software features that support learning for students with widely different ability levels. *Performance and Instruction, 24*, 2-17.

Meyer, B. (1988). *Object-oriented software construction.* London: Prentice-Hall.

More, A.J. (1969). Delay of feedback and the acquisition and retention of verbal materials in the classroom. *Journal of Educational Psychology, 60*, 339-342.

Nielsen, J. (1993). *Usability engineering.* San Diego, CA: Academic Press.

Norman, D.A.(1981). Categorization of action slips. *Psychological Review, 88*, 1-15.

Norman, D.A., & Draper, S.W. (Eds.) (1986). *User centered system design.* Hillsdale , NJ: Lawrence Erlbaum.

Ohlsson, S. (1986). Some principles of intelligent tutoring. *Instructional Science, 14*, 293-326.

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas.* New York: Basic Books.

Pask, G. (1976). Styles and strategies of learning. *British Journal of Educational Psychology, 46*, 128-148.

Payne, S., & Squibb, H. (1990). Algebra mal-rules and cognitive accounts of error. *Cognitive Science, 14*, 445-481.

Pijls, F., Daelemans, W., & Kempen, G. (1987). Artificial intelligence tools for grammar and spelling instruction. *Instructional Science, 16*, 319-336.

Putnam, R.T. (1987). Structuring and adjusting content for students: a study of live and simulated tutoring of addition. *American Educational Research Journal, 24*, 13-48.

Reiser, R.A., & Gagné, R.M. (1982). Characteristics of media selection models. *Review of Educational Research, 52*, 499-512.

Roschelle, J. (1990). *Designing for conversations*. Paper presented at the annual meeting of the American educational research association, Boston.

Sassenrath, J.M., & Yonge, G.D. (1968). Delayed information feedback, feedback cues, retention set, and delayed retention. *Journal of Educational Psychology, 59*, 69-73.

Sassenrath, J.M., & Yonge, G.D. (1969). Effects of delayed information feedback and feedback cues in learning and retention. *Journal of Educational Psychology, 60*, 174-177.

Skinner, B.F. (1960). Teaching machines. In A.A. Lumsdain & R. Glaser (Eds.) *Teaching machines and programmed learning*. Washington: National Education Association.

Sleeman, D., Kelly, A.E., Martinak, R., Ward, R.D., & Moore, J.L. (1989). Studies of diagnosis and remediation with high school algebra students. *Cognitive Science, 13*, 551-568.

Snow, R.E., & Swanson, J. (1992). Instructional psychology. *Annual Review of Psychology, 43*, 583-626.

Soloway, E. (1991). How the Nintendo generation learns. *Communications of the ACM, 34*, 23-26, 95.

Somerville, I. (1985). *Software Engineering*. London: Addison-Wesley.

Spaai, G.W.G. (1994). *The use of speech feedback in an interactive reading programme for beginners*. Ph.D. thesis, Eindhoven University of Technology, The Netherlands.

Sullivan, H.J., Schutz, R.E., & Baker, R.L. (1971). Effects of systematic variations in reinforcement contingencies on learner performance. *American Educational Research Journal, 8*, 135-142.

Suppes, P., & Morningstar, M. (1972). *Computer-assisted instruction at Stanford, 1966-68*. New York: Academic Press.

Tanenbaum, A.S. (1988). *Computer networks*. Englewood Cliffs: Prentice-Hall.

Taylor, M.M. (1988a). Layered protocols for computer-human dialogue I: principles. *International Journal of Man-Machine Studies, 28*, 175-218.

Taylor, M.M. (1988b). Layered protocols for computer-human dialogue II: some practical issues. *International Journal of Man-Machine Studies, 28*, 219-257.

Tobias, S. (1973). Review of the response mode issue. *Review of Educational Research, 43*, 193-204.

Tobias, S. (1976). Achievement treatment interactions. *Review of Educational Research, 46*, 61-74.

VanJoolingen, W.R. (1993). *Understanding and facilitating discovery. learning in computer-based simulation environments.* Ph.D. thesis, Eindhoven University of Technology, The Netherlands.

VanLehn, K. (1988). Student modelling. In M.C. Polson & J.J. Richardson (Eds.) *Foundations of intelligent tutoring systems.* Hillsdale, NJ: Erlbaum.

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems.* Los Altos, CA: Morgan Kaufmann

Wittrock, M.C. (1966). The learning by discovery hypothesis. In L.S. Shulman & E.R. Keislar (Eds.) *Learning by discovery, a critical appraisal.* Chicago: Rand McNally.

Wood, D., Wood, H., & Middleton, D. (1978). An experimental evaluation of four face to face teaching strategies. *International Journal of Behavioral Development, 1,* 131-147.

Chapter 2

Design and architecture of an agent-based interactive instruction system

Abstract

The degree of interactivity and adaptivity determines the effectivity of an interactive instruction system to a great extent. The problem addressed in this chapter is how a highly interactive and adaptive instruction system can be realized in such a way that for each new instruction domain only limited effort is needed to achieve the same amount of interactivity and adaptivity. An architecture and design process based on situated agents is specified and applied to the domain of interactive instruction. The interactive instruction system consists of a 'society' of agents, in which each agent has a particular competence, i.e., navigation, instruction, feedback, practice, or presentation. The agents operate in parallel and their perception of the external environment (e.g., user actions) and their own actions are closely interlinked. The interaction dynamics between the agents and between the agents and their environment leads to the emergent adaptive functionality of the system. A model of agent communication and a model of agent learning is proposed and also applied to the particular case of interactive instruction systems. The importance of empirical evaluations in the design process is emphasized.

2.1 General architecture of an IIS

An interactive instruction system that has been developed to meet the criteria of adaptivity, generativity, and mixed locus of control is the APPEAL (A Pleasant Personal Environment for Adaptive Learning) system. Its architecture has been based on the following observations.

On a macro level, the problems associated with an interactive instruction system are similar to the general problems associated with the navigation of a user through a large database. On the one hand, there is the database with course material, on the other hand, the student. When the student navigates independently, this may cause a high cognitive load and this complete control may not be good for all students. Therefore, the student could benefit from assistance in navigating, especially if this assistance is tailored to his or her needs. This does not only mean that there is an "assistant" which can answer questions and obeys the student's orders (like 'give me a more difficult exercise'). In accordance with the principle of a *mixed locus of control*, this assistant should also be able to take over control whenever necessary, reducing the student's cognitive load while guarding the effectiveness of the student's learning. Such an assistant can be called a "teacher" or "coach".

In order to optimize the *generativity*, an attempt is made to assure that this teacher is domain-independent as far as possible. The teacher is divided into a domain-independent and a domain-dependent component. The domain-independent part is called the Teaching Expert (TE), and is responsible for the adaptive dialogue with the student. It uses general principles concerning dialogues and human learning (see, e.g., Beun, Baker, & Reiner, 1995). The domain-dependent part is called the Domain Expert (DE), and generates exercises and examples on the basis of information provided by TE. It can also analyse the student's answers. These components collaborate in a master-slave configuration with TE as master and DE as slave. The degree of adaptivity and the locus of control are therefore determined by TE. The role of DE is rather to provide variety within a given context created by TE. For instance, a microworld or simulation would be generated by DE, where TE provides parameter settings for determining the kind of simulation, the level of difficulty, the method of visualization, etc.

In order to be *adaptive*, TE needs meta-information about the course material. For instance, it needs to have a representation of the goals that can be achieved by doing a certain exercise and of the prerequisites to be met before a student may start the exercise, or needs to know the number of the layers of abstraction with which certain information can be presented to the student. This kind of information is stored in the Information Database. While the Information Database contains information about, e.g., the structure of the course material, the Domain Model contains the contents or assets. For the student the Domain Model and the Information Database together constitute the course material.

In order to be adaptive, TE must also store information about the student. For that purpose, there is a Student History. TE can use the Student History to register loggings of the student's performance and his or her interaction with the system. For instance, TE can store data in the Student History concerning which exercises the student has studied, the student's performance on those exercises, how often the student has asked for help, etc. This information can be used in the current or later sessions to adapt to the student.

The architecture of the Appeal system is shown in Figure 1. An extra module called the Interface Manager provides the multimedia interface between system and student. In the current prototypes of the Appeal system, the domains are respectively "Dutch for English speaking persons" and "Square Dancing". In this dissertation, we will only be concerned with the architecture, design, and evaluation of TE.

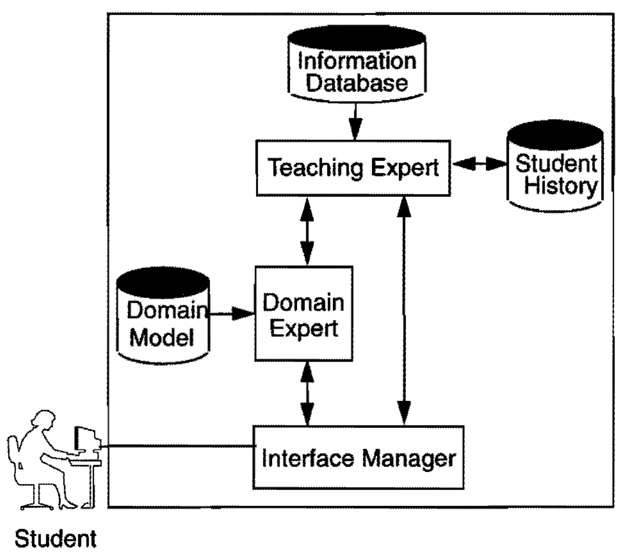


Figure 1: *Architecture of the Appeal system.*

2.2 TE is a Highly Autonomous Agent

In the present section, it will be argued that TE is a special case of a much more general notion, viz. that of a highly autonomous agent. So-called 'agents' are now being studied in various disciplines such as robotics, artificial intelligence, and human-computer interaction. There exists a large diversity of views on agents both within and between these disciplines. This makes it hard to identify what is meant by an agent (e.g., Riecken, 1994). Nevertheless, we propose here a definition of agents and a framework in which agent research can be situated.

Shoham's (1993) conception of an agent is taken as the starting point of our definition. Shoham (1993) defines an agent as an entity that functions continuously and autonomously in an environment in which other processes take place and other agents exist. According to this definition, a first property is that agents react *continuously* to input from the environment.

A second property is that agents are *autonomous*: they have their own resources and are able to make some choices themselves about what to do (i.e., without depending on the user or an external control program). The degree of an agent's autonomy is a dimension along which agent research can be situated. In the minimal sense, autonomy means that the agent can make choices which are imposed on it by the environment, like for instance a mail agent which chooses which e-mail it will present to the user (Maes, 1994). At the other end, autonomy implies that an observer can ascribe multiple goals to the agent and that an observer can believe that the agent is acting in its own interests (Brooks, 1991). For instance, an agent with limited autonomy can be sent to buy a specific VCR, and can choose for itself the salesman that offers the best combination of price and service. An agent with more autonomy can be sent to buy a VCR, and will decide for itself which one is the most appropriate device (as regards price and functionality) for its user. A highly autonomous agent can decide for itself to propose, given the circumstances of one of its users (the user, for instance, has a certain income and children of a certain age) to buy a particular VCR. This agent could have an interest in it, in the sense that it is programmed or kept 'alive' by somebody selling consumer equipment. So, agents vary from the butler-type of agents with limited autonomy to the salesman-type of agents with extensive autonomy. The latter kind of agents are what Tokoro (1994) calls spontaneous agents.

Other properties of agents which are not mentioned by Shoham are competence, intelligence, and believability. An agent has the *competence* to perform a certain task (Maes, 1993). An agent has a certain degree of *intelligence*, i.e., the ability to adapt its behaviour to the environment (McFarland and Bösser, 1993). Note that an agent can be intelligent either because it has an extensive and varied repertoire of fixed behaviour, or because it can learn new behaviours. An agent is *believable* to the extent that it suggests human or animal behaviour (Bates, 1994) and inspires confidence in the user that the agent can perform its task in an acceptable way (Maes, 1994).

To obtain a good interactive instruction system, TE should have all the properties of a highly autonomous agent. In the interaction with an instruction system, the student gives input *continuously* (for instance by pushing an example button, or performing an exercise at a particular rate). A lack of action on the part of a student is also a form of input. TE should react to all this input in a timely fashion. TE should be *autonomous* in the sense that it makes its own decisions with respect to what to present to the student at a given moment. It should be able to cope with a student without the continuous need for a human teacher. To the observer (this may be the student), it should look as if TE really wants the student to learn. This requires a high level of autonomy, and in that respect TE can very well be compared with a salesman. Note that high autonomy of both partners does not exclude a mixed locus of control. Furthermore, TE should have the *competence* to select exercises, give instruction, provide feedback, etc. As pointed out above, TE should be able to adapt its behaviour as regards all kinds of aspects (e.g., feedback, instruction, navigation, presentation) to the capabilities and interests of the student. So, according to our definition it can be called *intelligent*. Students should *believe* that TE supports their learning process.

Until recently, the dominant approach in agent design has been a planning approach in which internal representations and planning were emphasized. However, more and more researchers concerned with the design of *highly autonomous* agents are adherents of a so-called situated approach (Greeno, Chi, Clancey, and Elman, 1993). In the next section, the main ideas underlying the situated approach will be discussed, and their impact on the domain of interactive instruction will be analysed.

2.3 Situated Approach

The situated approach is a reaction against the dominant approach in cognitive science, i.e., the information processing approach (Greeno et al., 1993). The main hypothesis of the situated approach is that the behaviour of humans cannot be detached from the environment they live in. Hence, when modelling human processes proper attention should be given to the relation between a human being and his or her environment. It is important to study not only the mind, as in the information processing approach, or to study almost solely the environment as in behaviourism, but to study the interaction between both of these.

2.3.1 Emergent behaviour

A situated agent's most important resource in determining what to do next is its immediate situation. The organization of activity is seen as emerging from the interaction between an agent and its environment (Chapman, 1991). The complexity of activity does not arise principally from the complexity of the agent or the world, but from the complexity of their interaction. For instance, Simon (1968) notes that the complexity of the behaviour of an ant is more a reflection of the complexity of its environment than of its own internal complexity. To give another example, the complexity of the activity of the artificial agent Pengi, which plays an arcade video game, is the result of the interaction of simple opportunistic strategies with a complex world (Agre & Chapman, 1987).

On another level, the idea is also that it is possible to construct a system which shows high-level, complex behaviour, which is not explicitly defined, but emerges from the interaction of a collective of low-level, explicitly defined behaviours. There is currently a lot of interest in this kind of emergent behaviour (e.g., Forrest, 1991; Steels, 1991).

2.3.2 The world as its own representation

Stucky (1992) argues that agents do not only depend on their environment in a variety of ways, but that they actively use aspects of the environment to calculate and support actions in ways that suggest that they are not representing all relevant aspects of the situation. This view is supported by the findings of Kirsh and Maglio (1994) on cognitive processes in playing Tetris (Tetris is a real-time, interactive video game). They have shown that the use by expert players of many physical rotations that are apparently useless makes the task less cognitively demanding for the player and enables faster playing. So, instead of working on a representation of the world (in the case of Tetris, for instance, using mental rotations), it can be more cost-effective to work on the real world (for instance, using physical rotations). Kirsh & Maglio (1994) call these kinds of actions that are not used to implement a plan or a reaction, but to change the world in order to simplify the problem-solving task, epistemic actions.

Chapman (1989) explains why it is easy for his artificial agent Blockhead to solve the fruitcake problem (the fruitcake problem is to stack a set of labelled blocks so that they spell the word fruitcake) by stressing that Blockhead often uses vision instead of internal representations. Blockhead only represents information that is directly relevant for its task. Representing everything would lead to an explosion of storage space and calculation time. Chapman (1989) argues that in concrete activity, representation mostly just gets in the way. This is also the conclusion of Brooks (1991) after building a series of autonomous mobile robots. He argues for using the world as its own representation, both because it can be more efficient and because it can be a lot more effective in the dynamic, unpredictable environments in which he wants to use his robots.

As the interaction between an agent and its environment is important, and the world should be used as its own representation, this implies that perception becomes very important (Chapman, 1991). In perceptual control theory (Powers,1973) it is even argued that perceptions are the only reality a human being can know, and that the purpose of all actions is to control the state of the perceived environment (to bring it nearer to a goal state, or in the case of epistemic actions to improve the amount of information perceived). Behaviour is regarded as being based on feedback relationships between organism and environment.

2.3.3 Impact on the domain of interactive instruction

On a macro level, the seemingly complex teaching dialogue can emerge from the interaction between a relatively simple IIS and a student. Teaching may seem complex because students are time variant and unpredictable, and not necessarily because the teacher/interactive instruction system is complex in itself. On a micro level, it may be possible to construct TE as the simultaneous operation of a collective of simple instruction principles.

Using the world as its own representation means for TE that trying to represent its environment, in this case the student, may not be a good approach. In fact, trying to represent what a student knows (both correctly and incorrectly) is done in the Intelligent Tutoring Systems approach, and it has been argued in the previous section that this leads to an explosion of storage space and calculation time, and causes problems with robustness, without leading to a sufficient degree of adaptivity. It may be more efficient (requiring less computing power and storage space) and more effective (supporting the student's learning process better) to use direct perception instead. It is, for instance, possible to observe whether the student answers correctly or not. The IIS might also use epistemic actions to get more information than it can perceive at a given moment. For instance, when an error has been made it might ask the student whether he or she made a slip or really does not understand.

Based on these considerations, we have chosen to construct TE as a highly autonomous agent using a situated approach. This has implications for both the architecture and design of TE. The architectural and design principles are not restricted to the domain of IIS's. Therefore, in the next two sections, architecture and design will be discussed in a general way. Toy examples will be used, such as obstacle avoidance in a robot, to illustrate the design process. A separate section will deal with the application of the principles to the case of TE.

2.4 Consequences of the Situated Approach for the Architecture of an Agent

2.4.1 Decomposition of the architecture of an agent

Architectures of agents are often decomposed into functional modules, such as perception, modelling, planning, and plan execution. In the area of interactive instruction systems, the so-called intelligent tutoring systems (Wenger, 1987) also have this kind of architecture, as can be seen in Figure 2.

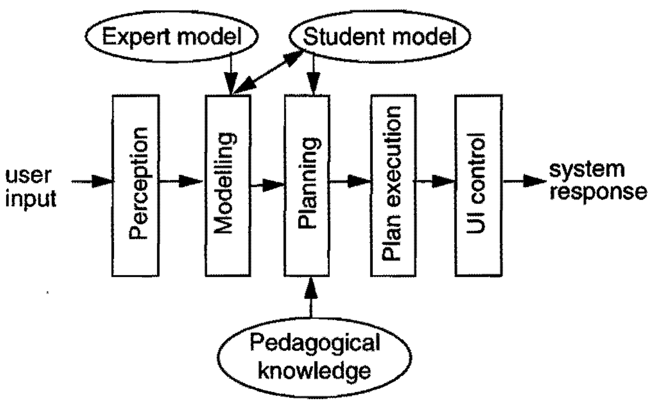


Figure 2: Architecture of an ITS.

These functional modules are developed largely independently of each other, often by specialists in the distinct areas. Completeness is aimed at in the design of all modules. The main advantage of this kind of decomposition is that in the ideal case it leads to modules that can be used in all kinds of applications. For instance, a vision system which is as good as human vision can be used in all kinds of robots with tasks varying from cleaning buildings, tutoring, to performing medical operations, etc.

There are, however, also many disadvantages to such an approach. Firstly, the system can only function when all modules are available. This implies that it is only possible to test the system at a late stage of the design process when all the modules have been completed. It also implies poor robustness: the system will be unable to function when one of the modules breaks down. Secondly, it is very difficult to make the modules complete. The aim of completeness can result in the fact that the distinct modules will never be put together. Moreover, there is the risk that when the modules are put together, the interfaces do not match. Last but not least, because information is processed sequentially in the many layers, there is no close coupling between perception and action, and the architecture is therefore not suitable for unpredictable and dynamic environments.

In a very different approach, Brooks (1986) decomposes an architecture into task-achieving modules, also called layers of competencies, as in the case of robots wandering around, avoiding objects, and exploring. Layers are not designed in parallel, but one at the time. Every layer extends the existing ones and, consequently, uses the skills present in the lower layers to compose higher level skills. The architecture is hierarchical in the sense that higher levels subsume the behaviours of lower levels. Such an architecture is called a subsumption architecture. In this case, the main aim of designers is to have a working (though elementary) system as soon as possible, which can be improved and extended iteratively. So, the agent functions as soon as the first layer has been constructed, and the addition of every layer results in a working system with more functionality. In every layer, a close coupling between perception and action is possible. Some form of perception is already present in the lowest layer.

This approach has several advantages. Firstly, the system can be tested quite early in the design, in fact every time a new layer has been completed. Secondly, this architecture makes the system more robust: when a higher level breaks down, the system keeps on working, although with a lower functionality. Thirdly, layers can be reused in all kind of applications. For instance, an object-avoidance layer can be used in all kind of robots. Lastly, because of the close coupling between perception and action, this kind of architecture is suitable for unpredictable and dynamic environments.

There are also some disadvantages, however. Firstly, the system will break down completely when the lowest level breaks down. Secondly, one of the major advantages of a decomposition of an architecture is lacking: the time gained by the parallel development of components by different persons. In this architecture the different components, i.e., layers of competence, are developed sequentially. Thirdly, in Brooks' subsumption architecture, the higher they are in the hierarchy, the more the layers of competence diverge from the principle of direct action on perception: layers such as reasoning and planning are mentioned. So, instead of adding new competencies based on more elaborate perception, new competencies are added as elaborations of the old ones, thereby making the distance between perception and action greater. This implies the same disadvantage as the functional architectures have, namely the danger that when the number of layers grows, the architecture becomes less suitable for unpredictable and dynamic environments.

We propose an architecture that includes all the advantages of Brooks' subsumption architecture, but without the disadvantages. As we see it, the main contribution of Brooks' architecture is its incremental nature: the aim to have a working system at each step of the design, where each component has a closed loop between perception and action. We adopt this in our architecture. We do not, however, adopt the hierarchical nature of Brooks' decomposition. The competencies added in a new layer should preferably not rely on the competencies already available, but operate in parallel. This has the advantage that the system becomes more robust: the high dependency on the lowest layer disappears. Another advantage is that this makes the architecture more suitable for unpredictable and dynamic environments.

In addition, we extend Brooks' architecture with the division of a complex agent into simpler interacting agents (Minsky, 1986). This makes it possible to work in parallel with various persons on different components of the system, viz. the different agents. Each agent represents a functionality of the system. The main differences as compared with the traditional functional decomposition are that now each agent has its own perception-action loop.

2.4.2 Communication between the agents

The main motivation behind the decomposition of an agent into sub-agents is that each agent can be made by a different designer. A designer is supposed to know everything about the agent he or she is designing, but may have a minimal knowledge about the other agents. The less knowledge a designer needs about the other agents, the less communication and agreements between the designers are needed, and hence the larger the advantages of decomposition can be, as regards both the time gain and the extensibility and updatebility of the system.

So, on the one hand we would like the designer to have minimal knowledge about the other agents, but on the other hand the agents still have to be designed such that their cooperation results in the desired behaviour of the complex agent. The latter is often seen as a reason to have explicit communication or interfaces between the agents, which requires a lot of agreements between the designers. However, there are many simpler ways to design agents in such a way that they can cooperate. We will describe eight ways, or levels, in which agents can influence each others behaviour in an increasing order of effort needed by the designer (see Figure 3). For each subsequent level the additional effort will be discussed.

1. The easiest way in which agents can influence each other's behaviour is by simply *being in the same environment*. By its actions, an agent causes changes in its environment. These changes may affect the behaviour of other agents which operate in the same environment. This is especially true in a situated approach, where the emphasis lies on a close coupling between perception and action. No effort at all is required from the designer for this primitive, but very natural, form of communication. The designer does not even have to know that there are other agents. Imagine a system in which agents search and retrieve information for users on the Internet. The cost for an agent to obtain certain information may depend on the demand for it by other agents. The time it takes the agent to retrieve information may depend on the load of the processor, which depends on other agents using it. So, the agent is influenced by other agents without any need for it or its designer to be aware of the existence of the other agents.

Use skills of other agent
Negotiation
Request other agent to perform task
Observation of other agents
Use of social rules
Use of environment to exchange information
Use of existence of other agents
Existence in the same environment

Figure 3: *A layered model of different types of agent communication.*

2. *The existence of other agents* may also be employed in the design. To do this requires, however, the designer has to know that there are other agents, and what the task of these other agents is. Suppose that the designer knows that there are transportation agents whose task is to deliver information left at certain places, say in mailboxes, to the user. Then the task of the information agent can be simplified to collecting information and leaving it in a mailbox, because the transportation agents will deliver it to the user. This makes the information agent more efficient in its task, and still guarantees

that the information reaches the user. It does not, however, require any internal representation of the existence of other agents inside an agent.

3. An agent can be designed to *use the environment to exchange information* with other agents. This requires agreements amongst the designers concerning the semantics of representation of information, and agreements that certain information will be given by the agents. For instance, in the case of information agents, the agents can leave notes to tell other ones in what direction certain information can be found. They can leave "footprints" while walking through the network, thus giving each other a way to predict the load of certain parts of the network. In software agents, this can, for example, be implemented by means of global variables. In order to minimize the agreements needed among the designers, the information provided must be kept as simple as possible, for instance using Booleans or simple counters instead of complex data types. Instead of having agreements regarding the semantics, it could also be possible to let the agent learn the semantics. This requires however that the designer provides the agent with a way to learn, which may require more effort than communication with the other designers and also takes the agent more time before becoming effective.

4. The designers can also agree on *"social rules"*, which tell an agent what to do or what not to do when it perceives certain information left by another agent. For instance, the designers in the example could agree that an agent is forbidden to enter a certain part of the network when it notices by the "footprints" that more than, say, five other agents are in that part. An example of a social rule in software agents is the use of a semaphore, in the technical computer science sense (Dijkstra, 1968), to ensure that certain actions of an agent cannot be interrupted by another agent.

5. The other agents are part of the environment of an agent and are a probable cause of changes in that environment. So, an agent could *observe the actions of other agents* in its neighbourhood in order to get a better prediction of future changes in its environment and thus anticipate them. This, however, requires the possibility for the agent to perceive the action of another agent even before it causes a change in the environment, or to recognize in some way what the other agent intends to do. Chapman (1991) argues that intention recognition is fairly easy in most concrete domains for the same reason that concrete action is easy: the situation not only tells an agent what to do, it also tells the agent what other agents are up to. However, this kind of intention recognition only works when agents have similar skills and personalities, or when the observing agent has a knowledge of the skills and personality of the observed agent. Apart from these difficulties, the designer should also have a knowledge of the relationship between a certain action of another agent and a subsequent change in the environment, or should equip the agent

with the means to learn this relationship. In the example of the information agents, an agent could observe that a lot of agents intend to obtain a certain piece of information, and that therefore the cost of obtaining that information will probably increase.

6. Besides cooperating in the ways mentioned above, agents can also do this more explicitly. They can *request other agents to perform a certain task*. As we see it, these kinds of requests should have a close connection to a desired change of the environment. For this form of cooperation, it is not necessary for the agent to know (or for the designer of the agent to know) the skills of the other agents. The agent can request another agent to perform a certain task or action, and can see whether the desired effect occurs. If it does not, then the question can be put to another agent, or the agent could do it itself. This, however, requires a representation of actions or tasks common to all agents (designers) and the agents must also have an insight into their own capabilities. In the example, an agent could ask to search for information about a certain topic and leave it at a certain place.

7. It is also possible for agents to answer requests, and that there is a form of *negotiation*. It is, however, not necessary to program negotiation explicitly into the agents. The negotiation process can emerge from simple rules such as "when an agent asks me to do something which I am able to do, and offers me a lower reward than I want for this job, I will answer that I can do it for more". This, however, requires an agent to have an insight into its own interests. In the example of the information agent, a negotiation could take place about the exchange of information or the exchange of favours regarding the obtaining of information for another agent.

8. Finally, a very elaborate form of communication is *using the skills of another agent*. The agents can be designed in such a way that they can perceive which skills another agent has. This is also suggested by Steels (1994). This can make observation and negotiation a lot easier and more effective, but it requires a common representation of the capabilities of an agent which is visible to all (designers of) agents. Another way to use the skills of an agent is for a designer to copy parts of the implementation of that agent inside his or her own agent, or for the agent to do this at run time, thereby benefiting in a very efficient way from the experiences/learning of a seemingly effective agent. For instance, an information agent could copy strategies for information retrieval from another agent which seems to be very good at this.

2.5 Consequences of the Situated Approach for the Design of an Agent

2.5.1 Identification of the agents

As described in the previous section, a complex agent can be decomposed into a collection of simpler agents. In the same way, such a simpler agent can be decomposed into a collection of even simpler agents, etc. Therefore, the first important step in the design of an agent-based system (apart from deciding what kind of agents one will use) is to determine when to stop decomposing an agent into simpler agents (this can be called the level of granularity) and to identify what the agents will be. It is not necessary for all the agents to be the same. On the contrary, heterogeneity, i.e., the existence of different skills and personalities for the different agents, is frequently present in biological systems.

Two main rules, or heuristics, can be given to identify what the agents are. In the first place, agents should be manageable units of thought for the designer. This means that each agent should have a functionality such that the designer is confident that a first version of it can be designed without having to worry about too many problems in parallel. *When there are different functionalities related to the task of an agent which require independent investigation, an agent should be associated with each of these functionalities.*

The second rule can best be used during the design process. *When different behavioural patterns can be used by an agent, the agent should be divided into sub-agents, where each sub-agent uses another behavioural pattern.* The main advantages of this are that evolution-like learning can take place, and that experiments can be done to determine which behavioural pattern is the best in a certain situation. Another advantage is that the patterns can be developed independently.

2.5.2 Design of an individual agent

As we have already argued when describing our architecture, agents have to be designed in a bottom-up fashion, beginning with very simple behaviours which define a minimal functionality. Each agent is responsible for certain tasks, and the main challenge for the designer of an agent is to propose those behaviours of the agent which accomplish these tasks. For the five lowest levels of communication described above, it is not necessary to build a representation of the task in the agent's implementation: the task can mainly exist in the head of the designer or an observer of the agent.

In a situated approach, a behaviour cannot be decoupled from the environment, so a behaviour is not the same as an action. Basically, each behaviour consists of two parts: an action of the agent and a situation in which this action is taken. We will use the notation *s-->a* to denote a behaviour in which action *a* is taken in situation *s*. The task of the designer is to establish a set of behaviours of this kind. Hence, the first step in the design is to determine one such behaviour. This means that both the action and the situation in which the action should be taken must be determined. There are two possible approaches. The first is to start by determining the most obvious action the agent should take at some point in view of its task, and then to determine a situation in which that action should obviously be taken. The second approach is the reverse of this: first determine the most obvious situation in which the agent, in view of its task, should do something, and then determine what action would be the most appropriate in that situation. Our experience is that in most cases a combination of these approaches must be adopted, using both knowledge about what simple actions related to the task the agent can take, and knowledge about what the agent can perceive of its situation related to its task with a simple form of perception.

A description of the agent's situation consists mostly of a part concerned with perception from its environment and a part concerned with internal state. Note that internal state can be interpreted by an observer as the agent having memory or even desires. A description of a possible action of the agent consists mostly of a part concerned with actions which directly affect its environment, which we will call motor actions, and a part concerned with changes in its internal state. Of course, it is also possible that there are behaviours of the agent which do not use sensory input directly or do not result directly in motor actions. These behaviours are part of an internal process. However, the situated approach advocates a minimum use of these kinds of behaviours.

Initially, for the first behaviour, only sensory input will be considered for the description of a situation, and only motor actions as possible actions in that situation. This results in a behaviour *p --> a*, where *p* denotes a certain sensory input and *a* a certain motor action. For instance, in the case of a robot that has to avoid obstacles, the first behaviour could be "*obstacle in front*" --> "*turn away*". The action "*turn away*" can be implemented as the motor action "*increase speed of one of the motors*". The situation "*obstacle in front*" is more difficult to implement. This relates to the fact that it is sometimes not easy to perceive whether a certain situation holds, and that the agent has to perform further actions to improve its perception. These are the epistemic actions mentioned above. For instance, when the robot emits infrared (IR), it can sense by an IR sensor whether there is an obstacle nearby. In that case there would be a behaviour "*always*" --> "*emit IR*", and "*obstacle in front*" can be implemented as *"front IR sensor receives signal exceeding certain strength"*. In the following, we will not go into implementation details, but will describe behaviour at the more abstract level.

2.5.3 Differentiation of an agent's behaviour

Determining the first behaviour may not be straightforward. In the first place, a problem may arise when trying to determine what action the agent should take in a given situation. More than one action may seem appropriate. It may be possible to generalize these actions into one description that fits all of them. This is what we have done in case of the robot avoiding obstacles: we did not worry about the difference between "*turn left*" and "*turn right*", but generalized it into "*turn away*". However, in order to make the behaviour effective, randomly choosing between appropriate actions during run time or a priori may not be the best idea. For instance, randomly choosing between "*turn left*" and "*turn right*" during run time could drive the robot towards the obstacle instead of enabling it to avoid it. For each possibly appropriate action it should be determined in which situation that is a special case of the original situation it can be best used.

A second and related problem may arise when trying to determine in which situation the agent should take a certain action. More than one situation may seem appropriate. It is possible to generalize these situations into one description that fits all of them. However, by analogy with the previous case, we prefer to determine for each of the situations the most appropriate action that can be used which is a special case of the original action.

Specifying a special case of a situation actually comes down to specifying a situation in greater detail. This can be done in two ways. The first is to describe the sensory input in more detail, either by using more of the information provided by the sensors involved, or by using information from other sensors as well. For instance, in the case of obstacle avoidance, we could get the following two behaviours:

(*"obstacle in front"* & *"left side of robot is nearer to obstacle than right side"*)
 --> *"turn right"*,

(*"obstacle in front"* & *"right side of robot is nearer to obstacle than left side"*)
 --> *"turn left"*.

To determine the second conjunctions, a front-left and a front-right IR sensor could be used. This also illustrates that it is not necessary to determine the kind of sensory input to be used entirely before the design process. The kind of sensory input needed will automatically become clearer during the design process.

The second way to specify a situation in more detail is to use the internal state (or memory) of the agent. This can be done when the history of the interaction makes a crucial difference between two situations which are identical from a perception viewpoint. For instance, in obstacle avoidance it may be important that the robot is not too unstable, which means that once it has decided to turn left or right it will stay with that decision for some time. Internal state variables could be used to represent whether the agent has decided to turn left or to turn right, say *decided-turn-left* and *decided-turn-right*. We could then get the following set of behaviours:

{
  ("*obstacle in front*" & "*left side of robot is nearer to obstacle than right side*"
    & no(*decided-turn-left*)) --> { "*turn right*", *decided-turn-right* },
  ("*obstacle in front*" & "*left side of robot is not nearer to obstacle than right side*"
    & no(*decided-turn-right*)) --> { "*turn left*", *decided-turn-left* },
  ("*obstacle in front*" & *decided-turn-left*) --> { "*turn left*" },
  ("*obstacle in front*" & *decided-turn-right*) --> { "*turn right*" }
}

The action part of the behaviours now also includes changes in the internal state of the agent. Of course the internal state variables do not have to be Boolean as in the example, but can assume all kinds of values. So, a possible action of an agent could be to modify the value of one of its internal state variables. As mentioned above, not every behaviour necessarily includes a motor action. There may be situations in which only the internal state of the agent changes without leading directly to a motor action.

To be able to make full use of a history of interaction, an agent should not only have an internal state, but also a sense of time. It is then possible that an action may be caused by the fact that the agent is in a situation for a certain amount of time. The time passed can also be used to cause a change of internal state. In the case of the obstacle avoidance, we can use this to include, for instance, the behaviours

decided-turn-right \ 2 sec --> { no(*decided-turn-right*) },
decided-turn-left \ 2 sec --> { no(*decided-turn-left*) },

where $s \setminus t$ denotes that the agent is in situation s for time t. This enables the robot to make new decisions every two seconds, based on its perception of its position with respect to the obstacle.

We will call the process explained above the *differentiation process*. Briefly, the differentiation process starts with one behaviour. In the process, distinctions are made within the situation part and/or action part of this behaviour, which leads to a set of behaviours. Typically, such a set of behaviours resulting from a differentiation process looks like this:

```
{ (p & p0 & z0) \ t --> { a0, c0 } ,
  (p & p1 & z1) \ t --> { a1, c1 },
  (p & p0 & z1) \ t --> { a2, c2 }
},
```

where *p, p0, p1* denote sensory input, *z0, z1* denote internal state, *t* denotes an amount of time, *a0, a1, a2* denote motor actions, and *c0, c1, c2* denote changes in internal state. The differentiation process may not only be necessary to get behaviour which is in some way effective. It is also an important way to improve the adaptivity of the agent.

2.5.4 Behavioural engineering

When a minimal functionality has been implemented, it can be evaluated by carrying out experiments, the outcome of which determines the update and inclusion of behaviours in the architecture. We call this design process based on iteratively adding behaviours and testing these behaviours by running experiments 'behavioural engineering'. The outcome of such experiments can also be used to update or construct models of human or animal behaviour, which in their turn can be used to find new behaviours.

The differentiation process does not only take place initially, but can be done during the whole design phase. So, a first way to use the outcome of the experiments or tests can be in the differentiation process: as a basis of knowledge about what special situations should be distinguished within the situation part of a behaviour, and what simple variations of the action part could be used for these special situations.

The outcome of the experiments and tests could be that the current behaviours of the agent do not completely produce the desired effect. The coupling between situation and action can mostly be done in such a way that each behaviour contributes obviously (perhaps after a differentiation process, or fine tuning) to approaching a desired situation for the agent. But there is no guarantee that the desired situation will be reached closely enough. Possibly more than one action should be taken in a certain situation. For instance, in the example of obstacle avoidance it may occur that the robot in fact tries to turn away from the obstacle but still collides with it. This may be due to the high speed of the robot. In that case braking when perceiving an obstacle might help. It can also happen that certain situations, different from those anticipated in the current behaviour set, might occur in which it is obvious to the designer that the agent should perform an action, which may be different from the actions in the current behaviour set. So, a second way to use the outcome of the experiments or tests is for the designer to include a new behaviour, which does not necessarily (in contrast to behaviours originated by a differentiation process) have anything in common with the existing behaviours.

2.5.5 How can an agent learn?

Two types of behaviour can be distinguished in an agent: pre-programmed behaviours and learned behaviours. The majority of the agent systems described in the literature behave in an adaptive manner because they have been explicitly programmed in that way. In the case of learned behaviours, an agent simulates learning processes by improving existing behaviour or acquiring new behaviour.

There are three main reasons for building learning agents. On the one hand, from a cognitive science point of view, a theory of learning is a necessary component of any theory of activity. Until recently, the predominant view was that one should first study the performance or behaviour continuum before the problem of learning could be tackled. As a consequence, the issue of learning has long been neglected. The impetus of recent developments in artificial life, situated agent research, cognitive psychology, and robotics has resulted in a renewed interest in learning. It might be argued that the reverse view is getting more support, i.e., the study of learning is needed before any other behavioural phenomena can be studied, because the best approach to understanding these phenomena may be to study how they develop with experience (Chapman, 1991). A theory of learning is also an engineering problem. There are several demonstrations of agents engaged in a complex activity based on a limited number of rules. However, as these systems get larger, it becomes impossible to program them entirely by hand. The need for learning is also a natural consequence of the situated view. If one argues that agents are mostly embedded in dynamic, not entirely predictable environments, then it is necessary for an agent to learn about its environment in order to act in a robust way.

How then do agents learn or evolve? Several learning processes have been proposed. Many of the models proposed take animal and/or human learning theories as a framework. We do not propose a specific theory of agent learning, but introduce a layered model of different types of learning (cf., Figure 4), in which current models of agent learning can be situated. The layered structure of the model implies that each subsequent layer subsumes the previous one and that higher layers are computationally more demanding than the lower ones.

1. The most basic form of learning is what we have called *memory representation and processing*. This is probably a controversial point of view. Many researchers in the field (e.g., Brooks, 1991) argue against the use of (memory) representations. However in view of the generally accepted hypothesis that animals (humans) are useful models for the development of robotic or synthetic agents, there is substantial evidence that learning organisms change as a result of their experience, and those changes are representations (Roitblat, 1994). That does not imply, however, that these representations are symbolic in nature. Hence, the basic principle of the layered model is that if experience acquired at one time is to affect the behaviour of the agent at another, then the agent must have some means of representing that experience. In its simplest form this means that the agent can represent and process facts about the world.

Teaching
Imitation
Projection
Generalization
Specialization
Refinement
Memory representation and processing

Figure 4: *A layered model of different types of agent learning.*

2. The second layer consists of the *refinement mechanism*. The idea of the refinement learning mechanism is that the agent attempts to learn how it can optimize the reward it receives for taking actions in certain situations. Hence the task of the refinement mechanism is to evaluate the agent's behaviours, improving them where possible, and favouring the better ones in application. Examples of refinement mechanisms are reinforcement learning (Maes, 1995; Kaelbling, 1995) and the Bucket brigade algorithm (Holland, 1975).

3. The third learning layer proposed is *specialization*. The idea of specialization is that the agent improves its behaviour by learning new conditions. A behaviour has a condition list, which are characteristics of the environment that have a certain value. If the behaviour of an agent is not reliable, i.e., the (positive or negative) feedback is not consistent, then one or more additional conditions are added to the behaviour in order to make it more reliable (Maes & Brooks, 1990). Examples of specialization are the learning algorithm of Maes and Brooks (1990) and classifier systems (Holland, 1986).

4. *Generalization* is the counterpart of specialization, since it enables a (new) behaviour to be applied in more situations than before. Several generalization mechanisms have been proposed, in which sometimes a distinction is made between exemplar-based or primary generalization and abstraction-based or secondary generalization (Vandierendonck, 1995). For a comprehensive taxonomy of generalization, see Dietterich and Michalski (1983). The essence of generalization is that an agent improves the application scope of its behaviour by means of (1) the simple deletion of part of an existing behaviours condition list, (2) inducing a new behaviour on the basis of the intersection of the condition list of two or more similar behaviours, and (3) producing a new rule that contains variables in its condition list which are a generalization of two or more examples. Several models of generalization learning exist, and these models usually include a simulation of the specialization mechanism. Examples are the already mentioned classifier systems (Holland, 1986), neural networks (McClelland & Rumelhart, 1986), genetic algorithms (Goldberg, 1989), exemplar-based learning (Kruschke, 1992) and memory-based learning (Stanfill & Waltz, 1986).

5. What all the learning processes so far described have in common is that learning is driven by feedback from the real world. Another, computationally more advanced, possibility is that the agent is able to project likely outcomes of events, and evaluate (i.e., refinement, specialization, and generalization) its behaviour on the basis of feedback from this projected world. *Projection* is the registration of what would happen in alternative possible future circumstances. Several means of projection are possible. The most complex one, which has been identified by some authors as incompatible with neurally plausible hardware (Chapman, 1991), is simulation. In this case, an agent builds a model of the current situation and applies rules that transform it according to its projected behaviours. Another possibility is visualization. For example, in Chapman's (1991) system Sonja, visualization projection is accomplished by running visual routines which register what would happen in alternative possible future situations, e.g., *this-monster-will-hurt-me-if-I-don't-do-something-about-it*.

6. So far we have considered learning at the level of one agent. There are, however, important learning mechanisms in which more than one agent is involved. The first one is *imitation*: an agent can learn by observing another agent performing the task. For real robots, this requires perceptual abilities that are not yet available (Kaelbling, 1995). In the case of software or synthetic agents, it is possible for an agent to "observe" the behaviour of another agent and to imitate this behaviour by copying the code, as has been described in the section on the communication between agents.

7. The upper learning layer in our model is *teaching*. The idea of teaching is that an agent can learn new behaviours or improve existing behaviour on the basis of the instructions of another agent. Imitation can be part of the teaching process. An example of teaching in the case of real robots is to have a human agent supply appropriate motor commands to a robot through a joystick or steering wheel (Pomerleau, 1993). In the case of software agents, Lieberman (1995) developed a technique called "programming by example" for teaching agents new behaviours by demonstrating actions on concrete examples. A similar procedure is possible between two software agents and again relies very much on the communication procedures described in the communication section.

Several comments must be added to this model of agent learning:

- The selection of the number and the type of learning layer is based on the same heuristic as in the communication model: learning is based as far as possible on the lower, computationally less expensive levels before turning to the higher levels. This also implies that learning is not limited to one mechanism, but involves several layers at the same time;

- Most of the learning mechanisms in the layered model rely on existing behaviour (e.g., specialization and generalization), which implies that tabula rasa learning techniques have in themselves a limited problem-solving range (see also Kaelbling, 1995);

- Learning is conceived as an incremental process as far as possible , i.e., learning is embodied in each action of the agent (e.g., reinforcement learning);

- Learning is a distributed process, i.e., in the case of our multi-agent architecture each agent refines and acquires behaviours according to its field of competence;

- Learning in a multi-agent architecture introduces redundancy in a double manner. First, different agents can acquire the same behaviour(s) which makes the system more robust, i.e., when an agent fails the system will not break down completely. Secondly, the system can consist of several agents for a particular competence. In these cases, learning mechanisms will sort out which agent acts in the most competent manner in the long run.

2.6 An example of the general approach: the architecture and design of TE

2.6.1 Architecture / Identification

Inspired by the literature on aspects of teaching that have to be adapted to the individual student (see Section 1.2.3), the following five functionalities of TE have been identified:

- *Navigation*

TE navigates through the course material, determining at any given time which topic to discuss. The rate of presentation and the order in which the material is taught depend on the performance and interests of the student.

- *Practice*

Within a given subject, those exercises and examples are given that correspond to the performance of the student.

- *Explanation*

In the case of low performance or absence of responses, the student is instructed and assisted with the task. The amount of explanation needed depends on the student.

- *Feedback*

Feedback on performance in a task is adapted to the student. A distinction is made between errors and slips, and the degree of enthusiasm or disappointment varies to keep the student motivated.

- *Presentation*

The way in which material is presented is adapted to the student's performance. For instance in mathematics, depending on the level of the student, additions are presented using formulas, pictures, or even real objects.

An agent has been associated with each of these functionalities, and we designed first-versions of these agents. This does not, however, imply that our system will always consist of these five agents. During the design of some of the agents, it turned out that their skills could be extended in all kinds of different directions; that there are indeed sub-functionalities which require separate research. It has, for instance, been suggested that requests of the student for the dictionary should be handled by a sub-agent of the Explanation Agent.

So, the agents will possibly have to be divided into more restricted sub-agents before extending their current skills. It might also be possible to develop different variants of agents for the same functionality, and exploit evolution-like learning to determine which agent will be used (survives) for a particular (group of) student(s).

In order to minimize the effort required from the designers to add new agents, from the course makers to choose between different agents which implement different teaching methodologies, and from the students to choose which functionality of the system, i.e., which agents or behaviours of the agents they want, the following measures have been taken:

- The agents have been designed to operate in parallel and quite independently of each other.
- There is no control component within TE: the functionality of TE emerges from the interaction between the agents and the student.
- Each agent has its own memory, in which it mainly stores interaction sequences with the student. There is redundancy in the memories of the different agents.
- There is also no control component within an agent: the functionality of each agent emerges from the interaction of behaviours or learning rules inside the agent.

In fact, we have a demonstration in which it is possible to remove or add an agent at run-time, to choose between different learning principles, to remove or add learning principles, and to change parameters of the learning principles (for instance the patience of the Explanation Agent).

2.6.2 Design of the Individual Agents

In the design of the individual agents, we have used the available knowledge about the human learning process. We will not describe the complete design of each of the agents, but will give some examples to illustrate the general design process explained above. More details on the agents can be found in the successive chapters of this dissertation.

The Feedback Agent.
The main task of the Feedback Agent is to give feedback adapted to the student. An obvious situation in which the agent has to act, is when the student has finished an exercise. So, the situation part of the first behaviour will be "*student has finished exercise*". A general description of the action the

agent should take in that situation is "*give feedback*". However, two possible actions which fit this description immediately come to mind, namely "*give positive feedback*" and "*give negative feedback*". Thinking of the situations in which each of these actions should be preferred leads to the following behaviours:

("*student finished exercise*" & "*exercise done correctly*")
--> { "*give positive feedback*" },

("*student finished exercise*" & "*exercise done incorrectly*")
--> { "*give negative feedback*" }.

In order to keep the student motivated, feedback depends on the student's results on the previous exercise. Therefore, an internal state variable *last-exercise-done-incorrectly* is introduced. A differentiation on the first behaviour using this internal state leads to the following set of behaviours:

("*student finished exercise*" & "*exercise done correctly*"
& *last-exercise-done-incorrectly*)
--> {"*give strong positive feedback*", no (*last-exercise-done-incorrectly*) },

("*student finished exercise*" & "*exercise done correctly*"
& no (*last-exercise-done-incorrectly*)) --> { "*give weak positive feedback*" },

("*student finished exercise*" & "*exercise done incorrectly*")
--> { "*give negative feedback*", *last-exercise-done-incorrectly* }.

The Feedback Agent also tries to make a distinction between real errors and slips, i.e., errors that occur when someone performs an action unintentionally (Norman, 1981). In case it suspects a slip, for instance when the student makes a mistake in an exercise which has been answered correctly before, the student gets a second opportunity to do the exercise. For further details on the Feedback Agent see Chapter 7.

The Explanation Agent.

The Explanation Agent should give an explanation adapted to the student whenever the student needs it. It is an important agent in the system as it not only determines the adaptivity, but also the interactivity level of the system. The behaviour of the Explanation Agent is determined by the rule "If the student succeeds, when next intervening offer less help. If the student fails, when next intervening take over more control." (Wood et al., 1978). Furthermore, the Explanation Agent is based on a layered model of intervention varying between the "general verbal encouragement" level and the "demonstration" level, whereby each layer represents a different level of control on the part of the agent. This implies that the behaviour of the Explanation Agent is contingent on the performance of the student (cf., situatedness principle); the most appropriate intervention is chosen on the basis of the student's success.

Currently our Explanation Agent uses a quite simple form of layered intervention, with only two levels. It includes the following behaviours, among others:

(*"student has not done anything"* & *student-supposed-to-exercise*) \ 25 sec
--> { *"remind student that he or she should do something"*, *have-reminded-student* },

(*"student has not done anything"* & *student-supposed-to-exercise*
& *have-reminded-student*) \ 15 sec --> { *"give student a hint"* }

This illustrates the use of a combination of internal state variables and timing in the differentiation of situations. For more detail on the Explanation Agent see Chapter 7.

The Navigation Agent.

The Navigation Agent navigates through the course material in response to the performance and interests of the student. Its main behaviour is based on the notion of controlled complexity as described in Wood et al. (1978). The student should be confronted with problems that lie beyond his current level of competence but not so far beyond that he or she is unable to master the problem being presented. Problems should be neither too easy nor too difficult.

After a large differentiation process starting from the behaviour *"student has finished exercise"* --> *"navigate in course material"*, which has led to around 16 behaviours, another situation was identified in which the agent obviously should do something, namely when the student wants to do something different. This led to the inclusion of another behaviour and a new start of the differentiation process.

To determine which new topic or exercise to address, or to which to return, information from the Instruction Database and the agent's memory is used. In the Instruction Database information can be found about which topics or kinds of exercise are considered beforehand to be the most difficult. The agent's memory provides information about which topics and exercises a student has already encountered and his or her performance on them. For more details on the design and evaluation of the Navigation Agent see Chapter 6.

The Practice Agent.

The Navigation Agent determines which topic and kind of exercise will be addressed. In most cases, however, the topic to be taught consists of a number of items, the relative difficulty level of which is not necessarily known beforehand. For instance, when the topic is to study a set of words, there could be an item for each word. The Practice Agent decides at any given time which item will be presented in an exercise or example. For further details on the design and evaluation of the Practice Agent see Chapter 3,4, and 5.

An essential factor for the behaviour of TE is that a tutorial dialogue with a mixed, varying locus of control emerges from the cooperation between the Practice Agent (i.e., controlled complexity) and Explanation Agent (i.e., contingency rule and layered intervention model). As long as the student cannot master the problems he or she is confronted with, TE is more or less in control. Once the student succeeds in solving the problem, he or she takes over more control. The layered intervention model of the Explanation Agent also encompasses different types of IIS's. The 'general encouragement' level corresponds to Microworld systems which are user-controlled systems, whereas the 'demonstration' level corresponds to classical ITS's in which control is in the (virtual) hands of the system.

The Presentation Agent.

The Presentation Agent tries to choose the optimal presentation form for the student at each moment in the instruction session . It is based on a layered model, whereby each layer represents a different level of abstraction. The general rule determining the behaviour of the Presentation Agent is that information is presented at the abstract level (e.g., formulas when teaching addition). However, when a student is having difficulty with a particular exercise or is asking for a further explanation, the Presentation Agent can decide to focus in on the problem in another, more concrete representation format (e.g., using pictures or video films). For more details on the Presentation Agent see Chapter 7.

2.6.3 Communication between the Agents

In TE, the agents mainly communicate by causing changes in the environment, i.e., the student and the position in the course material. They have deliberately been given complementary tasks. For instance, the Feedback Agent would not encounter many situations in which to give feedback if there were not a Practice Agent presenting exercises to the student. However, the designer only needs to know the task that the agent is supposed to do.

It proved to be rather difficult to give the agents the ability to perceive certain events directly. Hence, the agents exchange some information via the environment. For instance, some agents (e.g., the Explanation Agent) need to be able to perceive when an exercise is presented to the student. Therefore, the Practice Agent notifies all agents when it presents an exercise. An extra agent, called the Postman, has been added to broadcast incoming messages from the domain-specific components of the application (i.e., Domain Expert and Interface Manager) to all other agents. The addition of this agent is necessary since it is the only one able to perceive and translate these messages to the other agents of TE. This required agreements from the designers about what kind of information to provide, and the semantics of it. For instance, when a results sign is given, all the agents know that the student has finished an exercise and that the results of this exercise are available in a global variable.

In the design of the agents, some social rules have been imposed to prevent certain situations. For instance, the following kinds of behaviours exist in the agents:

"student finished exercise" --> { *"present exercise"* }
"student finished exercise" --> { *"give feedback"* }
"student finished exercise" --> { *"navigate in course material"* }.

To avoid the reaction of one agent –e.g., the presentation of a new exercise by the Practice Agent– being disturbed by the reaction of another agent, e.g., giving feedback by the Feedback Agent, semaphores are used. The social rule in this case is that an agent may only cause changes in the environment in reaction to an event, e.g., the *"student finished exercise"* event, when it owns the semaphore belonging to this event. This can be done with the following kinds of behaviours:

(*"event"* & *semaphore-event* & *preconditions*) --> { no(*semaphore-event*), *"reaction"* },
"reaction finished" --> { *semaphore-event* },
where *preconditions* denotes the use of internal state: the agent does not always have to react to an event; this may depend on its internal state. To use this construction for a certain event, designers have to agree to associate a semaphore with the event and to respect the social rule regarding semaphores. However, the designers do not have to know what other agents exist, or how the agents will react to that event.

The use of a semaphore does not guarantee that, for instance, feedback will be given before the presentation of a new exercise. For that purpose the notion of time is used: each behaviour ("*event*" & *semaphore* & ...) --> ... has been changed into ("*event*" & *semaphore* & ...) \ *time* --> ... In this way, the order in which the agents can react to an event is determined by their relative speed, or more specifically the time annotation given to their behaviours concerning that event. So, by giving the Practice Agent a higher time annotation than the Feedback Agent, the latter may act first. If the Feedback Agent does not want to react, or even does not exist, time will expire and another agent will automatically be allowed to react. Currently, these time annotations are determined by the designers, but it may well be that the agents will learn these annotations themselves in future applications. For instance, an initial learning phase could be employed in which the user indicates whether a certain sequence of the agents' actions is acceptable or not.

To conclude, in our current application the agents are completely unaware of each other's existence. This has the advantage of enabling us to develop the agents quite independently, and remove, change, and add agents without much difficulty. The only information the designer of a new agent needs is the agreements on the semantics, the social rule regarding the semaphore, and a list of time annotations.

2.6.4 How the Agents learn

In TE, each agent is equipped with a rudimentary memory-based learning process. The agent stores interaction sequences in its memory, and the agent's behaviour is tuned on the basis of the statistics of these episodic memory traces. For instance, among other things, the Navigation Agent keeps track of the correct and incorrect responses of the user. When the number of incorrect responses is quite high, the Navigation Agent can decide to present easier exercises.

2.6.5 Implementation

The implementation of the agents has been done in the RTA programming language. RTA is a concurrent and declarative programming language for developing agent-based systems. The language was developed at Philips Research Laboratories, U.K. (Wavish & Graham, 1994). The pseudo-code used in this chapter is very similar to the RTA code.

2.7 Conclusions

We have described an interactive instruction system in which a domain-independent tutor emerges from the interaction of a set of simple agents. Each agent consists of a set of behaviours that are based on knowledge and experimental research on human learning and human-computer interaction. The design process can be characterized by the combination of a differentiation method and behavioural engineering. One of the main characteristics of the behavioural engineering approach is that results of empirical research are incorporated in the design process.

It is argued that the architecture and design methodology described in this chapter is not restricted to interactive instruction systems. Therefore, the approach was first introduced on a more general level, and then illustrated with an interactive instruction system. The use of our agent-based architecture and design process for other kinds of adaptive user-interfaces or intelligent consumer products is worth investigating.

Throughout this chapter, the role of empirical evaluations in the design process of an interactive instruction system has been emphasized. In our view these evaluations are realized at two levels. First, the effectiveness and efficiency of parts of the IIS are experimentally tested and modelled according to the behavioural engineering approach. The behavioural engineering approach was briefly introduced in this chapter. For more detailed information see the next chapters. Secondly, much interactive instruction research focuses on a specific issue such as adaptive instruction and neglects the importance of the user-interface environment as a significant determinant of the user's learning performance. So, user-interface evaluations should be included in the evaluation of an IIS as a whole.

Future research should explore the use of genetic programming in the design process. Two major problems of machine learning such as genetic programming are (1) that they do not scale well to large problems, and (2) it is difficult to apply them on-line to model the incremental learning process of an agent. We therefore propose to decompose large problems into smaller subproblems and apply genetic programming techniques to those subproblems as an aid for the designer in finding the necessary behaviours of the agent he or she is developing.

References

Agre, P.E., & Chapman, D. (1987). Pengi: an implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial*

Intelligence. Menlo Park, CA: American Association for Artificial Intelligence.

Bates, J. (1994). The role of emotion in believable agents. *Communications of the ACM, 37,* 122-125.

Beun, R.-J., Baker, M., & Reiner, M. (Eds.) (1995). *Dialogue and instruction: Modelling interaction in intelligent tutoring systems* (Volume F142 of NATO ASI Series). Berlin: Springer.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation, 2,* 14-23.

Brooks, R. (1991). Intelligence without representation. *Artificial Intelligence, 47,* 139-159.

Chapman, D. (1989). Penguins can make cake. *AI Magazine, 10,* 45-50.

Chapman, D. (1991). *Vision, instruction, and action*. Cambridge, MA: MIT Press.

Dietterich, T., & Michalski, R. (1983). A comparative review of selected methods for learning from examples. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* Palo Alto: Tioga Press.

Dijkstra, E. (1968). Cooperating sequential processes. In F. Geneys (Ed.), *Programming languages*. London: Academic Press.

Forrest, S. (Ed.). (1991). *Emergent computation: Self-organisation, collective, and cooperative phenomena in natural and artificial computing networks*. Cambridge, MA: MIT Press.

Goldberg, D. (1989). *Genetic algorithms in search, optimization, and machine learning*. Amsterdam: Addison-Wesley.

Greeno, J.G., Chi, M.T.H., Clancey, W.J., & Elman, J. (Eds.) (1993). Situated Action. [Special issue]. *Cognitive Science, 17* (1).

Holland, J.H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. Cambridge, MA: The MIT Press.

Holland, J.H. (1986). Escaping brittleness: The possibilities of general purpose machine learning algorithms applied to parallel rule-based systems. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Volume 2). Los Altos, CA: Kaufmann.

Kaelbling, L. (1995). Reinforcement learning: A survey. In R. Pfeifer (Ed.), *Practice and future of autonomous agents*.

Kirsh, D., & Maglio P. (1994). On distinguishing epistemic from pragmatic action. *Cognitive Science, 18*, 513-549.

Kruschke, J.K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review, 99,* 22-44.

Lieberman, H. (1995). Letizia: An agent that assists web browsing. In *Proceedings of the international joint conference on artificial intelligence*, Montreal, August, 1995.

Maes, P. (1993). Behaviour-based artificial intelligence. In *Proceedings of the second conference on simulation of adaptive behaviour.* Cambridge, MA: MIT Press.

Maes, P. (1994). Agents that reduce work and information overload. *Communications of the ACM, 37*, 31-40.

Maes, P. (1995). Modelling adaptive autonomous agents. In C. Langton (Ed.), *Artificial life: An overview.* Cambridge, MA: The MIT Press.

Maes, P., & Brooks, R.A. (1990). Learning to coordinate behaviors. In: *Proceedings of AAAI-90.* Boston.

McClelland, J.L., & Rumelhart, D. (1986). A distributed model of human learning and memory. In J.L. McClelland, and D. Rumelhart (Eds.), *Parallel distributed processing* (volume 2). Cambridge, MA: The MIT Press.

McFarland, D., & Bösser, T. (1993). *Intelligent behavior in animals and robots.* Cambridge, MA: The MIT Press.

Minsky, M. (1986). *The society of mind.* New York: Simon and Schuster.

Norman, D.A.(1981). Categorization of action slips. *Psychological Review, 88*, 1-15.

Pomerleau, D.A. (1993). *Neural network perception for mobile robot guidance.* Amsterdam: Kluwer Academic Publishing.

Powers, W.T. (1973). *Behavior: The control of perception.* Chicago: Aldine Publishing company.

Riecken, (1994). Intelligent agents. *Communications of the ACM, 37*, 18-21.

Roitblat, H. (1994). Mechanism and process in animal behavior. Models of animals, animals as models. In: *Proceedings of the third conference on the simulation of adaptive behavior.*

Shoham, Y. (1993). Agent-oriented programming. *Artificial Intelligence, 60*, 51-92.

Simon, H. (1968). *The sciences of the artificial.* Cambridge, MA: MIT Press.

Stanfill, C., & Waltz, D. (1986). Toward memory-based reasoning. *Communications of the ACM, 29*, 1213-1228.

Steels, L. (1991). Towards a theory of emergent functionality. In J.-A. Meyer & S.W. Wilson (Eds.) *From animals to animats*, Proceedings of the first international conference on the simulation of adaptive behavior. Cambridge, MA: MIT Press.

Steels, L. (1994). Beyond objects. In: *Proceedings of the 8th European conference on object-oriented programming.*

Stucky, S.U. (1992). Situated cognition: a strong hypothesis. In F.L. Engel et al. (Eds.), *Cognitive modelling and interactive environments in language learning* (Volume F87 of NATO ASI Series). Berlin: Springer.

Tokoro, M. (1994). *On software agents and their society.* Invited talk at the 1994 modelling autonomous agents in a multi-agent world workshop in Odense, Denmark.

Vandierendonck, A. (1995). A parallel rule activation and rule synthesis model for generalization in category learning. *Psychonomic Bulletin & Review, 2,* 442-459.

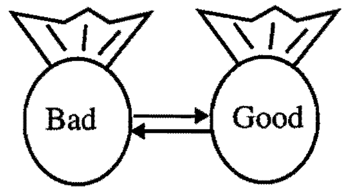
Wavish, P., & Graham, M. (1994). Roles, skills and behaviour. In M. Wooldridge and N. Jennings (Eds.), *Proceedings of the ECAI-94 workshop on agent theories, architectures, and languages.*

Wenger, E. (1987). *Artificial intelligence and tutoring systems.* Los Altos, CA: Morgan Kaufmann

Wood, D., Wood, H., & Middleton, D. (1978). An experimental evaluation of four face to face teaching strategies. *International Journal of Behavioral Development, 1,* 131-147.

Chapter 3

The Practice Agent: Theory



Abstract

Given a set of items that a student has to learn, the main task of the Practice Agent is to determine which item to present at any given time. In the case of paired associates learning the items are pairs, for instance word pairs. The question of how models of human learning processes, particularly models of paired associates learning, can be used in the design of an item sequencing strategy for the Practice Agent will be discussed. This is an illustration of how a model of one agent can be used to design the adaptive behaviour of another agent interacting with it. A new item sequencing strategy will be proposed which is based on functional properties of the models and which uses direct observation. In this "Situated" strategy, items have a higher likelihood of being presented when they have been answered incorrectly the previous time they were presented. The strategy has one parameter, namely the degree to which items are presented more frequently. The models are also used to explore the influence of this parameter and to predict the strategy's effectiveness compared to other strategies. It is predicted that using the Situated strategy will be very advantageous.

3.1 The task of the Practice Agent: item sequencing

As discussed in Chapter 2, given a set of items which the student has to learn the main task of the Practice Agent is to determine at any time which item to present to the student in order to optimize the learning process. For instance, when the student has to learn clock times in Dutch, as is the case in one prototype of the Appeal system, the Practice Agent has to determine whether to present a "full hour", a "half hour", etc. When the student has to learn the translation of a set of words, the Practice Agent has to determine the next word to present to the student each time. This task of determining the next item to be presented to the student each time is called item sequencing.

An item does not have to be characterized by only one dimension. For instance, the Practice Agent may have to determine the multiple parameters of a simulation. We will return to this issue in Chapter 5 on concept learning. Another complication may be that the items in the set are not independent of each other: the learning of one item may be prerequisite to the learning of another item. Dealing with these kinds of issues will be the task of the Navigation Agent, as described in Chapter 6. For the moment, we will restrict ourselves to the case in which there is no inherent structure in the item set, and the item set has a limited size.

As the Practice Agent is meant to be domain-independent, it cannot use knowledge about the individual items in the process of item sequencing, unless this kind of knowledge can be coded by a course designer in a domain-independent way. Suppose a course designer believes that the length of the translation is an important factor in determining the difficulty of learning the translation of a certain word. When the Practice Agent does not even know that the items to be learned are word pairs, the only way these preconceptions of the course designer can be made useful to the Practice Agent is by allowing the course designer to assign relative levels of difficulty to the items corresponding to his or her beliefs. Another way could be to let certain abstract classes of items be distinguishable to the Practice Agent, like paired associates and concepts, and provide the Practice Agent with some knowledge about these kinds of abstract classes. For instance, the agent could know that confusability is an issue in the case of paired associates, and that therefore measures on assumed confusability may be provided by the course designer. So, it is possible, in principle, to make some domain expertise accessible to the domain independent agent. However, this is not easy for either the course designer or the designer of the Practice Agent.

For course designers, it is preferable that the Practice Agent should not need all kinds of prior information on the items, such as their relative difficulty, but should be able to discover for itself which items cause the individual student most difficulty. For instance, domain experts would argue that most native English speaking students would have more trouble when learning Dutch clock times with half hours than with full hours, because the Dutch "half 5" is equivalent to "half past four" in English instead of the expected "half past five". On the other hand, for German students this kind of unequal difficulty is not expected. Now, the main idea is that this kind of domain knowledge does not have to be coded (and hence need not be invented by every course maker in all kinds of domains), but that the Practice Agent should discover for itself when it has to present a certain item, say "half hours", more frequently.

This approach deviates considerably from the intelligent tutoring systems approach: where ITS designers try to achieve adaptivity by incorporating highly detailed domain knowledge (for instance in the form of buggy rules), we try to achieve it by incorporating strategies that are domain-independent as far as possible. We expect that this will considerably reduce the time and cost of designing a course, and that it is at least as effective in producing adaptive behaviour.

3.2 Use of cognitive models

One of the main aims of cognitive science has been to construct and evaluate models of human behaviour controlled by human cognitive processes. Current studies tend to focus on modelling the behaviour of one individual agent in an environment that is experimentally controlled and hence independent of that agent. However, from a situated cognition point of view, the operation of the Practice Agent cannot be decoupled from its interaction with and adaptation to the student. We believe that the modelling of interaction should build on the insights gained so far. The main issue then is how a model of an individual agent's behaviour (e.g., a model of the learning processes of a student) can be used to explain, design, and predict the (adaptive) behaviour of a second agent (e.g., a teacher) interacting with it.

There are numerous models of learning which more or less fulfil the role of the student interacting with the Practice Agent, predicting the answer pattern of a student. The question now is how models of this kind can be used to design item sequencing behaviour in the Practice Agent which is effective in several kinds of learning tasks and does not require much effort from the course designer.

3.2.1 Models of learning processes relevant for item sequencing

There are various models of learning processes, including models of paired associates learning (e.g., Bower, 1961) and concept learning (e.g., Kruschke, 1992; Pearce, 1994). It is beyond the scope of this chapter to discuss all of them in detail. We will briefly describe a few of these to give the reader some idea of what kinds of models there are, and how these can be used to design an item sequencing strategy.

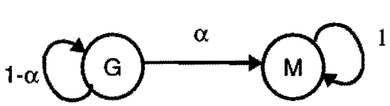
Markov models

Markov models are simple models which are mostly used in cognitive psychology to model the learning of paired associates. A Markov learning model can be characterized by a set of knowledge states, a transition matrix which indicates the probability of a transition from one knowledge state into another, and a response function which maps knowledge states onto probabilities of responses. The most important property of a Markov model is that the knowledge state in which a student is at a certain time t+1 only depends on the knowledge state at time t.

To model the learning of a set of items, the same kind of Markov model can be used for each individual item, with the student's knowledge state depending on the item. To model individual differences between students, different parameter values can be used in the models.To model different item difficulties, different parameter values can also be used. This is illustrated below.

The *All-or-None model* of Bower (1961) is a very simple Markov model with two states, say G (guessing) and M (mastered) (see Figure 5). An item can be either completely mastered, in which case the student will always give the correct response, or it can be completely unknown, in which case the student can only guess the correct response, say with likelihood g. Once an item is mastered, it will always be known, so there is no forgetting. An item has a certain likelihood, say α, to be learned each time it is presented.

$$\begin{array}{c c} & \begin{array}{cc} M_{t+1} & G_{t+1} \end{array} \\ \begin{array}{c} M_t \\ G_t \end{array} & \begin{bmatrix} 1 & 0 \\ \alpha & 1-\alpha \end{bmatrix} \end{array}$$



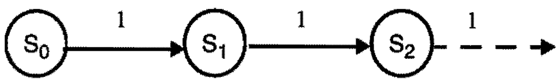
P (Correct response I in state M) = 1
P (Correct response I in state G) = g (guessing probability)

***Figure 5:** Transition matrix, corresponding diagram, and response function for the All-or-none model.*

To model differences between students, a different value of α can be used for each student: the higher the α, the faster the student learns. To model differences between item difficulties, a different value of α can be used for each item: the higher the α, the easier the item. To model differences in the interaction between students and items, different values of α can be used for each combination of student and item: the higher the α, the easier the item is for that particular student.

Markov models can be made more sophisticated by adding more states and associating a parameter with each state transition, or by making the response function more complex.

The assumption that learning may be partial, in the sense that stages may be distinguished in the learning process, would result in the incorporation of more states in the model: one for each stage. An extreme version of this is the *Linear model* (Atkinson, Bower, & Crothers, 1965) which can be viewed as a Markov model with an infinite number of states, and is shown in Figure 6. In this model, the probability of a state transition is 1. The assumption underlying this model is that each presentation of an item reduces the error probability by a constant factor, say α.



P (Correct response I in state S_0)= g (guessing probability)
P (Correct response I in state S_n) = $1-\alpha^n(1-g)$

Figure 6: *Diagram and response function for the Linear model.*

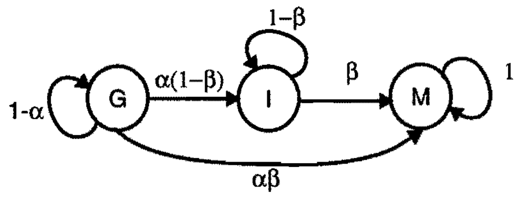
A distinction has often be made between two stages in the learning process, though the interpretation of the meaning of these stages is under discussion. A possible interpretation could be a first stage of storage of the stimulus-response pair in the memory and a second stage of learning to retrieve the response (Humphreys & Greeno, 1970). Another interpretation could be a difference between short-term and long-term memory.

To give an impression of how rapidly these kinds of models become complex, let us suppose we would like to incorporate forgetting in a simple model. A first idea would be to incorporate a state transition in the All-or-None model from M to G, and associate a parameter with it. This parameter would determine the rate of forgetting: the higher the parameter, the more easily an item is forgotten. However, two problems arise. The first problem is that it seems more natural that an item is forgotten when it is not presented to the student than when it is presented. This can be incorporated by having two state transition matrices: one for when the item is presented, and one for when another item is presented. Even this may be too simple: it could be argued that a separate transition matrix is needed for the presentation of every possible other item, because the rate of forgetting an item may depend on the interference between that item and the item presented, and this interference may in its turn depend on the similarity between the two items.

The second problem is that with such a model a student will never remember an item in the long run unless it is presented over and over again. This leads to the idea of incorporating an intermediate state in the model, such that forgetting takes place only from the intermediate state and not from the 'mastered' state. This introduces two extra parameters in the model: one for the extra state transition and one for the response probability in the intermediate state.The resulting model is shown in Figure 7. Though this model is probably still an oversimplification of the human learning process, it is already rather complicated, especially it is realized that every parameter mentioned may depend on the student-item combination.

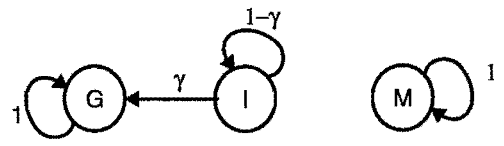
On presentation of the item:

$$\begin{array}{cc} & \begin{array}{ccc} M_{t+1} & I_{t+1} & G_{t+1} \end{array} \\ \begin{array}{c} M_t \\ I_t \\ G_t \end{array} & \begin{bmatrix} 1 & 0 & 0 \\ \beta & 1-\beta & 0 \\ \alpha\beta & \alpha(1-\beta) & 1-\alpha \end{bmatrix} \end{array}$$



On presentation of another item:

$$\begin{array}{cc} & \begin{array}{ccc} M_{t+1} & I_{t+1} & G_{t+1} \end{array} \\ \begin{array}{c} M_t \\ I_t \\ G_t \end{array} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1-\gamma & \gamma \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$



P (Correct response I in state M) = 1
P(Correct response I in state I) = h, h>g
P(Correct response I in state G) = g (guessing probability)

Figure 7: *Transition matrices and corresponding diagrams for a two-stage learning model with forgetting. The top matrix defines the state transitions for a certain item when that item is presented to the student. The bottom matrix defines the state transitions for that item when another item is presented to the student.*

One advantage of using of Markov models is that they are mathematically well understood. However, they are black box models in the sense that item characteristics can only be taken into account indirectly: through parameter values. Interactions in the learning of different items are difficult to incorporate into these models.

Connectionist models

Connectionist models such as ALCOVE (Kruschke, 1992) and Configural Cue (Pearce, 1994) have become very popular, especially for concept learning. A connectionist model consists of a set of interconnected processing units (see Figure 8). Each unit can be in a number of possible states; this is often expressed by saying that the unit can have a certain activation. Typically there is an input layer of units that receive input directly from the environment. The stimulus is encoded on these units. There is an output layer of units that produce output to the environment. The activations of these units usually represent the response probabilities. There may be one or more hidden layers of units that neither receive direct input from nor produce direct output to the environment. These layers can represent abstractions which are useful in producing a correct output pattern. For instance, when the input units encode the letters of a word, the hidden units may encode different lengths of words.

Weights are associated with the connections. These are used as weighting factors in determining how the activation of a certain unit influences the activation of a unit connected to it. The activation a_h of a unit h is determined by the formula

$$a_h = \sum_i a_i w_{ih}$$

where a_i represents the activation of unit i and w_{ih} represents the weight associated with the connection between the units i and h.

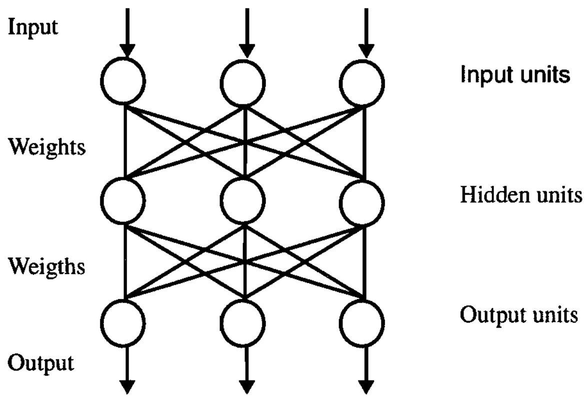


***Figure 8:** Schematic representation of a feed-forward network with one hidden layer*

The activation of a unit may represent a feature of an object (where the object may be a stimulus, response or abstract notion depending on the layer) or the object itself. For instance, in a concept learning task, each exemplar can be represented by a separate input unit, which gets a high activation when the exemplar is presented, while all other input units are not activated. It is also possible to represent an exemplar by the values it has on certain dimensions, and to associate an input unit with each dimension.

A connectionist model generally learns by receiving external feedback on the response pattern produced. This feedback is used to update the weights on the connections, using a certain learning rule.

This is just a very brief, simplified, and incomplete description of connectionist models. For more detailed and complete accounts see Quinlan (1991) and Schalkoff (1992). Some connectionist models for concept learning will be discussed in greater detail in Chapter 5.

An advantage of using connectionist models is that they can give more insight into the learning process, for instance as regards the interaction in learning a set of items. However, item characteristics have to be defined beforehand, and mathematically they are more complex than, for example, Markov models.

More complex models

Over time learning models have tended to become more and more complicated, which means that the number of parameters increases. These parameters are introduced to account for many kinds of cognitive phenomena. For instance, in the distributed associate TODAM model of Murdock (1992) parameters are used to represent types of material (the recognition memory differs for such items as pictures or text), forgetting, ease of encoding the stimulus, ease of encoding the response, attention to the stimulus, attention to the response, attention to the pair, habituation to the stimulus, habituation to the response, habituation to the pair, similarity in stimuli, and similarity in responses.

Likewise, in a more comprehensive theory like SAM (Raaijmakers & Shiffrin, 1981) the number of parameters is so large that the implementation of, say, a paired associates learning model based on this theory, and especially the choice of reasonable parameter values, becomes a rather difficult task.

3.2.2 Ways to use a Model of a Student's Learning Process in the Design of the Practice Agent's Behaviour

In this section, we discuss four ways in which a model of a student's learning process can be used to design the item sequencing behaviour of the Practice Agent. However, this discussion can be generalized to the question of how a model of an agent can be used to design the (adaptive) behaviour of another agent interacting with it. For that purpose, it suffices to see the student as an agent, the model of the student's learning process as a model of the agent's behaviour, the choice of an item as the choice of an action, the presentation of an item as the performance of an action, and the Practice Agent as an agent that has to interact with the first agent.

A first distinction is made between the use of an executable model by the Practice Agent to simulate the student's learning process, and the mere use of functional properties of such a model. A second distinction is whether or not observation is regarded as an input to the process of deciding which item to present next.

Use of a Model by the Practice Agent to Simulate the Student

The Practice agent can use an executable model of the student's learning process to obtain predictions about the internal state and possible actions of the student. In this case, all memory of the Practice Agent on the interaction with the student is indirect and intrinsic to the model.

A. Without on-line observation

The first way to use a model can be seen in Figure 9.

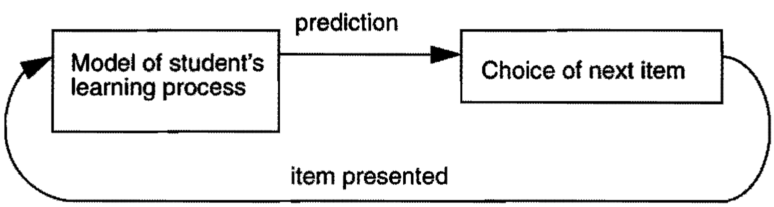


***Figure 9:** Model of the Practice Agent, interacting with the student*

The Practice Agent uses a model of the student's learning process to predict the likelihood that the student does not know the correct response to a certain item. These probabilities (perhaps amongst other factors) constitute the input for the process of selecting the next item. The Practice Agent may, for instance, select the item with the highest likelihood that the student does not know its response yet. The item selected is, in its turn, input to the learning model. Differences in the behaviour of the Practice Agent for different students can be explained by different values for the parameters of the model of the student's learning process. These parameters are determined in an initial phase.

Consider for example the All-or-None model of Bower (1961) discussed above. This model implies that the likelihood that the student does not know the response to a certain item equals $(1-\alpha)^n$, where n denotes the number of times that item and its response have been presented to the student. Parameter α can be estimated in an initial stage as the percentage of correct answers the second time the items are presented.

There are several problems associated with this approach:

1. If the model is incorrect, then the information on which the choice for the next item is based is incorrect, and the choice may turn out to be inappropriate. For instance, as discussed above, the All-or-None model is probably too simple: not all items will be equally difficult (so, there should be an α per item), people tend to forget, there may be differences between short-term and long-term memory. The use of a paired associate learning model based on a more comprehensive theory like SAM (Raaijmakers & Shiffrin, 1981) would result in more parameters. The more parameters the model has and the more complex it is, the more the subsequent problems arise.

2. It is difficult to determine parameters, and it is very likely that errors occur in this estimation process. Reasons for errors are that it is possible that different combinations of parameter values explain the observed behaviour of the student, and that it may be difficult to distinguish between slips and real errors, and between lucky guesses and actual knowledge.

3. It takes a lot of computing power and time both to run the simulation of a model (calculate the probabilities) and to determine the parameters. This makes it difficult to evaluate the possible use of a model by an agent with a computer simulation of the agent. Besides, it is questionable whether human agents would use these kinds of complex computations (even unconsciously).

B. With on-line observation

The second way is similar to the first, but now the reaction of the student is also taken into account and used as input for the model of the student's learning process. This can be seen in Figure 10.

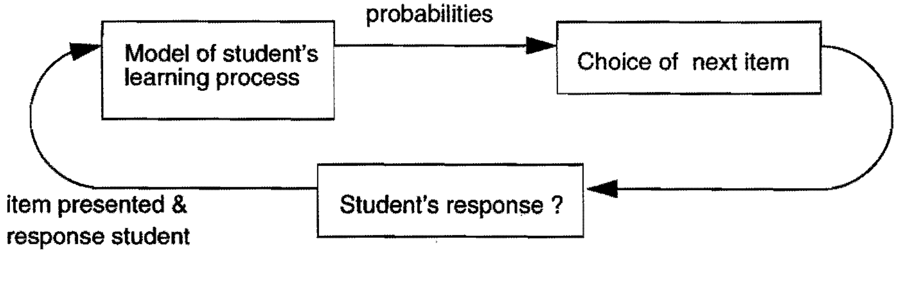


Figure 10: *Model of the Practice Agent, interacting with the student*

So, the Practice Agent can use the correctness of the student's answers to update the model of the student's learning process.

For instance, in the case of the All-or-None model the probability that the student does not know the correct response to a certain item now becomes zero if the student has answered correctly the last time the item was presented, $1-\alpha$ if the student has answered incorrectly, and 1 if the item has not been presented before. In connectionist models the correctness of an answer could be used to update the weights of the connections.

From a situated point of view, this approach is better than the previous one, in that it takes observation into account as well. However, the problems associated with this approach are still somewhat similar to that of the previous approach:

1. The possible incorrectness of the model remains a problem.The observation is only used indirectly through the model.

2. The difficulty of determining parameters now exists not only initially, but also during run time. An advantage is that observation can be used continuously to adjust the parameters, which could reduce the errors.

3. The difficulty of computing power and time consumption remains. An advantage is that calculations can become simpler because of the perception input (like in the All-or-None model as explained above). On the other hand, parameters will now be adjusted during run time, which will involve a large number of calculations.

This approach is the one mostly used in Intelligent Tutoring Systems (Wenger, 1987), and the problems it causes with respect to robustness and efficiency have been discussed in Chapter 1.

Use of the functional properties of a Model

The main difference between this approach and the previous one is that the model is not used to simulate the student's learning process, but only provides some properties of this learning process which are relevant for the item sequencing. Hence, the model does not have to be executable.

A. Without on-line observation

The first way in which the relevant properties of the student's learning process can be used instead of the simulation of a model can be seen in Figure 11.

In the case of the All-or-None model, it can be shown that without observation the optimal approach is to use a Random Recycling strategy, in which an item is only presented again when all the other items have already been presented. This strategy is used in most laboratory experiments (Kintsch, 1977). Random Recycling is also optimal when the Linear learning model (Atkinson, Bower and Crothers, 1965) holds. So, by concentrating on the properties relevant for the interaction, a strategy may be found that is optimal for various models, which diminishes the risk of using an incorrect model. However, the problem of using incorrect properties of course remains. Because the model does not have to be executed, it may not be necessary to determine its parameters. The computation power and time needed are reduced because a simulation is not used. A major drawback of this approach is, however, that no adaptation at all can occur.

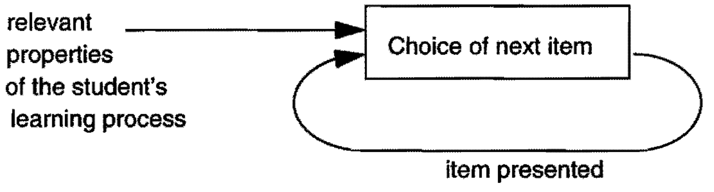


Figure 11: *Model of the Practice Agent, interacting with the student*

B. With on-line perception

This is the only approach which can be combined with the situated point of view that perception should be the most important resource in determining what to do next. Perception is used as an important input for the process of determining the next item, and the properties are also treated as a resource. We will call this approach the *situated method.*

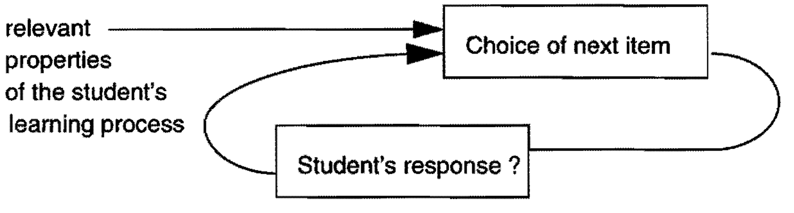


Figure 12: *Model of the Practice Agent, interacting with the student*

In the case of the All-or-None model, a relevant property is that an item that has been responded to correctly will always be responded to correctly. So, a simple strategy for the Practice Agent could be to present only items that have not been responded to correctly before. Perception can be used to see whether the student answers correctly, and a simple memory of items responded to correctly would suffice.

3.3 An item sequencing strategy based on the situated method

We have designed an item sequencing strategy based on perception and simple hypotheses about how a student learns that can be derived from the existing learning models. The first hypothesis is that the likelihood that a student responds incorrectly to an item is greater when he has responded to it incorrectly the last time it was presented than when he has responded to it correctly. So, behaviour in the past predicts behaviour in the future. This hypothesis holds if the All-or-None model is correct, but also if the Linear model or the connectionist models for concept learning like Alcove and Configural cue hold good. For instance, in the All-or-None model the likelihood of responding incorrectly to an item equals 1-α when that item has been responded to incorrectly the last time it was presented, but is smaller when the item has been responded to correctly, because it may already have been in state M.

We use two sets to represent the items to be taught, a 'good' set and a 'bad' set. The good set contains the items correctly responded to the last time they were presented. The bad set contains those responded to incorrectly. Initially all the items to be taught are in the bad set. The first hypothesis implies that the likelihood that a student does not know the correct response to an item is greater when that item is in the bad set than when it is in the good set. So, an item in the bad set should be presented to the student more often, say k times as often (with $k > 1$), as an item in the good set.

When k is very large, an item responded to correctly (i.e., in the good set) is only presented to the student again when the bad set is empty (i.e., all other items have been responded to correctly the last time they were presented). If the All-or-None model held good, i.e., a student would never forget the response to an item that he has responded to correctly once, this would be a very good strategy.

The second hypothesis is that a student tends to forget: the likelihood that a student will respond to an item incorrectly is greater than zero even when he responded correctly the last time it was presented. The value chosen for k was therefore not too large, namely $k=10$, based on an extra assumption that we would like the probability of an item from the bad set being presented to be at least 50%, even when the number of items in the bad set is only 5% of the total number of items. The implications of different choices of the value of k will be discussed below assuming of the correctness of various learning models.

There are a number of possible extensions to this strategy. In the first place, it seems very probable that the likelihood of a student responding incorrectly to an item is greater the first time the item is presented than it will be a second time. This can be incorporated by using three sets instead of two, where the extra set represents the items not yet presented to the student and a high probability is given to the presentation of an item from this set.

In the second place, it seems preferable to restrict the cognitive load of the student, in the sense that the student does not have to study too many items at the same time. This can be achieved by not putting all the items in the bad set initially, but adding new items to the bad set whenever the number of items in the bad set falls below a certain minimum.

In the third place, different sets can be used to represent different numbers of subsequent correct responses to an item. In that way, it is possible to present an item that has been responded to correctly, say, twice with a higher probability than an item that has been responded to correctly, say, five times.

Though each of these extensions is interesting and may improve the effectiveness of the strategy to a large extent, we have chosen to concentrate on the simplest version. This reduces the number of parameters to only one, and gives an opportunity to get a feeling for the different issues involved in the learning of paired associates and concepts. We would like to have empirical evidence for every adjustment we make to this simple strategy, and a feeling for which adjustments are most necessary.

3.4 How to define the effectiveness of a strategy

The effectiveness of a task sequencing strategy is often measured as the average number of correct responses to a test after a certain amount of practice trials. There are, however, several disadvantages to this approach.

In the first place, it is hard to determine after how many practice trials the test should take place. When the number of practice trials is too large, a difference in strategy effectiveness may disappear because eventually all the students learn the correct responses, regardless of the strategy used. On the other hand, if the number of practice trials is too small, the effect of the strategies on the last phase of the learning process is neglected. So, it seems preferable to measure the number of practice trials needed to learn the set, instead of measuring the number of correct responses after an arbitrary number of practice trials. However, this is hard to do in an experimentally controlled way.

In the second place, even when assuming that the test takes place at the right moment in time, there is still the problem that the only aspect measured is which strategy produces the best asymptotic behaviour, i.e., with which strategy are the most difficult items learned first. Another aspect which it seems reasonable to measure is the time required to learn *most* items of the set. Of course, this poses the problem of what criterion to use for defining "most".

In the third place, the effectiveness of a strategy may depend on the phase in the learning process. It may well be that one strategy is more effective when the student still does not know any of the items, while another strategy is more effective after a certain number of items have been learned. These kinds of effects cannot be observed with this approach.

Together, these three problems suggest that it is preferable to measure learning curves instead of just one test score. These curves can be obtained by alternating practice and test phases, and will give an insight into the complete learning process, instead of concentrating on just one particular phase.

Finally, a disadvantage of the approach is the use of averages over students. The effectiveness of a strategy may depend on the type of students. It may well be that one strategy is more effective for high-ability students, while another strategy is more effective for low-ability students. In an experiment involving a learning task, the variance between students is often large. So, instead of using an average learning curve it may be preferable to take the whole collection of learning curves into consideration in order to get an impression of what is really going on.

This implies that we do not have to define the effectiveness of a strategy in detail, but that we can use different definitions concurrently and obtain a more accurate insight into the effect of different strategies.

3.5 Predictions of the models

Models of the learning process can also be used to obtain predictions of the relative effect of different strategies, or the effect of different values of parameters in the strategy. In a way, the use of models for predictions also gives an indication of whether the relevant properties of the model are captured in the strategy: if they are, the model should predict a benefit of using the strategy.

3.5.1 How to obtain predictions of learning curves

There are two ways in which predictions can be obtained. The first is to calculate the estimated learning curves of a model, given parameter values for the model and a certain item sequencing strategy. In principle, this can be done with all Markov models, and particularly with the simplest models like the All-or-None model. Of course, the more complex the model becomes, the more difficult the mathematics involved. The same applies to the item sequencing strategy used. For instance, it is easier to calculate predictions with Random Recycling than with the Situated strategy, because Random Recycling does not use the response of the model as an input for the decision as to which item to present next.

The second way does not involve mathematics and is shown in Figure 13. The model is simply used as a substitute for a real subject in an experiment. For that purpose, the model has to be executable. A response generator is used to transform response probabilities generated by the model into responses. Our response generator produces the various responses with the same probabilities as those given by the model. Another method could be always to give the response with the highest probability.

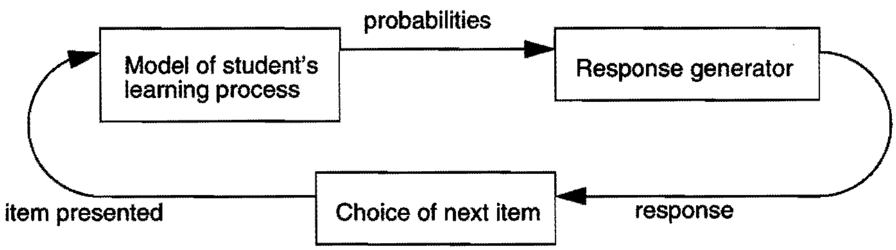


Figure 13: *Schematic representation of a way in which predictions of a model can be obtained.*

We have decided to use the second way, even when a mathematical calculation could be made. This enables us to use the same approach for all models (including the connectionist models) and all strategies. Moreover, in this way exactly the same condition is used for the model as for the real subjects, including all possible problems with random number generators. For every parameter setting of the model, the average is taken over 100 runs. The variation of the parameter values has been chosen such that the corresponding learning curves represent the spectrum of likely student performance. For that purpose, more simulations were done than will be reported.

The predictions will be given for a task in which 30 items have to be learned. There is no internal structure in the item set. Examples of these kinds of tasks will be given in the experiments below. An elementary mathematical analysis shows that the number of items in the set has no influence on the effectivity of both the Random Recycling strategy and the Situated strategy as long as the models do not use this kind of information either.

For the reasons described in Section 3.4, we have opted for an iteration of test and practice phases. The same procedure will be used in the experiments. Feedback regarding the correctness of the answer is only given in the practice phases, i.e., it is only in these phases that the model can learn. On the other hand, forgetting can take place in both practice and test phases. For the Situated strategy only the responses of the student (in this case the model) in the practice phases are used. For the predictions, the same number of test and practice phases were used as will be used in the experiments, namely eight test phases and, hence, seven practice phases. This number was determined on the basis of a pilot study.

3.5.2 Predictions about the effect of the strategy's parameter

As mentioned in Section 3.3, the value of the parameter k of the Situated strategy was chosen equal to 10. We stated as arguments for this choice that k should be large in order to profit from the information regarding past behaviour of the student, but not too large to be able to handle forgetting. The exact value of k was chosen on the basis of a seemingly arbitrary assumption about a preferred probability with which items from the bad set should still be presented to the student when almost all items have been learned.

In order to gain more insight into the effect various values of k would have, simulations of several models have been performed with different values of the parameter. There are many models of learning processes, as has been discussed in Section 3.2.1. In this section, we will restrict ourselves to only three of them.

First, the All-or-None model has been used, because it is well-known, relatively simple, and captures the idea that past behaviour predicts future behaviour in a simple, though perhaps trivial, way. Second, the Linear model has been used, because the Random Recycling strategy is based on this model and is optimal according to this model. Therefore, it is interesting to see what this model predicts as regards the effectivity of the Situated strategy. Third, a two-stage learning model with forgetting (as discussed in Section 3.2.1, see Figure 7) has been used in order to investigate the effect that forgetting has on the optimal value of the parameter.

Predictions of the All-or-None model about the strategy's parameter
The All-or-None model (see Figure 5) has two parameters: the guessing probability g in the initial guessing state, and the transition probability α from the guessing state to the mastered state. In the case of a recall task like that in Experiment 1 of Chapter 4, the likelihood of a correct guess may be neglected. So, we assume that g equals 0: the student always answers incorrectly when in the guessing state. The value of parameter α was varied from 0.1 to 0.5.

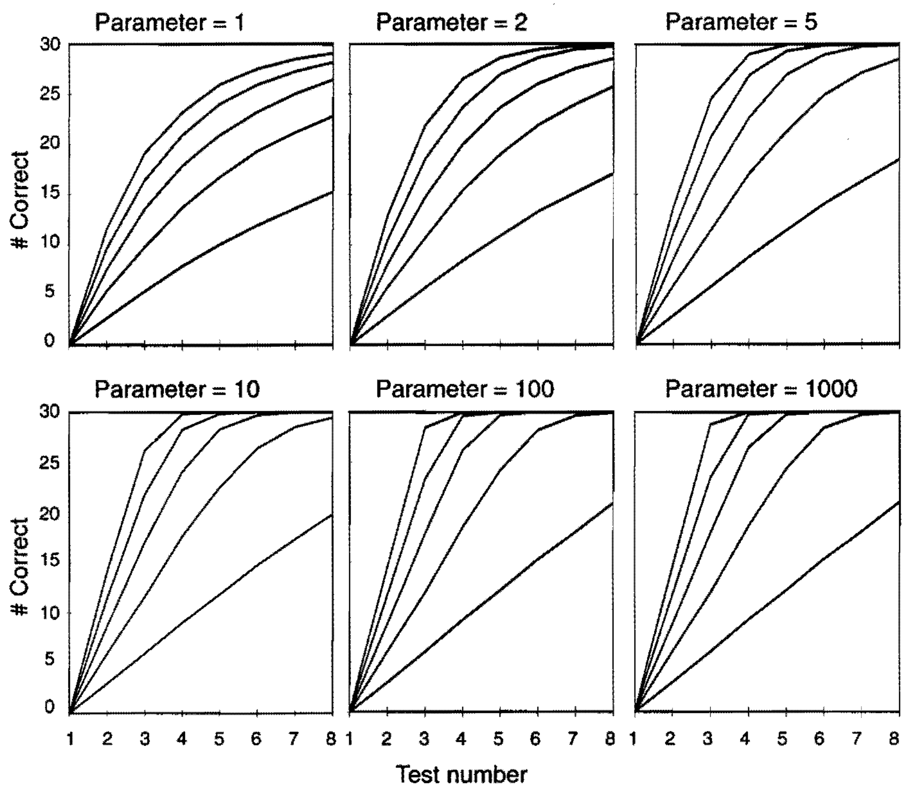


Figure 14: *Predictions of the All-or-None model for the Situated strategy for k equal to 1, 2, 5, 10, 100, and 1000. Each line in the graphs represents the average over 100 runs of the model with parameter α (from bottom to top) 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

The results of the simulations of students by the All-or-None model for several values of parameter k of the strategy can be seen in Figure 14. As expected, the All-or-None model predicts that the strategy will be more effective as the value of k becomes larger. The difference in the effectivity of parameter values 1, 2, and 10 is quite obvious. However, the difference between parameter values of 10, 100, and 1000 is not that large. The difference in effect seems to be located mostly at the end of the learning curves: an increase in parameter value makes the curves more like straight lines. This is logical, because the lower the parameter's value the sooner items which have been answered correctly are likely to be presented to the student again. In the All-or-None model, items will never be forgotten, so the presentation of a correctly answered item only obstructs the learning process.

With a parameter value of 10, the curves already approximate straight lines till the last few items have to be learned. Therefore, according to this model a choice of 10 as the value for the parameter seems reasonable, even though higher values of k have to be preferred.

Predictions of the Linear model about the strategy's parameter
The Linear model (see Figure 6) has two parameters: a guessing probability g in the initial state, and a probability α with which the error probability is reduced each time an item is presented. As in the case of the All-or-None model we assume that g equals 0. The value of parameter α has been varied from 0.6 to 0.9.

The results of the simulations of students by the Linear model for three values of parameter k can be seen in Figure 15.

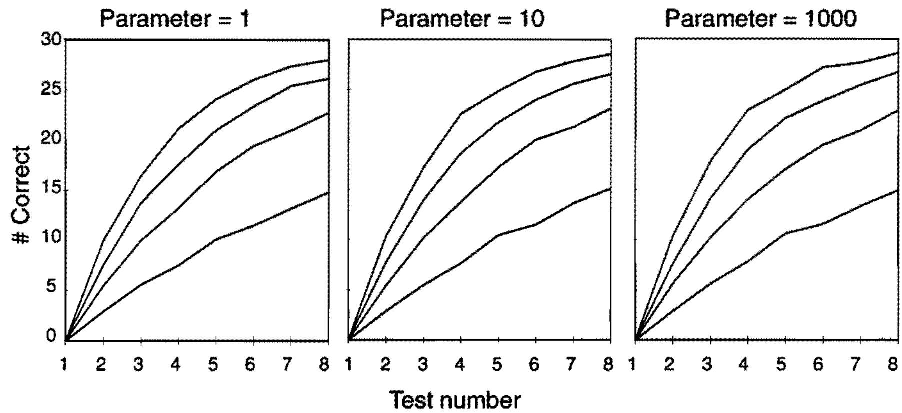


Figure 15: *Predictions of the Linear model for the Situated strategy for k equal to 1, 10, and 1000. Each line in the graphs represents the average over 100 runs of the model with parameter α (from top to bottom) 0.6, 0.7, 0.8, and 0.9, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

The value of k seems to have no influence on the effectivity of the strategy. This is not surprising. In this model, all items are assumed to be equally difficult and the error probability is exactly determined by the number of times the item has been presented. Therefore, the optimal item selection strategy is to present all items equally often, which is exactly what Random Recycling does. This is approximated with a parameter value of 1, though not as perfectly as with Random Recycling, because there will always be a probability that some items are presented more often than others. Increasing the value of k implies that items that have been answered incorrectly will have a higher probability of being presented again. However, these items are also likely to have a higher error probability, and hence, are likely to have been presented less often in the past. So, even with higher values of the parameter k, the strategy still approximates one in which all items are presented equally often.

Hence, the value of the parameter k is not important according to this model. However, this is likely to change when not all items are assumed to be equally difficult. Our hypothesis would then be that a higher value of the parameter results in the more difficult items being presented more frequently, thereby increasing the learning effectivity. We have therefore used the Linear model for a set of items of varying difficulties. We have assumed that parameter α of the model (representing the decrease in error probability) is the multiplication of a parameter β which represents a student's learning capability (the higher the value of β the slower the student learns) and a parameter χ which represents an item's difficulty (the higher the value of χ the more difficult the item). Parameter β has been varied between 0.06 and 0.09. Parameter χ has been kept at 1 for 10 of the 30 items, and at 10 for the other 20 items. Hence, one third of the items are 10 times as easy to learn as the other items. With these values of the parameters, the same range of values of α is reached for two thirds of the items as in the simulations with equally difficult items.

The results of the simulations are shown in Figure 16.

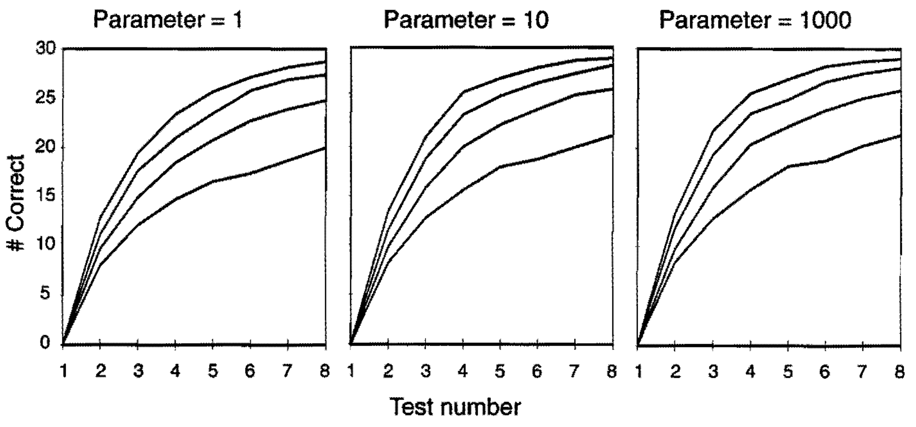


Figure 16: *Predictions of the Linear model in the case of varying item difficulties for the Situated strategy for k equal to 1, 10, and 1000. Each line in the graphs represents the average over 100 runs of the model with parameter* β *(from top to bottom) 0.6, 0.7, 0.8, and 0.9, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

Though the graphs are still quite similar, there is an influence of k's value. The curves in the case of k=10 end higher than when k=1. The lower the curves –i.e, the weaker the student– the clearer the effect becomes. There is, however, no noticable difference between values of 10 and 1000. This suggests that if the items are not equally difficult a somewhat higher value of k should be preferred, like 10, but that there is no reason to use a very high value.

Predictions of a two-stage model about the strategy's parameter

The predictions discussed so far have only resulted in arguments for the use of a high value of the parameter k. However, we have restricted the value of k to 10, because we wanted to take forgetting into account. When k is very high, a correctly answered item is only presented again when all items have been answered correctly. There are several drawbacks to this approach.

In the first place, an item may be forgotten. In itself this is not a very strong argument for restricting the value of k, because the item will be presented automatically again when all items have been answered correctly. However, the argument becomes stronger when the items are not equally difficult. When an easy item is forgotten, it may be more effective to present that item again than it would be to present a very difficult item.

In the second place, an item may be answered correctly, and hence become part of the good set, without really being mastered by the student. This may happen because the student always has a certain guessing probability of choosing the correct answer (for instance, when choosing between two categories), or just because learning is partial: there are states in which the likelihood of a correct answer is smaller than one but larger than zero. The same argument also applies in this case: the value of k should be restricted, especially when all items are not equally difficult.

In order to gain more insight into this and present some clear evidence that the value of k should not become very large, simulations were performed with a model incorporating both aspects of forgetting and guessing. This is the two-stage learning model as shown in Figure 7.

This model has five parameters. As before, we have chosen the initial guessing probability g equal to zero. We have assumed that parameter α of the model (representing the transition probability from the initial to the intermediate state) is the multiplication of a parameter δ which represents a student's capability for learning in the first phase (the higher the value of δ, the faster the student learns) and a parameter ε which represents an item's difficulty (the higher the value of ε, the easier the item). Parameter δ has been kept at 0.1. Parameter ε has been kept at 10 for 10 of the 30 items and at 1 for the other 20 items. Hence, as before in the simulation by the Linear model, one third of the items are 10 times as easy to learn as the other items.

Parameter β (representing the transition probability from the intermediate to the mastered state) has been varied between 0.3 and 0.9.

The forgetting parameter γ (representing the transition probability from the intermediate state to the initial state if another item is presented) has been fixed at 0.3. This choice has been inspired by the fact that forgetting should take place and that, hence, the parameter should not be to close to zero, but that on the other hand forgetting should not prevent the student from learning and that, hence, the parameter should not be too large.

Intermediate state parameter h (representing the probability of a correct response in the intermediate state) has been fixed on 0.9. In this way, taking into account the relatively high forgetting probability γ, the intermediate state can be interpreted as short term memory.

The results of the simulations are shown in Figure 17.

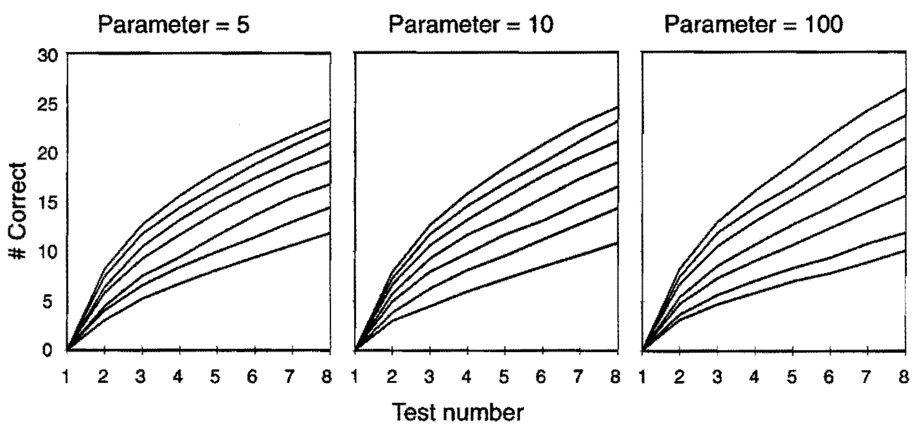


Figure 17: *Predictions of a two-stage learning model in the case of varying item difficulties for the Situated strategy for k equal to 5, 10, and 100. Each line in the graphs represents the average over 100 runs of the model with parameter β (from bottom to top) 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

Two effects can be observed. In the first place, for the higher values of β, i.e., for the faster students, effectivity is increased by an increase in the value of the strategy's parameter. In the second place, for the lower values of β, i.e., for the weaker students, effectivity is decreased by an increase in the value of the parameter.

An explanation for these results is that for a high value of β the probability that an item is added to the good set, though the student has not yet mastered it, is relatively low: an item answered correctly because it is in the intermediate state has a high likelihood of passing on to the mastered state by exactly that presentation on which it is correctly answered.

The conclusion from these results is that the value of the parameter should be restricted, especially if we are interested in the effectivity of the strategy for the weaker students. The choice of k equal to 10 seems reasonable.

3.5.3 Predictions about the relative effect of the strategy

We have chosen to contrast our strategy with two existing item sequencing strategies, namely Random Recycling (Kintsch, 1977) and the so-called Variable Interval Performance (VIP) queuing (Alessi & Trollip, 1985; DeKlerk & VanBussel, 1990). As discussed above, in Random Recycling an item is only presented again when all other items have been presented.

In VIP queuing, all the items to be learned are ordered in a queue. Each time, the first item from the queue is presented to the student and removed from the queue. When the student answers correctly, the item is placed at the end of the queue. When the student answers incorrectly, the item is inserted at various places in the queue. In that way, it will presented again earlier than items that have been answered correctly. The decision as to how often and where to insert an incorrectly answered item in the queue is rather arbitrary, but can have a major impact on the performance of the strategy. We have decided to use the same settings as in the study of De Klerk and Van Bussel (1990). Hence, an incorrectly answered item is inserted at three places in the queue, namely at the third, tenth, and last position.

Obviously, there are more item sequencing strategies than these two (for a more complete overview see Ellerman, 1991). Random Recycling was chosen because it can serve as a kind of baseline: it is a very simple strategy which is used in most laboratory experiments and also quite often used in educational software. VIP queuing was chosen because it has been designed as a relatively simple strategy that adapts to the student's performance.

Predictions have been obtained regarding the effect of the Situated strategy relative to the other strategies. The same kind of models as in the last section were used. In all the simulations, the Situated strategy was applied with parameter k=10.

Predictions of the All-or-None model about the relative effect of the strategy

The same settings were used as in the previous section. The results of the simulations of students by the All-or-None model are shown in Figure 18.

According to the model, there is a clear advantage of using the Situated strategy compared to the other strategies: all the learning curves are higher in the case of the Situated strategy. Returning to the issue of the parameter setting for the Situated strategy (as discussed in the previous section), the Situated strategy already outperforms Random Recycling with a parameter value of 2.

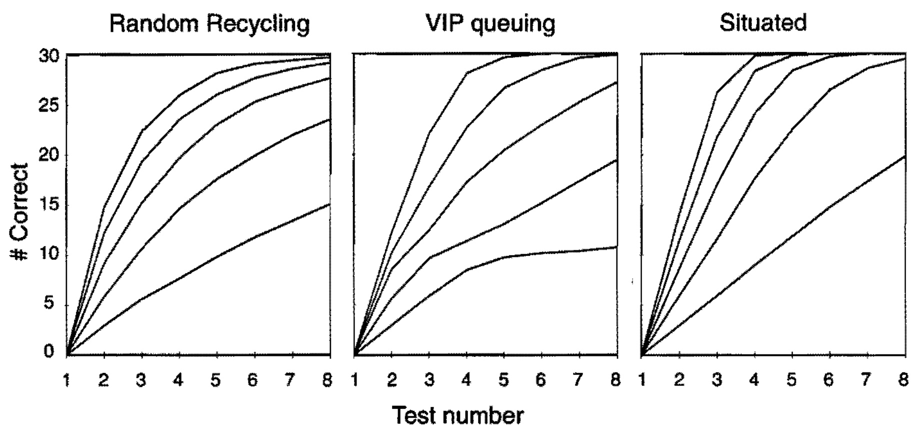


Figure 18: *Predictions of the All-or-None model for Random Recycling, VIP queuing, and Situated, respectively. Each line in the graphs represents the average over 100 runs of the model with parameter* α *(from bottom to top) 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

According to this model, VIP queuing is more effective than Random Recycling for very good students: the increase in the two highest learning curves for VIP queuing lasts longer than for Random Recycling and therefore the curves end higher. However, VIP queuing performs rather poorly for weak students: the two lowest learning curves for VIP queuing end lower than those for Random Recycling and the lowest learning curve especially hardly increases at all in the end. A possible explanation for this is a kind of blockage effect. Weak students are likely to answer an item incorrectly a couple of times before answering correctly. Each time an item is answered incorrectly it will appear twice at the beginning of the queue (in third and tenth position). So, it is possible that by the time the item is answered correctly a lot of appearances of that item are still in the front part of the queue, thereby blocking the learning of still unmastered items. These extra presentations of the item do not make any sense in the All-or-None model, because the item will never be forgotten once it has been answered correctly.

It seems likely that the blockage effect becomes even worse when not all items are equally difficult: more difficult items may block the learning of easier items. It also seems likely that the advantage of using the Situated strategy becomes even larger when not all items are equally difficult: then there is a clear reason why some items have to be presented more frequently.The advantage of the Situated strategy in the case of equally difficult items is merely due to the probabilistic nature of learning in the All-or-None model.

To check the correctness of these hypotheses, simulations were done with a set of items of varying difficulties. We have assumed that parameter α of the model (representing the transition probability from the initial to the mastered state) is the multiplication of a parameter β which represents a student's learning capability (the higher the value of β, the faster the student learns) and a parameter γ which represents an item's difficulty (the higher the value of γ, the easier the item). Parameter β has been varied between 0.01 and 0.1. Parameter γ has been kept at 10 for 10 of the 30 items, and at 1 for the other 20 items. Hence, as before, one third of the items are 10 times as easy to learn as the other items.

The results of the simulations are shown in Figure 19.

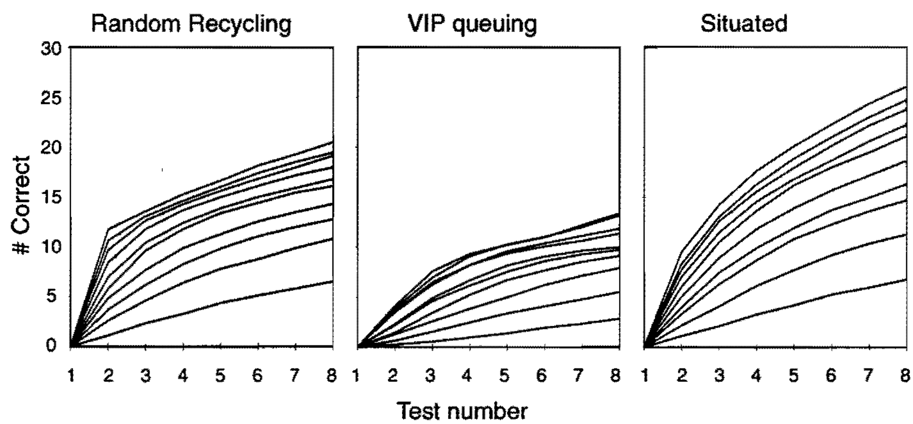


Figure 19: *Predictions of the All-or-None model in the case of varying item difficulties for Random Recycling, VIP queuing, and Situated, respectively. Each line in the graphs represents the average over 100 runs of the model with parameter β (from bottom to top) from 0.01 to 0.1. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

Two observations can be made. In the first place, our hypothesis regarding the blocking effect seems correct. All the learning curves in the case of VIP queuing are very low, approximately as low as, or even lower than, the lowest learning curve in Figure 18. This is not the case with the other strategies.

In the second place, the Situated strategy is more effective than Random Recycling, especially for the higher learning curves. However, the effect is not as great as we might have expected. A possible reason for this can be seen in the graphs. It can be observed that Random Recycling scores better in the beginning, at the second test phase. The learning curves start steeper. This can be explained by the fact that in Random Recycling all items, including all easy items, are presented to the student in the first practice phase. Hence, the student has an early opportunity to learn these easy items. On the other hand, in the Situated strategy it is possible that some of the easy items have not yet been presented in the first practice phase.

From these results, it can be concluded that Random Recycling should be used initially, so that all items are presented once, before using the Situated strategy. However, we have decided not to do this yet, as the models are not necessarily correct, but to look first at the results of experiments with real users. The results of the experiments (see next chapter) indeed contradicted this prediction.

Predictions of the Linear model about the relative effect of the strategy
The same settings have been used as in the previous section, with a set of equally difficult items. The results of the simulations of students by the Linear model are shown in Figure 20.

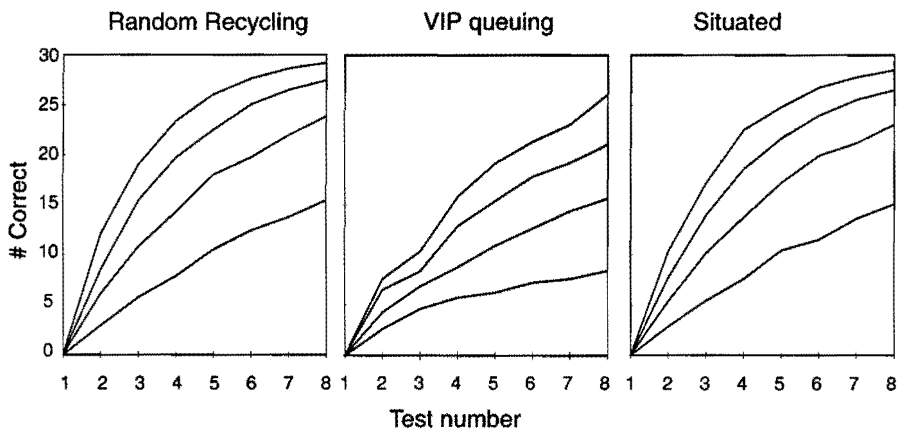


Figure 20: *Predictions of the Linear model for Random Recycling, VIP queuing, and Situated, respectively. Each line in the graphs represents the average over 100 runs of the model with parameter α (from top to bottom) 0.6, 0.7, 0.8, and 0.9, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

As mentioned above, Random Recycling is the optimal strategy when the Linear model is correct. Indeed, the simulations show an advantage of using Random Recycling. However, the results for the Situated strategy approximate the results of Random Recycling: the difference is rather small. VIP queuing performs worse than Situated, even for the best students. The blocking effect still seems to be present.

Predictions of a two-stage learning model about the relative effect of the strategy

The same settings have been used as in the previous section. The results of the simulations of students by the two-stage learning model are shown in Figure 21.

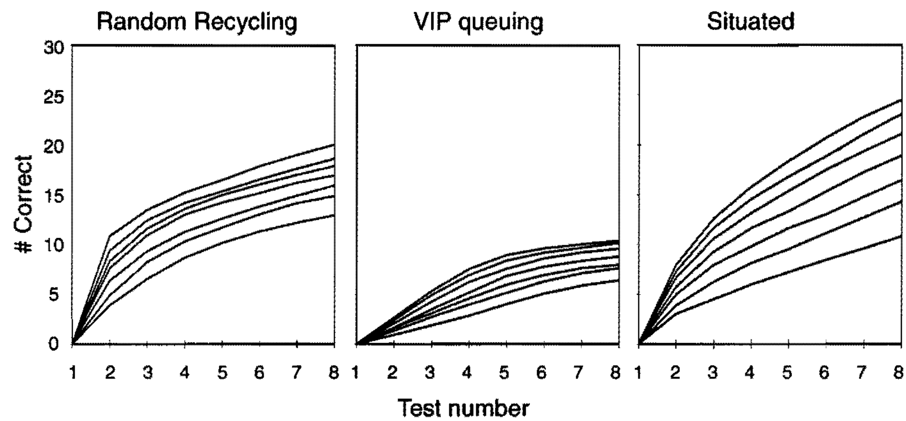


Figure 21: *Predictions of a two-stage learning model in the case of varying item difficulties for Random Recycling, VIP queuing, and Situated, respectively. Each line in the graphs represents the average over 100 runs of the model with parameter* β *(from bottom to top) 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9, respectively. The test phases are given on the x-axis, and the number of correct responses on the y-axis.*

Three observations can be made regarding these results. In the first place, the results for VIP queuing are again very bad, probably due to the blocking effect. In the second place, the learning curves for Random Recycling again start steeper than those of Situated: it seems preferable to use Random Recycling initially. In the third place, the Situated strategy is clearly more effective for the faster students. However, for the weakest students Random Recycling is more effective. An explanation is that the Situated strategy can only be effective when the student has learned a certain number of items: only the fact that those items can be presented less frequently produces a positive effect.

3.6 Conclusions

A question addressed in this chapter is how models of an individual agent, say a student, can be used to design the adaptive behaviour of another agent, say the teacher, interacting with it. It has been argued that instead of using an executable model to simulate the student, only the functional aspects of the model should be used, in combination with direct observation.

According to this approach, a 'Situated' item sequencing strategy has been designed. In this strategy, items have a higher likelihood of being presented when they were answered incorrectly at their previous presentation than when they were answered correctly. The strategy has one parameter, namely the degree to which items are presented more frequently, say *k*.

Executable models of the human learning process have been used to obtain predictions regarding the effect of different values of *k*, and regarding the relative effect of the Situated strategy compared to other item sequencing strategies, namely Random Recycling (Kintsch, 1977) and VIP queuing (Alessi and Trollip, 1985). It has been argued that the effectiveness of a strategy should be determined on the basis of learning curves rather than asymptotic performance, and that the learning curves of individual students should be considered rather than averages over students. In the next chapter, the validity of the models will be tested by comparing their predictions with experimental data.

As far as the effect of the value of *k* is concerned, both the All-or-None model (Bower, 1961) and the Linear model (Atkinson, Bower, & Crothers, 1965) predicted that the value of *k* should be as large as possible. However, the difference in the learning curves of parameter values of ten and larger was very limited. On the other hand, a two-stage learning model, which incorporated forgetting and guessing, predicted that a higher value of *k* would have a negative effect on the learning curves of the low performers. Therefore, the value of *k* should be restricted. Based on these observations, we have opted for a value of k=10, and this value will be used in the experiments in the next chapter.

As far as the relative effect of the Situated strategy is concerned, the All-or-None model predicted that the Situated strategy is much more effective than the other two. VIP queuing was predicted to outperform Random Recycling for the very high performers, but to be much worse for the lower performers. In the case of items of varying difficulty, the Situated strategy was predicted to be more effective than Random Recycling, but with a rather limited difference. This was due to a relatively steep initial rise in the learning curves of Random Recycling, which suggests that it may be better to use Random Recycling at the very beginning of the learning process followed by the Situated strategy. The same effect was predicted by the two-stage learning model. This model also predicted that the Situated strategy would be better than Random Recycling for the high performers, but worse for the low performers. The Linear model, on which Random Recycling is based, predicted an advantage of using Random Recycling, but the difference with the Situated strategy was predicted to be very limited.

In the next two chapters, the effectivity of the Situated item sequencing strategy will be evaluated experimentally in both a paired associate learning task (see Chapter 4) and a concept learning task (see Chapter 5). The validity of the models will also be tested.

References

Alessi, S.M., & Trollip, S.R. (1985). *Computer-based instruction*. Englewood Cliffs (NJ): Prentice Hall.

Atkinson, R.C., Bower, G.H., & Crothers, E.J. (1965). *An introduction to mathematical learning theory.* New York: John Wiley and Sons

Bower, G.H. (1961). Application of a model to paired-associate learning. *Psychometrika, 26*, 255-280.

DeKlerk, L.F.W., & VanBussel, F. (1990). Learning and instruction with ELLA: a computer-based program. In J.M. Pieters, P.R.J. Simons, and L. de Leeuw (Eds.), *Research on computer-based instruction*. Amsterdam/Lisse: Swets and Zeitlinger.

Ellerman, H. (1991). *MIR, a monitor for initial reading*. Ph.D. thesis, Eindhoven University of Technology, The Netherlands.

Humphreys, M., & Greeno, J.G. (1970). Interpretation of the two-stage analysis of paired-associate memorizing. *Journal of Mathematical Psychology, 7*, 275-292.

Kintsch, W. (1977). *Memory and cognition*. New York: John Wiley and Sons.

Kruschke, J.K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review, 99*, 22-44.

Murdock, B.B. (1992). Item and associative information in a distributed memory model. *Journal of mathematical psychology, 36*, 68-99.

Pearce, J.M. (1994). Similarity and discrimination: A selective review and a connectionist model. *Psychological Review, 101*, 587-607.

Quinlan, P. (1991). *Connectionism and psychology: A psychological perspective on connectionist research.* London: Harvester Wheatsheaf.

Raaijmakers, J.G.W., & Shiffrin, R.M. (1981). Search of associative memory. *Psychological Review, 88*, 93-134.

Schalkoff, R.J. (1992). *Pattern recognition: statistical, structural and neural approaches*. New York: John Wiley.

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems*. Los Altos, California: Morgan Kaufmann

Chapter 4

思

The Practice Agent in a paired associates learning task

Abstract

In both recall and recognition tasks, it has been empirically tested whether the models' predictions about a positive effect of using the Situated item sequencing strategy were valid. The results of these experiments have implications both for the behaviour of the Practice Agent and the models. In a recall task, the Situated strategy reduced the variance between subjects, improving the results of the poor performers. This was due to faster learning of the more difficult items, as they were presented more frequently, without obstructing the learning of the easier items. The models turned out to be invalid, in the sense that they predicted a too sharp initial rise of the learning curves, and largely underestimated the effect of VIP queuing which suggests that they lack a mechanism simulating the limited cognitive load a student can handle. Successive experiments, in both a recognition and a recall task, showed that the Situated strategy already had an advantage with only a limited opportunity to adapt to the student. However, later experiments failed to show that using the Situated strategy had any effect. This was probably due to a too high performance on the part of the subjects. Besides, a change of the strategy seems necessary in order to prevent incorrect transfers to the good set as a consequence of short-term memory.

4.1 Experiment 1: Paired associates learning in recall

Our hypothesis is that the use of the Situated strategy is superior, and the predictions of the models in the previous chapter have provided some support for this. In this section we provide some experimental data to support this claim. Some researchers from the field of situated cognition have argued that the traditional, well-controlled laboratory experiments are not valid, because the interaction with a complex changing environment is purposely removed (Lave, 1988). We believe, however, that it is possible to use well-controlled experiments to provide experimental support for the situated approach. In this section, we will give an example: we will show that the Situated item sequencing strategy gives better results than Random Recycling because of its adaptivity.

4.1.1 Method

Design and Procedure
Comparisons were made between groups of subjects learning Japanese translations of Dutch words in different experimental conditions. In each condition, the subjects were presented with alternate test and practice phases. In both phases, Dutch words were presented on the computer screen (see Figure 22) with the request to type as accurately and quickly as possible the translation into Japanese (words written as pronounced). Answers and response times were recorded. In test phases, all 30 words were presented to the subjects in a random order. No feedback was given with respect to the correctness of the translation. In practice phases, feedback was given with respect to the correctness of the translation, and the correct translation was presented on the screen for 2 seconds. Depending on the condition assigned to the subject, in the practice phases Random Recycling, VIP queuing, or Situated was used to determine the sequence of 30 words presented to the subjects. So, depending on the condition, it could occur that in a practice phase the same word was presented more than once, and other words not at all.The experiment ended after eight test phases.

Subjects
Twenty-seven subjects with university or higher vocational training participated voluntarily in the experiment.The average age was 23. No subject had any prior experience with Japanese. Subjects were randomly assigned to one of the three experimental conditions.

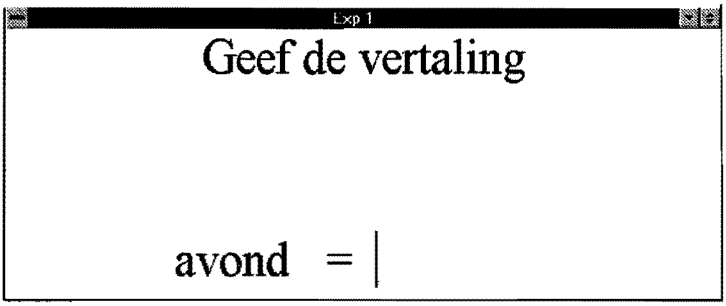


***Figure 22:** Screen layout of the recall Experiment 1, 2b, and 3b.*

Equipment

The material was presented in black and white on the screen of a Sun Sparc station. The subjects used an ordinary keyboard to type in the responses, use of the mouse was not needed. No use was made of audio. All instructions and feedback were given in a text format on the screen.

Materials

Word pairs were selected from the first lessons of a beginners' course in Japanese. The word pairs are shown in Table 2.

***Table 2:** Word pairs used in Experiment 1. English translations are given in parentheses.*

bank	benchi	(bench)	koud	samui	(cold)
raam	mado	(window)	omdat	kara	(because)
avond	yoru	(evening)	hand	te	(hand)
acht	hachi	(eight)	kleur	iro	(colour)
deze	kore	(this)	zee	umi	(sea)
huis	ie	(house)	geur	nioi	(scent)
neus	hana	(nose)	groot	okii	(large)
werk	baito	(work)	regen	ame	(rain)
film	eiga	(movie)	boom	ki	(tree)
goed	ii	(good)	maan	tsuki	(moon)
rood	aka	(red)	moed	gattsu	(courage)
daar	soko	(there)	bier	nama	(beer)
hoed	boshi	(hat)	boek	hon	(book)
riem	beruto	(belt)	ander	hoka	(other)
vijf	go	(five)	zoon	musuko	(son)

No effort was made to construct a set of equally difficult word pairs. On the contrary, on the basis of the above predictions we had reasons to prefer a set of items of varying difficulty. However, we did not construct such a set in an artificial manner: if the items vary in difficulty this is a natural result of the words present in the first lessons of a course.

The number of letters in the stimuli varied between three and five. The number of letters in the response words varied between two and six. Twenty of the forty words were nouns, three were adjectives, and seven were of another kind of word.

4.1.2 Results: general

The results of the experiment are shown in Figure 23. Average results are shown in Figure 24. A MANOVA was performed on logit-transformed proportions of correct responses, with the test phase as a within-subjects repeated measures factor and the experimental condition (strategy used in the practice phases) as a between-subjects factor. The results of the analysis are summarized in Table 3.

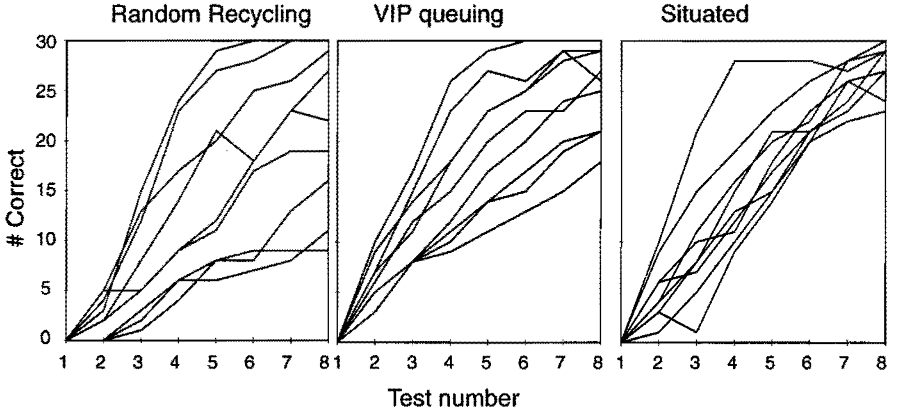


Figure 23: Results of Experiment 1 for Random Recycling, VIP queuing, and Situated, respectively. Each line in the graphs represents a subject. The test phases are given on the x-axis and the number of correct responses on the y-axis.

There was a main effect of test phase [$F(7,15)=33.2$, $p < .001$], but no main effect of strategy. There was neither a significant effect for Random Recycling versus Situated or VIP queuing nor for Situated versus VIP queuing.

A significant interaction was found between test phase and strategy [$F(14,30)=2.5$, $p < .05$]. In particular, a significant interaction was found between test phase and Random Recycling versus Situated or VIP queuing [$F(7,15)=5.72$, $p < .01$]. However, the interaction between test phase and Situated versus VIP queuing was not significant. Testing the effect of strategy (and the contrasts) per test phase revealed that it was only significant in the second test phase.

Table 3: *Results of the MANOVA on the data of Experiment 1.*

Source	Num DF	Den DF	F	
Between subjects				
Strategy	2	1	2.60	
Random Recycling vs. {Situated, VIP queuing}	1	1	2.97	
Situated vs. VIP queuing	1	1	2.24	
Within subjects				
Test	7	15	33.20	***
Test × Strategy	14	30	2.50	*
Test × Random Recycling vs. {Situated, VIP queuing}	7	15	5.72	**
Test × Situated vs. VIP queuing	7	15	0.68	

$p < .05$, $^{**}p < .01$, $^{***}p < .001$

4.1.3 Discussion: general

The significant main effect of the test phase merely indicates that subjects learn, which is quite obvious from the increasing learning curves in Figure 23. The absence of a significant effect of strategy, more particularly, of a significant effect for the contrasts between the Situated strategy and the other strategies prohibits claims regarding the relative effect of the Situated strategy. The average curves per strategy (see the left-hand graph in Figure 24) seem to indicate, however, that on average the performance in the Situated condition was better than that in the other conditions, especially than that in the case of Random Recycling. They also seem to indicate an advantage of using VIP queuing as compared to Random Recycling.

The significant interaction between strategy and test phase indicates a different effect of strategy for the different test phases. Strikingly, the effect of strategy and, more particularly, of Random Recycling versus Situated and VIP queuing was only significant for the second test phase. So, both the use of VIP queuing and of the Situated strategy led to better performance in the second test phase than the use of Random Recycling. For VIP queuing, this effect may be explained by the fact that subjects are confronted with the same few words very often till they answer them correctly. In that way, the subjects are likely to learn some words in the first practice phase. For the Situated strategy, it is striking because during the first practice phase the strategy has hardly any memory as regards the past performance of the subjects, namely only the memory with respect to the items that have already been presented in that first practice phase. So, the only adaptation that can occur is that items which have been answered incorrectly (and the first time all items are likely to be answered incorrectly) have a probability of being presented again in the same practice phase (at the expense of other items that have not yet been presented), while this is impossible in the case of Random Recycling. So, when there are many items that can be learned by being presented at least twice, this may produce an initial advantage of the use of the Situated strategy over the use of Random Recycling.

In conclusion, the analysis of the results of the experiment does not show the clear advantage of the Situated strategy as predicted by the models. Nevertheless, Figure 23 seems to indicate an advantage of using the Situated strategy: the variance between subjects is reduced, indicating adaptation in the sense that in the Situated condition there is less difference between high and low performers. Therefore, a post-hoc analysis was performed in which subjects were divided into high and low performers.

4.1.4 Results: high performers versus low performers

For the post-hoc analysis the subjects were divided into two groups per strategy: a group for the high performers and a group for the low performers. The median performance level per strategy was used as a criterion: if the number of correct responses of a subject in most test phases was above or equal to the median of that test phase, the subject was assigned to the group of high performers. On the basis of this criterion, in all strategies five subjects were assigned to the group of high performers and four to the group of low performers. The average learning curves per group and per strategy are shown in Figure 24.

For each group, a MANOVA was performed on logit-transformed proportions of correct responses, with the test phase as a within-subjects factor and the experimental condition (strategy used in the practice phases) as a between-subjects factor. The results of this analysis are summarized in Table 4 and 5. Because of insufficient error degrees of freedom, only the between-subjects effects are available in group 2.

For the high performers, there was a significant main effect of test phase [$F(7,3)=26.4$, $p < .01$] and strategy [$F(2,1)=4.6$, $p < .05$]. The contrast between Random Recycling and Situated or VIP queuing was significant [$F(1,1)=5.18$, $p < .05$], as well as the contrast between Situated and VIP queuing [$F(1,1)=6.38$, $p < .05$]. There was no significant interaction effect between test phase and strategy.

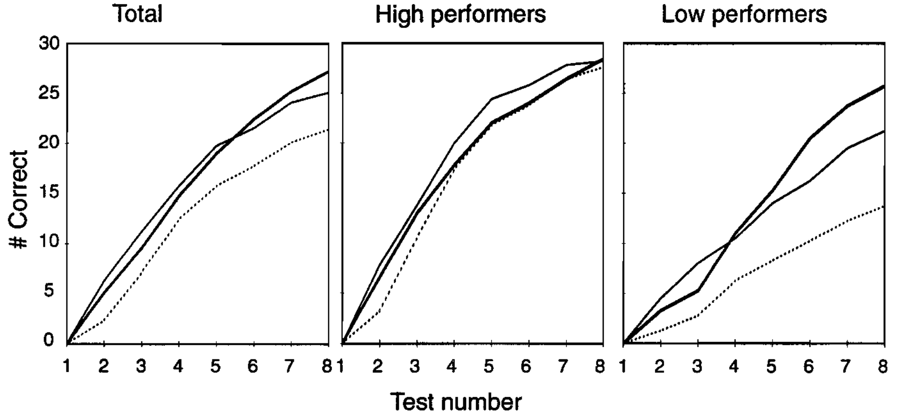


Figure 24: *Averages of the results of Experiment 1 for all subjects, the high performers, and the low performers, respectively. The lines in the graphs show the average of Random Recycling (- - -), VIP queuing (—), and Situated (**—**). The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Table 4: *Results of the analysis for the high performers*

Source	**Num DF**	**Den DF**	**F**	
Between subjects				
Strategy	2	1	4.60	*
Random Recycling vs. {Situated, VIP queuing}	1	1	5.18	*
Situated vs. VIP queuing	1	1	6.38	*
Within subjects				
Test	7	3	26.40	**
Test × Strategy	14	6	1.08	
Test × Random Recycling vs. {Situated, VIP queuing}	7	3	0.71	
Test × Situated vs. VIP queuing	7	3	1.47	

***p* <.05, ***p* <.01, ****p* <.001**

Table 5: *Results of the analysis for the low performers*

Source	**Num DF**	**Den DF**	**F**	
Between subjects				
Strategy	2	1	13.36	**
Random Recycling vs. {Situated, VIP queuing}	1	1	25.51	**
Situated vs. VIP queuing	1	1	0.03	

***p* <.05, ***p* <.01, ****p* <.001**

For the low performers, there is a significant main effect of strategy [$F(2,1)=13.36$, $p < .01$] and a significant effect of the contrast between Random Recycling and Situated or VIP queuing [$F(1,1)=25.51$, $p < .01$]. However, the effect for Situated versus VIP queuing was not significant.

4.1.5 Discussion: high performers versus low performers

For the high performers, the significant effects of the contrasts between Random Recycling and Situated or VIP queuing and between Situated and VIP queuing indicate, in combination with the middle graph in Figure 24, that there is a positive effect of using the Situated strategy and VIP queuing compared to Random Recycling, and of using VIP queuing compared to the Situated strategy. However, this effect is relatively small (see Figure 24). Striking, again, is the advantage in the second test phase of the average learning curves of both the Situated strategy and VIP queuing compared to the average learning curve of Random Recycling. This is the opposite of the predictions of the models.

For the low performers, the significant effect of the contrast between Random Recycling and Situated or VIP queuing indicates, in combination with the right-hand graph in Figure 24, that there is a positive effect of using both the Situated strategy and VIP queuing compared to Random Recycling. As can be seen in Figure 24, this effect is quite large. The effect of Situated versus VIP queuing is not significant, but with only four subjects per condition, the power of the test was of course very small. As can be seen in Figure 24, there is certainly a trend in favour of the Situated strategy for the low performers: though the average learning curve of VIP queuing is steeper initially, the average learning curve of Situated is much steeper after the third test phase.

4.1.6 Results: word level

To obtain more insight into the difficulty distribution of the words, for each combination of strategy and stimulus word the number of subjects that answered that word correctly was calculated for each test phase. An impression of the results is given in Figure 25.

To test the relationship between the difficulty of a word pair, on the one hand, and the number of letters in the response word and the kind of word, on the other hand, an analysis was performed on the Random Recycling results. Unlike the other strategies, in the case of Random Recycling each word pair was presented to the student equally often, and the results of the subjects in the case of Random Recycling therefore give a clear indication of the relative difficulty of the word pairs. As regards the kind of word, a distinction was made between nouns and other words. The curves dependent on the number of letters of the response word and the kind of word are shown in Figure 26.

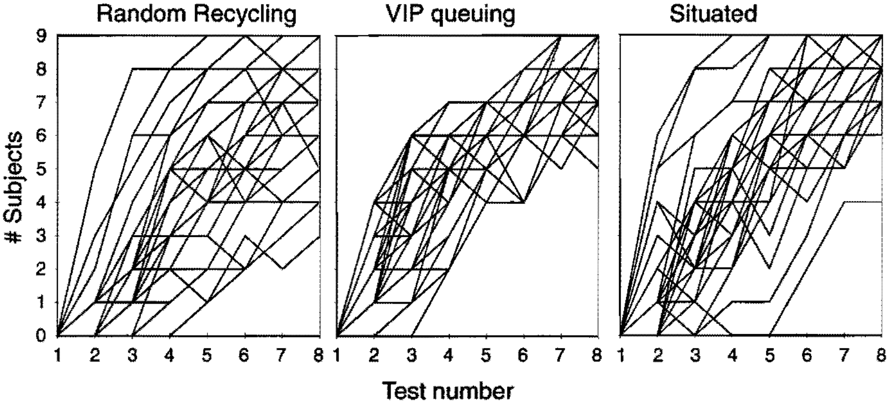


Figure 25: *Results of Experiment 1 on word level for Random Recycling, VIP queuing, and Situated respectively. Each line in the graphs represents a word. The test phases are given on the x-axis and the number of subjects who responded correctly on presentation of the word is shown on the y-axis.*

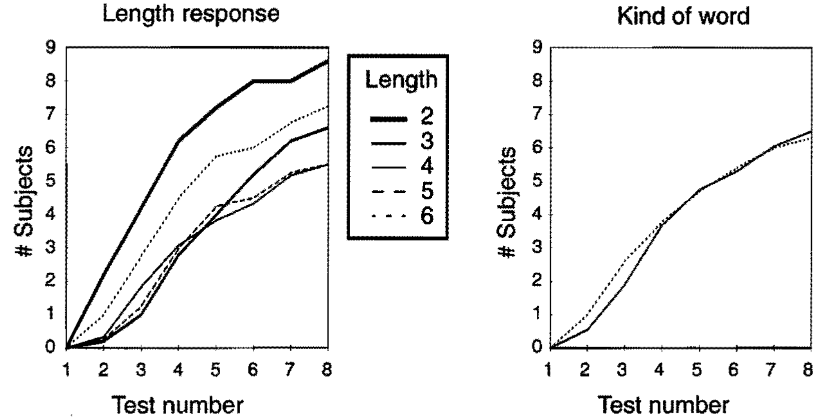


Figure 26: *Random Recycling results of Experiment 1 on word level with average curves for the length of the response word and the kind of word, respectively. Each line in the left-hand graph represents the average over response words of length 2, 3, 4, 5, and 6, respectively. The lines in the right-hand graph represent the average over nouns (solid line) and other kinds of words (dashed line), respectively. The test phases are given on the x-axis and the number of subjects who responded correctly on presentation of the word is shown on the y-axis.*

For Random Recycling, a MANOVA was performed on the numbers of subjects who responded correctly to the word, with the test phase as a within-subjects[1] factor and the number of letters in the response word and the kind of word as between-subjects factors. Two contrasts were performed, namely between a word length of two and a word length of three, four, or five letters, and between a word length of three and a word length of four or five letters. The results of this analysis are summarized in Table 6.

Table 6: Results of the analysis on word level for Random Recycling

Source	Num DF	Den DF	F	
		Between subjects		
Length	4	1	5.14	**
2 vs. {3,4,5}	1	1	15.72	***
3 vs. {4,5}	1	1	0.43	
Kind	1	1	1.77	
Length × Kind	3	1	1.41	
		Within subjects		
Test	7	15	68.01	***
Test × Length	28	56	1.75	*
Test × 2 vs. {3,4,5}	7	15	5.63	**
Test × 3 vs. {4,5}	7	15	2.42	
Test × Kind	7	15	0.88	
Test × Length × Kind	21	44	2.09	*

***p <.05, **p <.01, ***p <.001**

There are significant main effects of test phase [$F(7,15)=68.01$, $p < .001$] and of the length of the response word [$F(4,1)=5.14$, $p < .01$]. The contrast between a word length of two and a word length of three, four, or five letters showed a significant effect. The contrast between a word length of three and a word length of four or five letters was not significant.

A significant interaction was found between test phase and the length of the response word [$F(28,56)=1.75$, $p < .05$] and, more particularly, between test phase and a word length of two versus a word length of three, four, or five letters.

4.1.7 Discussion: word level

Figure 25 gives some insight into the functioning of the different strategies. The graph for Random Recycling clearly shows that not all word pairs are

1. In this case the subjects are word pairs.

equally difficult to learn. Some word pairs are relatively easy to learn, for instance, the highest learning curves are for the words "go" (five), "ii" (good), and "benchi" (bench). Other word pairs are relatively hard to learn, for instance, the lowest learning curves are for the words "tsuki" (moon), "hoka" (other), and "yoru" (evening).

The graph for VIP queuing shows that the large differences between word pairs disappear due to this strategy. Compared to Random Recycling, it takes the subjects longer to learn the easy word pairs, and less time to learn the more difficult word pairs. An explanation may be that VIP queuing reduces the number of word pairs a subject is learning concurrently and that, hence, it may occur that an easy word is only presented to a subject after a number of hard word pairs have been learned. So, VIP queuing may obstruct fast learning of the easy word pairs. On the other hand, once a difficult word has been presented to the subject, it is repeated till the subject has learned it. Thus, VIP queuing benefits the learning of the difficult word pairs.

The graph for the Situated strategy is a mixture of the graphs for the other two strategies. As in the case of Random Recycling, some word pairs are learned relatively fast. Exactly as in the case of Random Recycling, the highest learning curves in the case of the Situated strategy are for the words "go" (five), "benchi" (bench), and "ii" (good). On the other hand, with one exception, namely the word "hoka" (other), all the difficult word pairs are learned faster than in the case of Random Recycling, as in the case of VIP queuing. So, the Situated strategy benefits the learning of the difficult word pairs without obstructing the learning of the easy word pairs. However, sometimes the learning of a very difficult word pair, as in the case of "hoka", is sacrificed for the learning of the easier word pairs. This seems a reasonable use of resources, however.

As far as the analysis of word-pair characteristics is concerned, the significant main effect of test phase merely indicates that the word pairs are learned.The significant main effect of the number of letters in the response word, in combination with Figure 26, indicates that word pairs (in our set) with a response of only two letters are easier to learn than word pairs with a response of greater length. However, as can be seen in Figure 26, a rule of thumb like: the more letters in the response, the more difficult the word pair does not hold good. For instance, the word pairs with a response of six letters tend to be easier to learn than those with four letters. The kind of word also does not give an indication of the relative difficulty. This suggests that more complex factors are involved, such as the abstractness of the word, the resemblance of the response to words in other languages, the ease of constructing memorizing aids. In conclusion, this supports our choice of letting the agent focus on the student's responses in order to discover which word pairs are more difficult, rather than letting it use formal properties of the individual word pairs.

4.1.8 Fit of the models

The fits of only two models are presented, namely the All-or-None model and the Linear model, both on the assumption of equally difficult items. For both models, only one parameter had to be estimated. The model fittings of the other models are omitted, because the added values were marginal having the predictions (see Section 3.5.3). In the discussion of the fits (Section 4.1.9), these predictions will also be compared with the experimental data.

The fits were obtained by using the same procedure as in the case of the predictions, as discussed in Section 3.5.1 and shown in Figure 13. However, in case of the fits, the parameters of the models were varied while calculating which parameter setting produced the best fit.

Instead of fitting the model to the average of the subjects in a certain experimental condition, we opted to fit all subjects individually. In that way, the model also has to explain the variance between the subjects.

Though we could have fitted all the subjects with one parameter setting, we opted to use one parameter setting for each individual subject. This enabled the model to explain individual differences between subjects on the basis of different parameter settings.

We used a least squares measure, in which the fitting process minimizes the sum over test phases of the squares of the difference between the number of correct responses of a subject and the number of correct responses of the model. We will denote this minimal sum with lsq (least square).

Fit of the All-or-None model

The All-or-None model was fitted on the data of Experiment 1 by varying the value of parameter α. The mean and standard deviations of the lsq were m=41.27, sd=33.14 for Random Recycling, m=12.52, sd=6.81 for VIP queuing, and m=19.39, sd=12.26 for the Situated strategy. Figure 27 shows the distribution of the values of α needed for the fits for the various strategies. The learning curves corresponding to these parameter values are shown in Figure 28.

Pearson correlation coefficients were determined between the experimental data and the fit per strategy, and between the data and the fit per strategy per test phase. These coefficients are shown in Table 7.

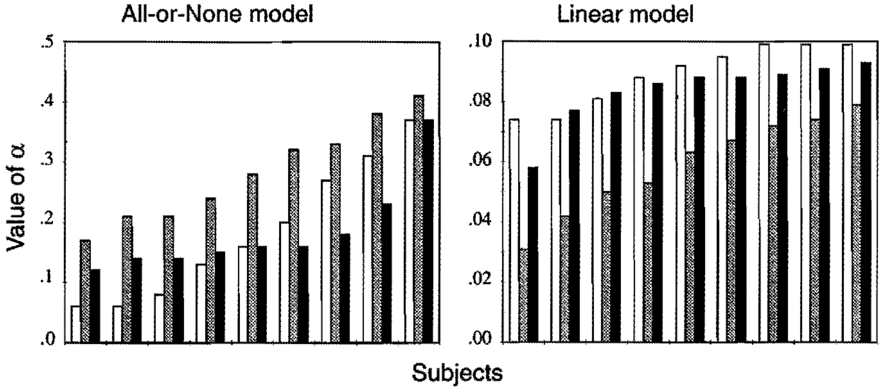


Figure 27: *Distribution over the subjects of the estimated parameter of the All-or-None model and the Linear model, respectively, for (from left to right) Random Recycling (white), VIP queuing (gray), and Situated (black), respectively. The subjects are given on the x-axis and the value of the parameter is shown on the y-axis.*

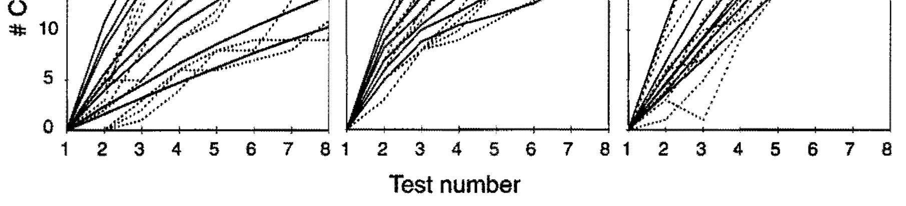


Figure 28: *Results of the fit of the All-or-None model on the data of Experiment 1 for Random Recycling, VIP queuing, and Situated, respectively. Each solid line in the graphs represents a fitted subject, each dashed line represents a real subject. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Table 7: Pearson correlation coefficients between the fit of the All-or-None model and the data of Experiment 1.

Source	**Test phase**							
	Total	T 2	T 3	T 4	T 5	T 6	T 7	T 8
Random Recycling	.977	.689	.981	.973	.962	.983	.990	.988
VIP queuing	.990	.904	.952	.994	.999	.982	.975	.976
Situated	.982	.842	.936	.974	.961	.947	.807	.870

Fit of the Linear model

The Linear model was fitted on the data of Experiment 1 by varying the value of parameter α. The mean and standard deviations of the lsq were m=179.21, sd=44.48 for Random Recycling, m=8.89, sd=4.82 for VIP queuing, and m=97.67, sd=60.01 for the Situated strategy. The distribution over the subjects of the value of α for which the best fit was obtained is shown in Figure 27. The learning curves corresponding to these parameter values are shown in Figure 29. Pearson correlation coefficients were determined between the experimental data and the fit per strategy, and between the data and the fit per strategy per test phase. These coefficients are shown in Table 8.

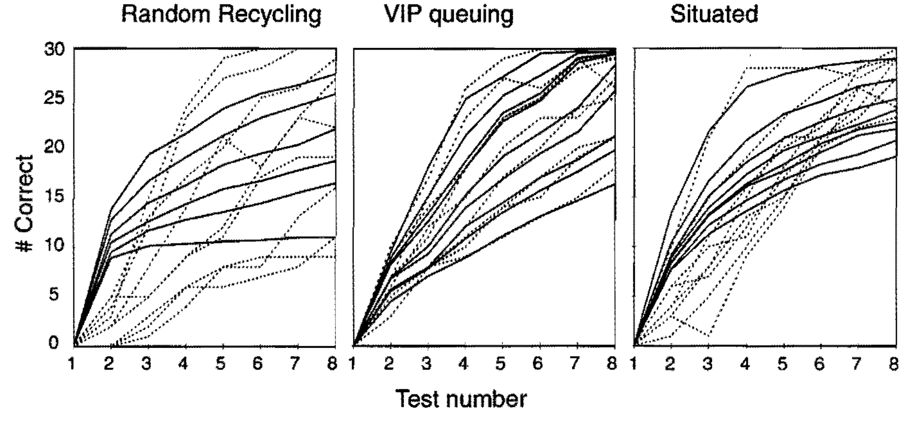


Figure 29: Results of the fit of the Linear model on the data of Experiment 1 for Random Recycling, VIP queuing, and Situated, respectively. Each solid line in the graphs represents a fitted subject, each dashed line represents a real subject. The test phases are given on the x-axis and the number of correct responses on the y-axis.

Table 8: *Pearson correlation coefficients between the fit of the Linear model and the data of Experiment 1.*

Source	**Test phase**							
	Total	T 2	T 3	T 4	T 5	T 6	T 7	T 8
Random Recycling	.932	.640	.968	.984	.975	.979	.967	.967
VIP queuing	.990	.924	.954	.984	.994	.989	.982	.966
Situated	.823	.452	.437	.586	.455	.331	-.045	-.165

4.1.9 Discussion: fit of the models

For the All-or-None model, both the correlation coefficients (see Table 7) and the near-coincidence of the learning curves of the model and those of the real subjects (see Figure 28) indicate that the model fits the data very well. Only the beginning of the curves, particularly in the second test phase, is fitted relatively poorly, especially in the case of Random Recycling (correlation coefficient of .689). The learning curves of the real subjects start less steeply than those of the model.

However, even if the relatively bad fit of the beginning of the curves is neglected, the good fit does not imply that the model is correct. As can be seen in Figure 27, there is definitely an effect of strategy on the α distribution. According to this fit, the subjects in the VIP-queuing condition were relatively good (high α values) and those in the Situated condition were relatively bad (low α values). This is not very likely, given the random assignment of subjects to conditions.

The All-or-None model predicted (see Section 3.5.3) that the Situated strategy would have a great advantage over the other two strategies as regards its effect on the subjects' performance. It also predicted that VIP queuing would produce better results than Random Recycling for the very high performers among the students, but very poor results for the lower performers. Both predictions turned out to be incorrect.

The All-or-None model with varying difficulty levels predicted (see Section 3.5.3) even worse performance of VIP queuing due to a blockage effect. This is not reflected in the experimental data. It also predicated a better performance of Random Recycling in the second test phase. The experimental data indicate the opposite effect. The same incorrect predictions were given by the two-stage learning model (see Section 3.5.3).

For the Linear model, both the correlation coefficients (see Table 8) and the obvious difference between the learning curves of the model and those of the real subjects for both Random Recycling and the Situated strategy (see

Figure 29) indicate that the model fits the data rather poorly. The data of the Situated strategy, especially, are fitted very poorly. With this model, the second test phase of Random Recycling also shows an relatively poor fit, with a much steeper increase for the model than for the real subjects. Moreover, the α distribution is unequal for the different strategies (see Figure 27).

The Linear model predicted (see Section 3.5.3) a small advantage of the use of Random Recycling over the use of the Situated strategy, and an advantage of both over the use of VIP queuing. These predictions also turned out to be incorrect.

In conclusion, all the models discussed have shortcomings of various kinds. Two of them are very striking, and may provide a direction for possible improvements. In the first place, all the models discussed produce steeper starts of the learning curves than the real subjects. This points in the direction of neural networks, which, like the real subjects, have a tendency to start slowly. It may also indicate partial learning.

In the second place, all the models largely underestimate the effect of VIP queuing. This suggests that they lack a mechanism which explains the positive features of VIP queuing. Such a feature may be that the cognitive load of the student is reduced by not presenting all the items at once, but only presenting an item when most of the items previously presented have been learned. This points in the direction of including in the models a limit to the maximum number of items a student can learn at the same time. If this limited capacity is true, then the Situated strategy may be improved by adding an extra set of items, which represents the non-presented items, and presenting one of these items only whenever the number of items in the bad set is below a certain limit.

4.2 Experiments 2a (recognition) and 2b (recall)

Two more experiments in the domain of paired associates learning were performed as a sequel to Experiment 1. For both experiments, a within-subjects design was used, and the comparison of strategies was restricted to Random Recycling and the Situated strategy, because we expected that this would give us enough insight into the learning process. For both experiments, both a recognition and a recall variant were designed in order to explore possible differences between these two kinds of paired associates learning as regards the effectivity of the strategies.

First, an experiment will be presented in which the memory of the Situated strategy was cleared after each test phase: the strategy had to start all over again after every test phase with all the items in the bad set. The goal of this experiment was to investigate whether the steep rise of the learning curves in the second test phase in the case of the Situated strategy compared to Random Recycling (see above) would continue during the whole learning session when the strategy worked in all practice phases under the same condition as in the first practice phase (namely with no initial assumptions of the student's know-ledge). Next (Section 4.3), an experiment will be presented in which the memory was not cleared, so the same Situated strategy was used as in Experiment 1.

4.2.1 Method

Design and Procedure.
In both experiments, 2a and 2b, the subjects had to learn Japanese translations of Dutch words (Japanese words written as pronounced). In recognition Experiment 2a, Dutch words were presented on the screen together with the translations of all the words (see Figure 30) with the request to select as accurately and quickly as possible the translation into Japanese (words written as pronounced). To prevent the subjects from learning positions on the screen rather than translations, at each presentation of a Dutch word, the translations were randomly assigned a position. This has the disadvantage that the subjects have to search the whole screen in order to find a correct translation.

In recall Experiment 2b, Dutch words were presented on the screen (see Figure 22) with the request to type as accurately and quickly as possible the translation into Japanese (words written as pronounced).

In both experiments, the subjects were presented with alternative test and practice phases. In test phases, all 30 Dutch words were presented to the subjects in a random order. No feedback was given with respect to the correctness of the translation. In practice phases, feedback was given with respect to the correctness of the translation, and the correct translation was displayed on the screen for 2 seconds. Answers and response times were recorded.

Trials stopped when a subject had translated all the words correctly in two subsequent test phases, or when eight test phases had passed in the recognition variant, and ten test phases had passed in the recall variant. This maximum number of test phases was determined by running pilots such that a subject would need no more than approximately one hour to complete the experiment.

In both experiments, a within-subject design was used: each subject learned two sets of Japanese words, one with Random Recycling, the other with the Situated item sequencing strategy in the practice phases. The two sessions were taken one week apart. To control for possible session and set effects, the subjects were divided into four groups: one group that started with Random Recycling and Word set 1, followed by the Situated strategy and Word set 2, a second group that started with Random Recycling and Word set 2, followed by the Situated strategy and Word set 1, a third group that started with the Situated strategy and Word set 1, followed by Random Recycling and Word set 2, and a fourth group that started with the Situated strategy and Word set 2, followed by Random Recycling and Word set 1. So, there was an experimental between-subject variable "condition" with four levels, dependent on the strategy and word set used in the first session.

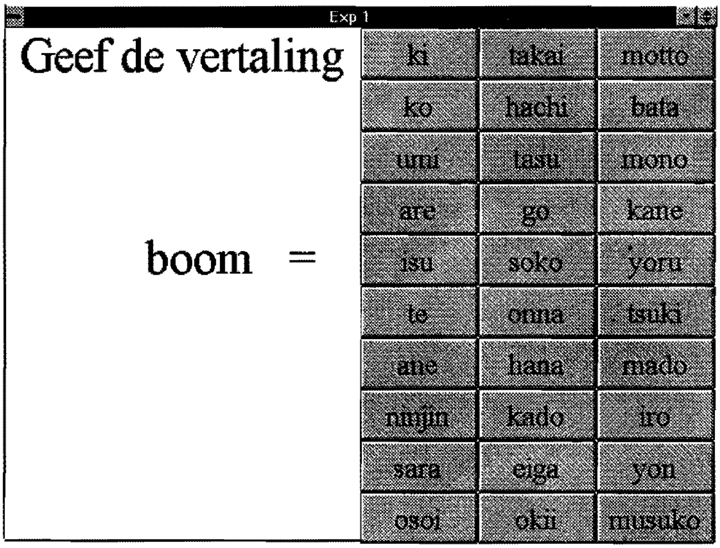


Figure 30: *Screen layout of the recall Experiment 2a and 3a.*

Subjects

In both Experiment 2a and Experiment 2b, twenty-four subjects of the higher classes (4 and 5) of a local high school participated voluntarily, with an average age of 16. Subjects were paid for their participation. All subjects had no prior experience with Japanese. Subjects were randomly assigned to one of the four experimental conditions.

Equipment

The material was presented in black and white on a PC. In the recall Experiment 2b the subjects used an ordinary keyboard to type in the responses; use of the mouse was not needed. In the recognition Experiment 2A subjects used a mouse to select the translation. No use was made of audio. All instructions and feedback were displayed on the computer screen.

Materials

Because of the within-subject design, two sets of word pairs were needed with approximately the same level of difficulty. The sets used in these experiments are shown in Table 9. As in the previous experiment, all word pairs were selected from the first lessons of a beginners' course in Japanese.

Table 9: *The two sets of word pairs used in Experiment 2a and 2b. The English translations are given in parentheses.*

Set 1			**Set 2**		
raam	mado	(window)	omdat	kara	(because)
avond	yoru	(evening)	water	mizu	(water)
acht	hachi	(eight)	kamer	heya	(room)
kaart	kado	(card)	spin	kumo	(spider)
maan	tsuki	(moon)	geur	nioi	(scent)
neus	hana	(nose)	zwaar	omoi	(heavy)
groot	okii	(large)	regen	ame	(rain)
film	eiga	(movie)	snel	hayai	(fast)
hand	te	(hand)	drie	san	(three)
duur	takai	(expensive)	moed	gattsu	(courage)
vier	yon	(four)	bier	nama	(beer)
zus	ane	(sister)	twee	ni	(two)
laat	osoi	(late)	lente	haru	(spring)
vrouw	onna	(woman)	doos	hako	(box)
plus	tasu	(plus)	hier	koko	(here)
die	are	(that)	werk	baito	(work)
ding	mono	(thing)	huis	ie	(house)
kind	ko	(child)	goed	ii	(good)
geld	kane	(money)	rood	aka	(red)
zee	umi	(sea)	hoed	boshi	(hat)
daar	soko	(there)	voet	ashi	(foot)
zoon	musuko	(son)	deur	doa	(door)
boter	bata	(butter)	klok	tokei	(clock)
stoel	isu	(chair)	berg	yama	(mountain)
meer	motto	(more)	tand	ha	(tooth)
kleur	iro	(colour)	voor	mae	(before)
peen	ninjin	(carrot)	bank	benchi	(bench)
boom	ki	(tree)	boek	hon	(book)
bord	sara	(plate)	ander	hoka	(other)
vijf	go	(five)	land	kuni	(land)

To obtain sets of approximately equal difficulty, the sets were constructed according to the following criteria.

1. The distribution of the length of the Japanese translations was the same for both sets. The results of Experiment 1 on word level suggest that this is an important issue.

2. Similar kinds of words were used in both sets, both as regards grammatical category (for instance noun or adjective) and as regards frequency of occurrence in Dutch. The latter was checked with the aid of the word frequency lists of Uit den Boogaart (1975).

3. The number of word pairs which may be highly confusable, because the translations are alike, was the same for both sets. For instance, boom=ki, kind=ko in Set 1, and goed=ii, huis=ie in Set 2.

4. Interdependence between the sets, in the sense that the learning of words from one set has an obvious influence on the learning of a word from the other set was prevented (as far as possible). For instance, learning the words ki (tree) and iro (colour) might have had an influence on the learning of kiiro (yellow).

A pilot study was carried out to get an indication of whether both sets were indeed of approximately equal difficulty. Six subjects took part in the pilot study. Each subject performed the recall experiment as described above, only now Random Recycling was used in both sessions. Half of the subjects started with Set 1, the other half with Set 2. The results of this pilot study are shown in Figure 31.

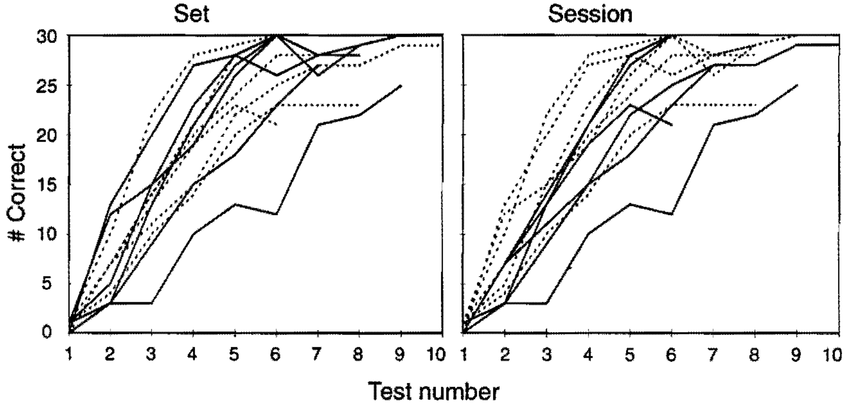


Figure 31: *Results of the pilot study. Each line in the graphs represents the learning curve of a subject in a certain session with a certain set of word pairs. So, there are two lines per subject. In the left-hand graph, the solid lines represent the learning curves of the subjects when using the word pairs of Set 1 and the dashed lines the learning curves of the subjects when using the word pairs of Set 2. In the right-hand graph, the solid lines represent the learning curves of the subjects during the first session and the dashed lines the learning curves of the subjects during the second. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

The graphs give no indication of an effect of set on the performance of the subjects, which has been confirmed by statistical analysis. There is, however, a statistically significant effect of session [$F(1,4)=9.99$, $p < .05$]. Based on this pilot study, we have assumed both sets to be sufficiently equivalent, and the sets were consequently used in the experiments.

4.2.2 Results

The results of the experiments are shown in Figure 32 and 33. A MANOVA was performed on logit-transformed proportions of correct responses, with the test phase and the strategy used in the practice phases as within-subjects factors, and the experimental condition (determined by the strategy and word set used in the first session) as a between-subjects factor. The results of this analysis are summarized in Table 10 and 11.

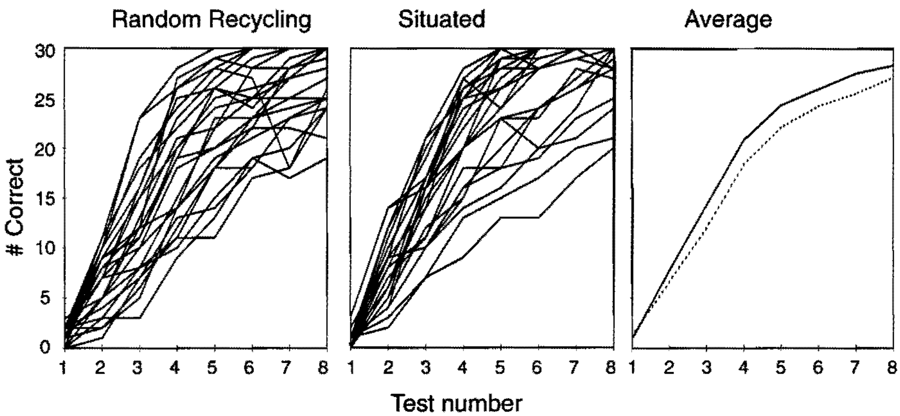


Figure 32: *Results of Experiment 2A (recognition) for Random Recycling, Situated, and the average of both strategies, respectively. Each line in the two left-hand graphs represents a subject, the two lines in the right-hand graph represent the average of Random Recycling (dashed line) and the average of Situated (solid line) respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

For recognition Experiment 2A, there are significant main effects of test phase [$F(7,14)=109.98$, $p < .001$] and strategy [$F(1,20)=8.58$, $p < .05$]. A significant interaction was found between test phase and strategy [$F(7,14)=3.3$, $p < .05$]. Testing the effect of strategy per test phase revealed that strategy was significant [$p < .05$] in all test phases except the first two.

Table 10: Results of the MANOVA on the data of Experiment 2a (recognition)

Source	Num DF	Den DF	F	
	Between subjects			
Condition	3	1	0.08	
Set 1 vs. Set 2	2	1	0.01	
Session 1 vs. Session 2	2	1	0.10	
	Within subjects			
Test	7	14	109.98	***
Strategy	1	20	8.58	*
Strategy × Condition	3	20	2.24	
Strategy × Set 1 vs. Set 2	2	20	0.56	
Strategy × Session 1 vs. Session 2	2	20	2.92	
Test × Condition	21	41	1.36	
Test × Set 1 vs. Set 2	14	28	1.48	
Test × Session 1 vs. Session 2	14	28	1.09	
Strategy × Test	7	14	3.30	*
Strategy × Test × Condition	21	41	2.87	**
Strategy × Test × Set 1 vs. Set 2	14	28	2.16	*
Strategy × Test × Session 1 vs. Session 2	14	28	2.33	*

***p <.05, **p <.01, ***p <.001**

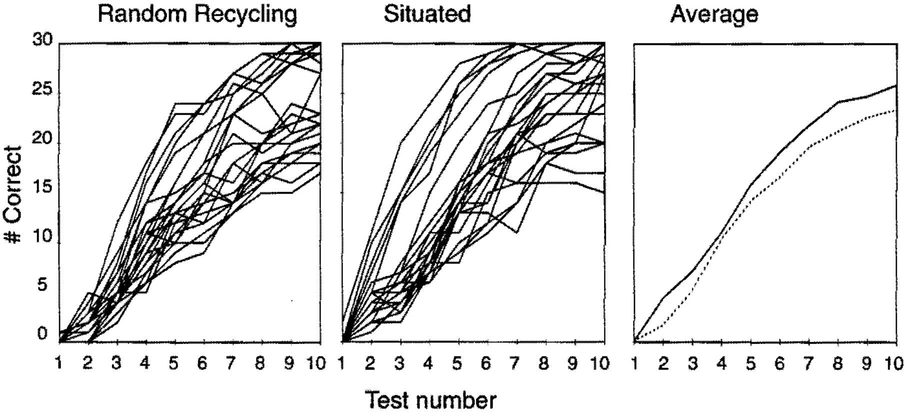


***Figure 33:** Results of Experiment 2B (recall) for Random Recycling, Situated, and the average of both strategies, respectively. Each line in the two left-hand graphs represents a subject, the two lines in the right-hand graph represent the average of Random Recycling (dashed line) and the average of Situated (solid line) respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

For recall Experiment 2B, there are significant main effects of test phase [$F(9,12)=332$, $p < .001$], strategy [$F(1,20)=14.41$, $p < .01$], and condition [$F(3,1)=3.89$, $p < .05$], and also (with pairwise comparison) of experimental session [$F(2,1)=5.72$, $p < .05$].

Table 11: *Results of the MANOVA on the data of Experiment 2b (recall)*

Source	Num DF	Den DF	F
Between subjects			
Condition	3	1	3.89 *
Set 1 vs. Set 2	2	1	0.67
Session 1 vs. Session 2	2	1	5.72 *
Within subjects			
Test	9	12	332.00 ***
Strategy	1	20	14.41 **
Strategy × Condition	3	20	4.30 *
Strategy × Set 1 vs. Set 2	2	20	1.31
Strategy × Session 1 vs. Session 2	2	20	5.23 *
Test × Condition	27	36	1.87 *
Test × Set 1 vs. Set 2	18	24	0.98
Test × Session 1 vs. Session 2	18	24	2.84 **
Strategy × Test	9	12	5.88 **
Strategy × Test × Condition	27	36	2.29 *
Strategy × Test × Set 1 vs. Set 2	18	24	3.17 **
Strategy × Test × Session 1 vs. Session 2	18	24	1.67

***p <.05, **p <.01, ***p <.001**

Significant interactions were found between strategy and test phase [$F(9,12)=5.88$, $p < .01$], between condition and test phase [$F(27,36)=1.87$, $p < .05$], and, more particularly, between session and test phase [$F(18,24)=2.84$, $p < .01$]. Testing the effect of strategy and session per test phase revealed that strategy was significant [$p < .05$] in all test phases except the first, fourth, and last two, and that session was significant [$p < .05$] in all test phases except the first two.

Another significant interaction was found between strategy and condition [$F(3,20)=4.3$, $p < .05$] and, more particularly, between strategy and session [$F(2,20)=5.23$, $p < .05$]. Testing the effect of condition per level of strategy (see Table 12) revealed that the effects of condition and, more particularly, session were only significant for the Situated strategy [$F(2,1)=8.41$, $p < .01$].The average learning curves per strategy per session are shown in Figure 34.

Testing the effect of strategy per level of condition (see Table 13) revealed that it was only significant for one of the four conditions, namely the condition in which Random Recycling and Word set 2 were used in the first session (followed by Situated and Word set 1 in the second session). The average learning curves per strategy per condition are shown in the left-hand graph in Figure 34

Table 12: *Results of the MANOVA on the data of Experiment 2b (recall) per strategy*

Source	Num DF	Den DF	F	
Random Recycling: Between subjects				
Condition	3	1	1.00	
Set 1 vs. Set 2	2	1	0.39	
Session 1 vs. Session 2	2	1	1.45	
Situated: Between subjects				
Condition	3	1	5.99	**
Set 1 vs. Set 2	2	1	1.10	
Session 1 vs. Session 2	2	1	8.41	**

***p <.05, **p <.01, ***p <.001**

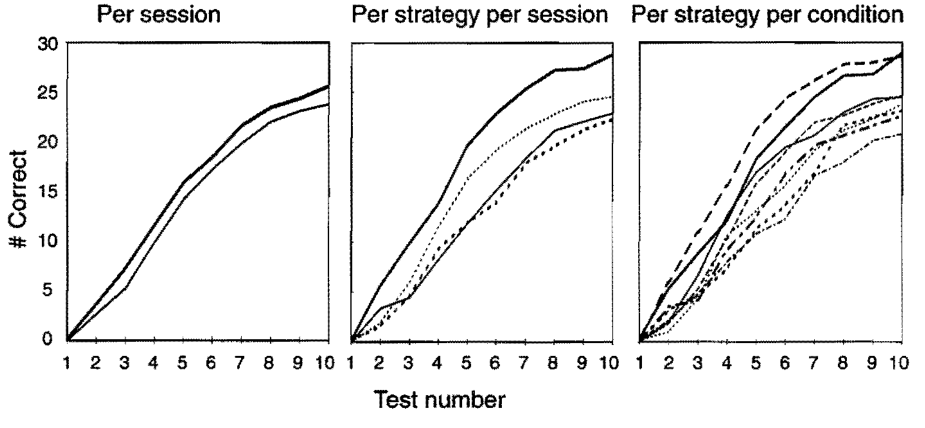


Figure 34: *Results of Experiment 2B (recall) regarding session effects. The lines in the left-hand graph represent the average over subjects of the first session (thin line) and the second session (bold line), respectively. The lines in the middle graph represent the average of Random Recycling in the first session (dashed line) and second session (dashed bold line), and Situated in the first session (solid line) and second session (solid bold line), respectively. The lines in the right-hand graph represent the average of Random Recycling (thin) and Situated (bold) per condition. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Table 13: Results of the MANOVA on the data of Experiment 2b (recall) per condition

Condition	Source	Num DF	Den DF	F	
Random Recycling 1 - Situated 2	Strategy	1	5	5.43	
Situated 1 - Random Recycling 2	Strategy	1	5	2.34	
Random Recycling 2 - Situated 1	Strategy	1	5	28.24	**
Situated 2 - Random Recycling 1	Strategy	1	5	0.04	

****p* <.05, ***p* <.01, ****p* <.001**

4.2.3 Discussion

In both the recognition and recall study, the significant main effect of test phase merely indicates that subjects learn, which is quite obvious from the increasing learning curves of Figure 32 and 33.

The effect of the between-subjects variable condition is only significant for the recall study, and the contrasts showed that this effect was due to a session effect. The lack of an effect of word set indicates that both word sets are of approximately equal difficulty, as intended. By Analogy with the findings of the pilot study as discussed above, we would expect subjects to perform better in the second session than in the first. On average this is the case, as reflected in the left-hand graph in Figure 34. The session effect, and the lack of it in the case of the recognition study, may be explained by subjects learning the typical structure, in the sense of frequent letter combinations, of Japanese words. This improves the learning of the response words in the case of recall, but has no impact on the learning of the associations between stimuli and response words.

Focusing, in the recall study, on the effect of condition per strategy revealed that the session effect was only significant for the Situated strategy, and, as can be seen in the middle graph of Figure 34, the trend in the case of Random Recycling was even reverse: subjects tended to perform worse in the second session. A possible explanation may be that subjects who had the Situated strategy during the first session may have been demotivated when confronted with Random Recycling during the second session, either because they understood it is a control condition, or because they interpreted the lack of adaptation (which they got used to during the first session with the Situated strategy) as a consequence of their own poor performance.

The significant main effect of strategy, in both the recognition and recall study, demonstrates that using the Situated strategy had an advantageous effect on the subjects' performance. The average learning curves of the Situated strategy clearly lie above the average learning curves of Random Recycling (see Figure 32 and 33). The difference between the curves is not very great, but we had not expected a large difference, as the memory of the Situated strategy is cleared after every test phase in this experiment. The effect found confirms our hypothesis that a difference can be found even with such a small amount of time for the Situated strategy to work. We expect that this effect will increase once the strategy is given more time to adapt to the student by no longer clearing its memory.

Focusing, in the recall study, on the effect of strategy per condition revealed that the effect of strategy was significant for only one of the four conditions: namely the condition in which Random Recycling and Word set 2 were used in the first session (followed by Situated and Word set 1 in the second session). Nevertheless, for three of the four conditions the average learning curve of the Situated strategy clearly lies above the average learning curve of Random Recycling (see the right-hand graph in Figure 34). So, there is certainly a trend in favour of the Situated strategy. With only six subjects per condition, the power of the test was, of course, relatively small.

In conclusion, there is a small positive effect of using the Situated strategy both in the recognition and the recall study, even though the memory of the Situated strategy was cleared regularly (after each test phase), so that the Situated strategy did not have much time to adapt to the subjects. In the following experiments, we test the hypothesis that this effect will increase when the memory of the Situated strategy is no longer cleared.

4.3 Experiments 3a (recognition) and 3b (recall)

4.3.1 Method

Design and Procedure.
The same design and procedure were used as in Experiment 2a and 2b, except that now the memory of the Situated strategy was no longer cleared after each test phase. Therefore, the same version of the Situated strategy was used as in Experiment 1.

Subjects
In both Experiment 3a and 3b twenty-four subjects with university or higher vocational training participated voluntarily. The average age was 24. No subject had any prior experience of Japanese. Subjects were randomly assigned to one of the four experimental conditions.

Equipment
The same equipment was used as in Experiment 2a and 2b.

Materials
The same sets of word pairs were used as in Experiment 2a and 2b, as shown in Table 9.

4.3.2 Results: general

The results of the experiments are shown in Figure 35 and 37. A MANOVA was performed on logit-transformed proportions of correct responses, with the test phase and the strategy used in the practice phases as within-subjects factors, and the experimental condition (determined by the strategy and word set used in the first session) as a between-subjects factor. The results of this analysis are summarized in Table 14 and 17. One subject was excluded from the analysis of Experiment 3A because he did not complete the experiment. This subject had the condition in which Random Recycling and Word set 1 were used in the first session.

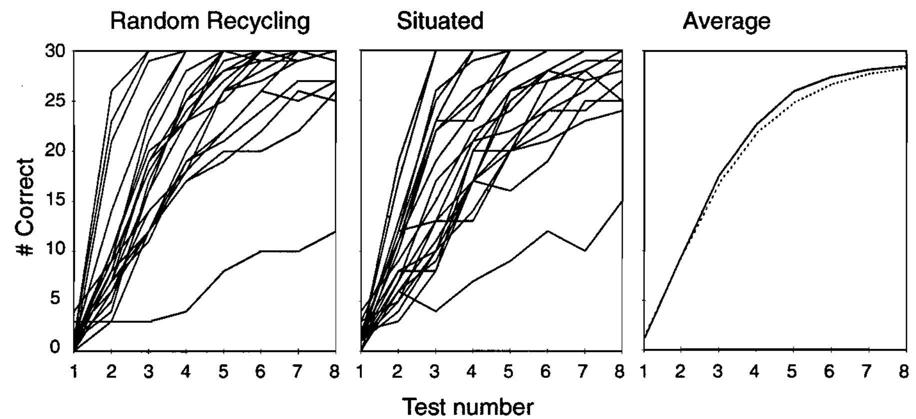


***Figure 35:** Results of Experiment 3A (recognition) for Random Recycling, Situated, and the average of both strategies, respectively. Each line in the two left-hand graphs represents a subject, the two lines in the right-hand graph represent the average of Random Recycling (dashed line) and the average of Situated (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

For recognition Experiment 3A, there are significant main effects of test phase [$F(7,13)=56.19$, $p < .001$] and (with pairwise comparison) session [$F(2,1)=8.73$, $p < .05$]. Significant interactions were found between strategy and condition [$F(3,19)=5.56$, $p < .01$], and, more particularly, between strategy and session [$F(2,19)=8.2$, $p < .01$]. Testing the effect of condition per level of strategy (see Table 15) revealed that the effects of condition, and, more particularly, session were only significant for the Situated strategy [$F(2,1)=12.12$, $p < .001$].The average learning curves per strategy per session are shown in Figure 36.

Table 14: *Results of the MANOVA on the data of Experiment 3a (recognition)*

Source	Num DF	Den DF	F	
Between subjects				
Condition	3	1	2.54	
Set 1 vs. Set 2	2	1	0.14	
Session 1 vs. Session 2	2	1	3.73	*
Within subjects				
Test	7	13	56.19	***
Strategy	1	19	0.35	
Strategy × Condition	3	19	5.56	**
Strategy × Set 1 vs. Set 2	2	19	0.87	
Strategy × Session 1 vs. Session 2	2	19	8.20	**
Test × Condition	21	38	0.87	
Test × Set 1 vs. Set 2	14	26	0.68	
Test × Session 1 vs. Session 2	14	26	0.86	
Strategy × Test	7	13	0.29	
Strategy × Test × Condition	21	38	1.15	
Strategy × Test × Set 1 vs. Set 2	14	28	1.04	
Strategy × Test × Session 1 vs. Session 2	14	26	1.25	

$^*p < .05$, $^{**}p < .01$, $^{***}p < .001$

Table 15: *Results of the MANOVA on the data of Experiment 3A per strategy*

Source	Num DF	Den DF	F	
Random Recycling: Between subjects				
Condition	3	1	0.40	
Set 1 vs. Set 2	2	1	0.29	
Session 1 vs. Session 2	2	1	0.48	
Situated: Between subjects				
Condition	3	1	8.09	**
Set 1 vs. Set 2	2	1	0.20	
Session 1 vs. Session 2	2	1	12.12	***

$^*p < .05$, $^{**}p < .01$, $^{***}p < .001$

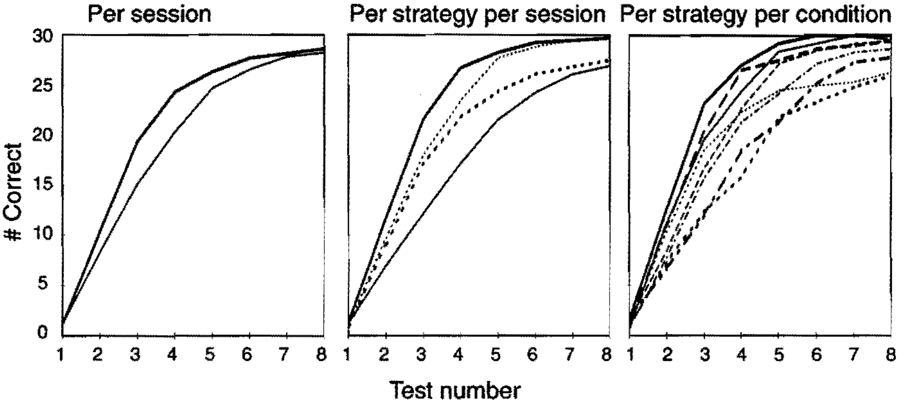


Figure 36: *Results of Experiment 3A (recognition) regarding session effects. The lines in the left-hand graph represent the average over subjects of the first session (thin line) and the second session (bold line), respectively. The lines in the middle graph represent the average of Random Recycling in the first session (dashed line) and second session (dashed bold line), and Situated in the first session (solid line) and second session (solid bold line), respectively. The lines in the right-hand graph represent the average of Random Recycling (thin) and Situated (bold) per condition. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Testing per level of condition (see Table 16) revealed that the effect of strategy was only significant for one of the four conditions, namely the condition in which Random Recycling and Word set 1 were used in the first session (followed by Situated and Word set 2 in the second session). The average learning curves per strategy per condition are shown in the left-hand graph of Figure 36.

Table 16: *Results of the MANOVA on the data of Experiment 3A per condition*

Condition	Source	Num DF	Den DF	F
Random Recycling 1 - Situated 2	Strategy	1	4	33.40 **
Situated 1 - Random Recycling 2	Strategy	1	4	2.23
Random Recycling 2 - Situated 1	Strategy	1	4	1.82
Situated 2 - Random Recycling 1	Strategy	1	4	5.05

***p <.05, **p <.01, ***p <.001**

For recall Experiment 3B, there are significant main effects of test phase [$F(9,12)=137.59$, $p < .001$] and strategy [$F(1,20)=4.51$, $p < .05$]. All other effects are not significant.

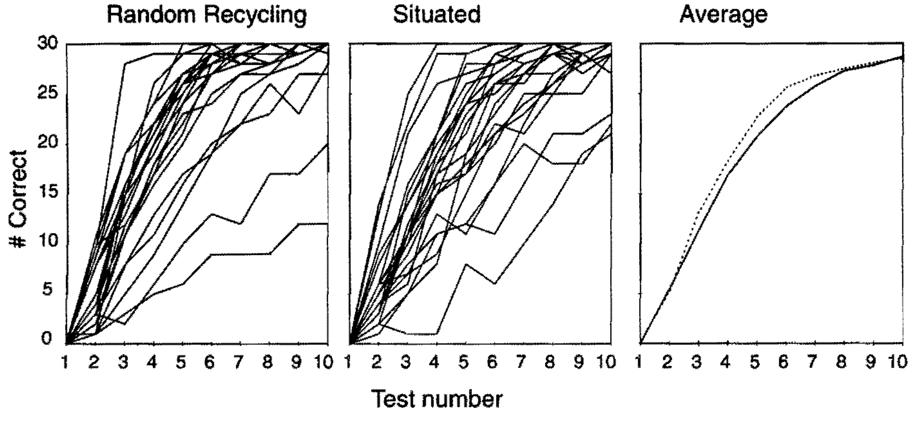


Figure 37: *Results of Experiment 3B (recall) for Random Recycling, Situated, and the average of both strategies, respectively. Each line in the two left-hand graphs represents a subject, the two lines in the right-hand graph represent the average of Random Recycling (dashed line) and the average of Situated (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Table 17: *Results of the MANOVA on the data of Experiment 3b (recall)*

Source	**Num DF**	**Den DF**	**F**	
Between subjects				
Condition	3	1	1.73	
Set 1 vs. Set 2	2	1	2.52	
Session 1 vs. Session 2	2	1	0.87	
Within subjects				
Test	9	12	137.59	***
Strategy	1	20	4.51	*
Strategy × Condition	3	20	2.90	
Strategy × Set 1 vs. Set 2	2	20	3.42	
Strategy × Session 1 vs. Session 2	2	20	1.33	
Test × Condition	27	36	1.65	
Test × Set 1 vs. Set 2	18	24	1.78	
Test × Session 1 vs. Session 2	18	24	1.91	
Strategy × Test	9	12	1.31	
Strategy × Test × Condition	27	36	1.65	
Strategy × Test × Set 1 vs. Set 2	18	24	1.83	
Strategy × Test × Session 1 vs. Session 2	18	24	1.54	

***p <.05, **p <.01, ***p <.001**

4.3.3 Discussion: general

There is only a significant session effect for the recognition study. As in Experiment 2, the lack of an effect of word set indicates that both word sets are approximately equally difficult, as intended. In this experiment again, the subjects performed better on average in the second session than in the first (see left-hand graph of Figure 36). However, our explanation after Experiment 2 of both the session effect in the case of recall and the absence of a session effect in the case of recognition –namely, the subjects having learned the typical structure, in the sense of frequent letter combinations, of Japanese words in the first session– turns out to be incorrect. After all, in this case there is an effect in the case of recognition (and even none in the case of recall). An alternative explanation may be that the session effect is due to a change in the motivation of the subjects over the two sessions, and can thus, in principal, occur in both the recall and the recognition variant.

Focusing on the effect of session per strategy in the recognition study revealed that the session effect was only significant for the Situated strategy, and, as can be seen in the middle graph of Figure 36, the trend in the case of Random Recycling was even the reverse: subjects tended to perform worse in the second session. This is exactly what we found in the case of the recall variant of Experiment 2 and, therefore, supports the same explanation. This explanation is that subjects who had the Situated strategy during the first session may be demotivated when confronted with Random Recycling during the second session, either because they understand it is a control condition, or because they interpret the lack of adaptation (which they got used to during the first session with the Situated strategy) as a consequence of their own bad performance. The positive session effect for the Situated strategy may then be explained by an extra motivation of subjects who had Random Recycling in the previous session, because of a difference in the strategy which was experienced as positive.

In the recognition study, the absence of a significant main effect of strategy and the near coincidence of the average learning curves per strategy (see the right-hand graph in Figure 35) suggests that using the Situated strategy had no advantageous effect on the subjects' performance. In the recall study, it is even worse: there is a significant main effect of strategy, but this indicates (see the right-hand graph in Figure 37) an advantage, though very limited, of using Random Recycling. This is not at all what we expected, especially as Experiment 2 indicated, for both recognition and recall, a significant advantage of using the Situated strategy even when the memory of the Situated strategy was restricted.

There are several reasons why the use of Situated strategy may not have resulted in the substantial expected advantageous effect on the subjects' performance.

1. *Too slow transfers to the good set.* Adaptation to a student's performance takes time. The system can only discover that a student has learned the translation of a certain word when that word has been presented again and answered correctly. This implies that when a student learns very fast, the Random Recycling strategy may result in slightly better performance, because in this strategy every word is presented at least once in every practice phase. A comparison of the Random Recycling graphs of Experiment 2 and 3 (see the left-hand graphs in Figure 32, 33, 35, and 37) shows that the subjects in Experiment 3 performed much better than the subjects in Experiment 2, in exactly the same experimental condition. A possible explanation is that the subjects in Experiment 2 were high school students, while the subjects in Experiment 3 had a higher education. So, a possible explanation of the lack of effect of the Situated strategy in this experiment is that the subjects performed too well, not giving the Situated strategy the time it needed to adapt.

2. *Incorrect transfers to the good set.* There are two main causes for transferring an item to the good set when the student does not really know the correct translation.

In the first place, in the recognition task, there is a guessing probability: the student may press the right answer button though he or she does not know the correct answer. Initially this guessing probability equals 1/30, and it becomes higher the more correct translations the student knows. This guessing probability implies that it is possible that words will become members of the good set though the student has never known their correct translation. Especially initially, when the number of elements in the bad set is relatively high, it will take quite a long time before such a word is presented again and the mistake can be remediated. In Experiment 2, in which the memory of the Situated strategy was cleared after each test phase, this effect could not occur. The effect is especially large and problematic when the students learn very fast, which is, as has already been argued, the case for the subjects in this experiment. Of course, this effect cannot occur in the recall variant, hence it does not explain the disadvantage of the Situated strategy in this variant of the experiment.

In the second place, the student may also translate a word correctly, because the translation is still in the short-term memory. When a word is presented only a short time after it has been presented before, the student may answer correctly because of the very recent exposure to the correct translation, though the translation is not remembered in the longer run. We will analyse the data of the experiment in order to find out whether this occurs frequently.

4. *Incorrect transfers to the bad set.* The student may accidentally press the wrong button, though he or she knows the correct translation. It is not expected that this will happen very often. Nevertheless, in combination with the first point, it would be more problematic for fast learning subjects.

5. *Incorrect stays in the good set.* Students may forget. The Situated strategy deals with forgetting by having parameter k not too large. It could, however, be that forgetting is more frequent than expected. However, the learning curves of the subjects indicate otherwise.

In order to discover which of these reasons (or combination of reasons) are the case, some further analysis was carried out on the data, especially with respect to incorrect transfers to the good set and the role of short-term memory in this. For the sake of simplicity, we have restricted ourselves to the recognition variant of the experiment.

4.3.4 Results: short term memory

First, an analysis was carried out to determine how frequently incorrect transfers to the good set occur. The percentage of items belonging to the good set but answered incorrectly was determined per subject, per test phase (see the left-hand graph in Figure 38). In the right-hand graph in Figure 38 the average number of items belonging to the good set are shown per test phase, divided into those incorrectly answered (black) and those correctly answered (white). A regression analysis was performed per test phase on the number of incorrect responses on items from the good set as a function of the total number of items from the good set. The results of this analysis are summarized in Table 18.

Table 18: *Results of the regression analysis on the good set data of Experiment 3a.The value of the parameter indicates the percentage of items from the good set that were answered incorrectly.*

Source	Parameter (%)	Num DF	Den DF	F
Test 2	32	1	1	56.81 ***
Test 3	21	1	1	22.69 ***
Test 4	17	1	1	26.94 ***
Test 5	14	1	1	20.00 ***
Test 6	9	1	1	13.07 **
Test 7	6	1	1	7.18 *
Test 8	5	1	1	5.70 *

***p <.05, **p <.01, ***p <.001**

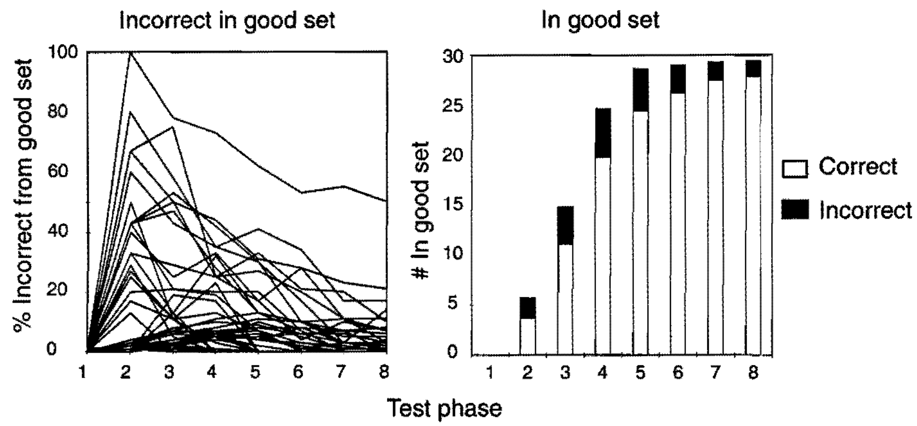


Figure 38: *Results of Experiment 3A (recognition) regarding performance on the good set. Each line in the left-hand graph represents a subject and the average over subjects is shown in the right-hand graph. The test phases are given on the x-axis, on the y-axis of the left-hand graph the percentage of items of the good set that have been answered incorrectly, on the y-axis of the right-hand graph the number of items of the good set that were answered incorrectly (black) or correctly (white).*

Next, a distinction was made between items that were transferred to the good set within three trials after the last presentation of the correct answer, and the other items of the good set. We will call the first items the stm-items, representing short-term memory, and the other items the ltm-items, representing long-term memory. This nomenclature has been chosen because we expect that a correct answer on stm-items has a high probability of being caused by the presence of the answer in the short-term memory, as it has recently been presented. The numbers of stm-items and ltm-items answered correctly and incorrectly were determined per subject, per test phase. The average results are shown in Figure 39.

A MANOVA was performed on the logit-transformed proportions of items answered incorrectly, with the test phase and the kind of items (stm versus ltm) as within-subjects factors. The results of this analysis are summarized in Table 19. There are significant main effects of test phase [$F(7,16)=12.18$, $p < .001$] and kind of item (stm or ltm) [$F(1,22)=15.74$, $p < .001$]. A significant interaction was found between test phase and kind of item [$F(7,16)=3.63$, $p < .05$].

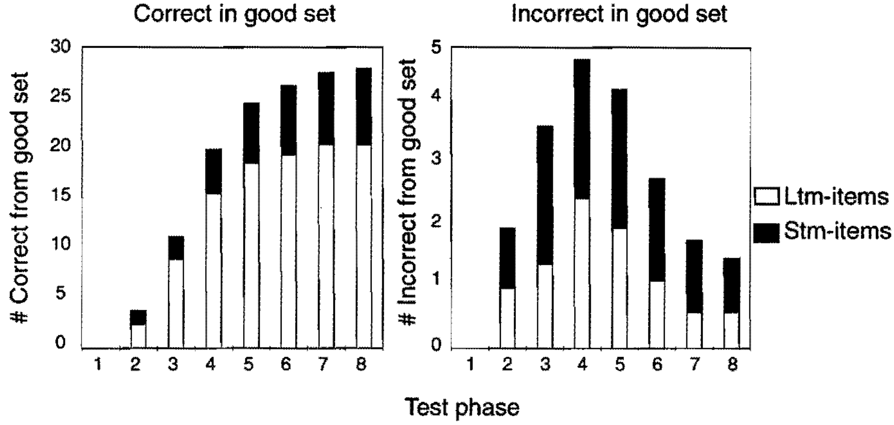


Figure 39: *Results of Experiment 3A (recognition) regarding the correlation between short-term memory and performance on the good set. In both graphs the average over subjects is shown. The test phases are given on the x-axis, on the y-axis of the left graph the number of items of the good set that were answered correctly, on the y-axis of the right graph the number of items of the good set that were answered incorrectly. The stm-items are represented in black and the ltm-items in white.*

Table 19: *Results of the MANOVA on the good set data of Experiment 3A.*

Source	**Num DF**	**Den DF**	**F**
	Within subjects		
Test	7	16	12.18 ***
Kind of item (stm or ltm)	1	22	15.74 ***
Test × Kind of item	7	16	3.63 *

$*p<.05$, $**p<.01$, $***p<.001$

A regression analysis was performed to determine the ratio between the percentage of ltm-items that were answered incorrectly and the percentage of stm-items that were answered incorrectly. The results of this analysis are summarized in Table 20. The ratio found varies between .49 for the second test phase and .3 for the sixth test phase.

***Table 20:** Results of the regression analysis on the good set data of Experiment 3a. The value of the parameter represents the ratio of the percentage of ltm-items that were answered incorrectly and the percentage of stm-items that were answered incorrectly.*

Source	Parameter	Num DF	Den DF	F	
Test 2	0.49	1	1	16.80	***
Test 3	0.33	1	1	43.92	***
Test 4	0.46	1	1	49.23	***
Test 5	0.41	1	1	101.62	***
Test 6	0.30	1	1	33.16	***
Test 7	0.33	1	1	36.04	***
Test 8	0.38	1	1	63.41	***

$^{*}p<.05$, $^{**}p<.01$, $^{***}p<.001$

4.3.5 Discussion: short-term memory

Figure 38 clearly indicates that items from the good set are frequently answered incorrectly. The results of the regression analysis confirm this: for all test phases, at least five percent of the items of the good set were answered incorrectly, with high percentages of 32, 21, 17, and 14 on the second to the fifth test phase. Apparently, incorrect transfers to the good set occur frequently. The left-hand graph in Figure 38 indicates that they occur more frequently for some subjects.

Figure 39 supports our hypothesis that there is a connection between incorrect transfers to the good set and short term memory. A very high proportion of the stm-items were answered incorrectly. The significant main effect on the MANOVA of the kind of item confirms that this proportion is significantly higher than that of the ltm-items. The regression analysis gives an indication of how much higher this proportion is compared to that of the ltm-items: the ltm-items only show at most half, but mostly even only one third, as many incorrect answers as the stm-items.

This leads to the conclusion that the Situated strategy should be changed by adding the restriction that an item may only be presented again when it has not been presented very recently, e.g. within the space of three trials.

4.4 Conclusions

The models' predictions of a positive effect of using the Situated item sequencing strategy have been empirically tested in both recall and recognition tasks to establish whether they were valid. In the first experiment, in a recall task, the Situated strategy reduced the variance between subjects, improving the results of the poor performers. This was due to faster learning of the more difficult items, as they were presented more frequently, without obstructing the learning of the easier items. In complete contrast to the predictions of the models, the Situated strategy produced a steeper initial rise of the learning curves than Random Recycling.

A comparison of the fits and predictions of the models with the experimental data showed two major shortcomings of the models. In the first place, all the models discussed produced steeper starts of the learning curves than the real subjects. In the second place, all the models largely underestimated the effect of VIP queuing, which suggests that the models lack a mechanism simulating the limited cognitive load a student can handle.

As far as the models are concerned, these shortcomings point in the direction of using neural networks, which, like the real subjects, have a tendency to start slowly, and of including in the models a limit to the maximum number of items they can learn at the same time.

As far as the Situated item sequencing strategy is concerned, this idea of a limited capacity argues in favour of adding an extra set of items, which represents the non-presented items, and to presenting one of these items only whenever the number of items in the bad set is below a certain limit.

Successive experiments, in both a recognition and a recall task, showed that the Situated strategy already had an advantage when its memory was cleared after each test phase, hence with only a limited opportunity to adapt to the student. However, in the last experiments, when the memory of the strategy was no longer cleared, the learning curves of the Situated strategy and Random Recycling almost coincided. A likely explanation is that the subjects in those experiments performed too well, not giving the Situated strategy time to adapt.

Another reason why the Situated strategy did not perform as well as expected might be that items were incorrectly transferred to the good set. An extra analysis revealed that incorrect transfers occurred regularly, especially of items that were answered correctly when the correct answer had been presented only a few trials before and could therefore still be in the short-term memory. This led to the conclusion that the Situated strategy should be changed by adding the restriction that an item may only be presented again when it has not been presented very recently, e.g. within the space of three trials.

An analysis of the occurrence of session effects in the experiments gave the impression that the choice of item sequencing strategy seemed to have an impact on the motivation of the subjects: subjects who had experience with the Situated strategy seemed demotivated when confronted successively with

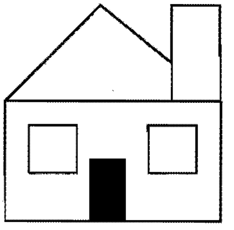
Random Recycling, while subjects who had experience with Random Recycling seemed to be additionally motivated when confronted successively with the Situated strategy. So, the use of the Situated strategy may have a motivating effect on the students.

References

Lave, J. (1988). *Cognition in practice.* Cambridge: Cambridge University Press

Uit den Boogaart, P.C. (ed.) (1975). *Woord frequenties in geschreven en gesproken Nederlands.* Utrecht: Oosthoek, Scheltema & Holkema

Chapter 5



The Practice Agent in a concept learning task

Abstract

In paired associates learning, a subject has to learn and remember the response to each individual stimulus. In concept learning, a response has to be learned for a group of stimuli, often such that the response can be generalized to stimuli the student has not seen before. The subjects learn an underlying classification rule or a property that holds for the group as a whole. This difference with paired associated learning may have implications for the effectiveness of the item sequencing strategy of the Practice Agent as discussed in Chapter 3. It will, however, be shown how the item sequencing strategy can be effectively used in a concept learning task.

5.1 Concept learning

There are more kinds of learning than paired associates learning. One of them is concept learning. The learning of concepts is very important, because it is a way to reduce information and to generalize on the basis of it. In this chapter, we explore whether the sequencing strategy as discussed in the previous two chapters can also be used to support this kind of learning.

5.1.1 Concepts

There are two main views in psychology about what concepts are. According to the first view, concepts are based on formal categories that are definable in terms of properties common to all members of a category and are both necessary and sufficient for category membership (Katz & Postal, 1964). According to the second view, concepts are based on "fuzzy" categories that are better characterized by family resemblance than by sets of necessary and sufficient properties (Rosch & Mervis, 1975). In the latter case, individual exemplars may vary in the number of characteristic properties they possess and, hence, some exemplars may be more typical of a concept than others.

A principle common to both views is that exemplars within a category are more similar to one another than to exemplars from alternative categories. Similarity is viewed as an organizing principle by which concepts are formed and generalizations are made. There are basically two approaches to similarity.

In the first approach, a geometric model is used in which stimuli are represented as points in a multi-dimensional space such that dissimilarities between stimuli correspond to metric distances between the respective points. According to Shepard (1987), the best approximation for this psychological space is a Euclidian distance metric for unitary stimuli, such as colours differing in brightness and saturation, and a city-block metric for analysable stimuli, such as shapes differing in size and orientation. There are two assumptions underlying the geometric model: that stimuli can be represented as combinations of values on different dimensions, and that dissimilarity behaves like a metric distance function. Both assumptions have been questioned by Tversky (1977).

In the second approach, similarity between stimuli is assessed as a comparison of features rather than as the computation of a metric distance between points. Similarity between two stimuli is a combination of an increasing function of the common features and a decreasing function of the distinctive features. An example of this approach is the contrast model of Tversky (1977).

In most concept learning models, the categorization of a stimulus depends on the similarity of that stimulus to the representation (e.g., exemplars) of the different categories (Medin & Schaffer, 1978; Nosofsky, 1986; Gluck & Bower, 1988b; Kruschke, 1992; Pearce, 1994). However, there are also some theoretical and experimental arguments against similarity as a basis for categorization (Nosofsky, 1992; Goldstone, 1994).

Firstly, similarity is viewed as a vague and meaningless concept. Stimuli may possess infinitely many features and, even in a controlled experimental situation, subjects may use other dimensions for the categorization than the intended ones. Also, similarity, and more particularly the relative importance of features, often depends on context.

Secondly, similarity is viewed as being situated on the level of perception, while categorization depends more on foreknowledge, culture, goals, and other higher level factors (Gelman & Markman, 1986; Vandierendonck, 1993).

Thirdly, categorization is viewed as being based on information regarding the whole category, such as within-category variation, rather than on the features of the individual items.

5.1.2 Learning strategies

A distinction is often made between two learning strategies of subjects in a concept learning task: an analytical strategy and a non-analytical strategy (Medin & Smith, 1981). Using the analytical strategy, subjects categorize on the basis of rules according to the occurrence in the stimulus of relevant features. Using the non-analytical strategy, subjects categorize on the basis of the appearance of the stimulus as a whole, using merely the global similarity between the stimulus and the earlier stimuli. It seems possible to influence the choice of learning strategy with the instructions given to the subjects (Elio & Anderson, 1984; Roberts & MacLeod, 1995).

The usual way to test the kind of learning strategy used by the subjects is as follows. First, the subjects are presented with a training phase in which, depending on the instructions, they are supposed to train in applying the rule or to discover the category structure. Next, a transfer phase is presented in which both trained and new stimuli are presented for classification (Elio & Anderson, 1984; Regehr & Brooks, 1993; Livingston & Andrews, 1995). The new stimuli, also called transfer stimuli, can be divided into two kinds: (1) good transfers that appear similar to some of the training stimuli and also belong to the same category as those stimuli, and (2) bad transfers that also appear similar to some of the training stimuli but according to the rule belong to another category. Subjects using an analytic strategy are expected to perform equally well and equally fast on the good and on the bad transfers. Subjects using a non-analytic strategy are expected to perform worse and slower on the bad transfers than on the good transfers.This has been confirmed by the study of Regehr and Brooks (1993).

Elio and Anderson (1984) found an interaction effect of learning strategy and the moment of exposure to total category variation. Subjects using a non-analytic strategy performed better when they were progressively exposed to the category variation. Subjects using an analytic strategy performed better when they were immediately exposed to the total category variation.

For the analytic learning strategy, Livingston and Andrews (1995) found that the salience of features was important for response selection, and that the subjects changed incorrect hypotheses gradually by paying attention to other features.

5.1.3 Relevance of item sequencing

In concept learning, the subjects learn an underlying classification rule or a tendency that holds for the set as a whole. This change of focus from individual items to a set of items may lead to the question of how relevant item sequencing is in the case of concept learning.

For several reasons the frequency of presentation of individual items is still important for learning a concept. In the first place, increasing the frequency of an item increases the accuracy of classification of both the item itself and similar items belonging to the same category, but decreases the accuracy of classification of similar items belonging to another category (Nosofsky, 1988; Nosofsky & Kruschke, 1992). It has been suggested that frequency also influences similarity, in the sense that increasing the frequency of an item may increase the perceptual differentiation in the region of that item (Nosofsky, 1986).

In the second place, increasing the frequency of an item increases the typicality of that item and of similar items belonging to the same category, but decreases the typicality of similar items belonging to another category (Nosofsky, 1988). So, the frequency distribution over the items may influence the formation of hypotheses regarding the classification rules.

In the third place, the frequency of presentation influences the order in which items are presented and thereby the exposure to category variation. As already mentioned above, this has an effect on learning (Elio & Anderson, 1984).

These three consequences of frequency imply that increasing the frequency of presentation of an item not only has an effect on the learning of that item, but also on the learning of other items. This is in contrast to the case of paired associates learning, and may have a major impact on the effect of the Situated item sequencing strategy.

5.2 Models of concept learning

There are mainly three types of models of concept learning: rule models, prototype models and exemplar models. According to rule models, subjects construct abstract classification rules and classify items on the basis of these rules (e.g., Ashby & Gott, 1988; Vandierendonck, 1995). According to prototype models, subjects store an abstract summary representation of a category in their memory and make classifications on the basis of the similarity of items to the abstracted prototype (e.g., Posner & Keele, 1968). By contrast, according to exemplar models, subjects store the individual exemplars of a category in their memory and make classification decisions on the basis of the similarity of items to these exemplars (e.g., Medin & Schaffer, 1978; Kruschke, 1992). Currently, there is a tendency towards the use of exemplar models, as the empirical evidence for these models grows (see, e.g, Nosofsky, Kruschke, & McKinley, 1992). Two specific versions of these models are discussed below.

5.2.1 Alcove

ALCOVE (attention learning covering map) (Kruschke, 1992) is a feed-forward connectionist model that combines the exemplar-based representational assumptions of Nosofsky's (1986) generalized context model and the error-driven learning assumptions of Gluck and Bower's (1988a,1988b) network models. There is considerable empirical support for this model (Nosofsky, Kruschke, & McKinley, 1992; Kruschke, 1992, 1993; Friedman, Massaro, Kitzis, & Cohen, 1995).

Description of the model

The model consists of three layers.

The input units represent the stimulus, with each input unit encoding a psychological dimension. An attention strength is associated with each input unit that reflects the relevance of that dimension for the categorization task. These attention strengths are learned during the training of the network. Each input unit is connected to all hidden units.

The hidden units correspond to positions in the multidimensional stimulus space. The simple version of ALCOVE we have used in the simulations uses a hidden node for each training exemplar. Each hidden unit is connected to all output units. A hidden unit is activated according to the similarity of the stimulus to the exemplar corresponding to that hidden unit. The similarity function is based on Shepard (1987) and in its simplest form it implies an activation of the hidden unit according to the equation:

$$a_h = \exp\left[(-c)\left(\sum_i \alpha_i |h_i - a_i|\right)\right]$$

where a_h represents the activation of the hidden unit, the a_i 's represent the activations of the input units, α_i the corresponding attention strength of the input units, and h_i the value of the exemplar corresponding to the hidden unit on dimension i. Positive constant c is a parameter of the model called the specificity of the unit. This parameter can be interpreted as indicating the overall cognitive discriminability or memorability of the corresponding exemplar: the larger the specificity the more rapidly similarity decreases. We have used the same specificity parameter for all hidden units.

The output units encode the degree to which the alternative response categories are activated. Weights are associated with the connections between hidden units and output units. The activation of an output unit is the sum of the activation of all the hidden units, modulated by the weights between the output unit and the hidden units. The weights are adjusted during training by an error-driven learning rule.

Category activations are mapped onto response probabilities by a Luce (1963) choice rule:

$$P(k_r) = \exp(\phi a_k) / \sum_m \exp(\phi a_m)$$

where $P(k_r)$ denotes the probability of classifying a stimulus in category k, a_k represents the activation of the output unit associated with category k, the a_m 's represent the activations of the output units, and parameter ϕ is a response mapping constant.

During training, the attention strengths and weights are changed by a small amount such that the error decreases. Two parameters, or learning rates, are involved: λ_a for the attention strengths and λ_w for the weights. The larger the learning rate, the faster the attention strengths or weights change. Exact formulas are given in Kruschke (1992).

Because the activation of the hidden units is based on a notion of similarity, presentation of a certain stimulus will not only lead to adjusting the weights of the connections of the corresponding hidden unit, but also to adjusting the weights of the connections of hidden units that correspond to similar exemplars.

Important property of the model

The attention strengths, which decrease for irrelevant dimensions and increase for relevant dimensions, explain why subjects perform better in concept learning tasks in which only one dimension is relevant for categorization than when more dimensions are relevant (Nosofsky & Kruschke, 1992).

5.2.2 Configural Cue

Description of the model
Pearce's version of a configural cue model (Pearce, 1987, 1994) is a connectionist model with three layers: an input layer, a hidden layer called the configural layer, and an output layer.

The input units represent the stimulus, with each input unit encoding a psychological dimension. The activation of an input unit is either zero or $1/\sqrt{n}$, where n represents the number of input units active for that stimulus. Each input unit is connected to all configural units.

The configural units correspond to training exemplars. Weights are associated with the connections between input units and configural units. Initially all the weights are zero. On the first presentation of a stimulus, the weights associated with the connections between the active input units and the configural unit corresponding to the stimulus are permanently set to $1/\sqrt{n}$, where n represents the number of active input units.

The activation of a configural unit is the sum of the activation of all the input units, weighted by the weights between the configural unit and the input units. This implies that the activation of the configural unit corresponding to the presented stimulus equals one. It also implies that when a stimulus y is presented to the network the activation a_x of the configural unit corresponding to a stimulus x meets the equation:

$$a_x = n_{xy}\frac{1}{\sqrt{n_x}}\frac{1}{\sqrt{n_y}}$$

where n_x and n_y represent the number of input units active in the encoding of stimulus x and stimulus y, respectively, and n_{xy} represents the number of input nodes both active in the encoding of stimulus x and y. So, the activation of a configural unit depends on the similarity between the exemplar corresponding to that unit and the stimulus presented.

The output units encode the degree to which the alternative response categories are activated. Weights are associated with the connections between configural units and output units. The activation of an output unit is the sum of the activation of all the configural units, modulated by the weights between the output unit and the configural units.

For the fits of the Configural Cue model on the data of the experiment below we have used the same choice rule as in the case of ALCOVE (see above) to map category activations onto response probabilities. In this process, one parameter is used, namely a response mapping constant ϕ.

Error-driven feedback is used to adjust the weights between the maximally activated configural unit and the output units. An important difference between this model and ALCOVE is that in the latter the weights for all configural units are adjusted, and not only the weights for the highest activated configural unit. Two parameters or learning rates are involved in the adjustment of the weights: a parameter α that reflects the conditionability of the configural units and a parameter β that reflects properties of the output units. The larger the learning rates, the faster the weights change. Exact formulas are given in Pearce (1994). According to Pearce (1994), the value of α is usually one. So, there is only one free parameter in the model: learning rate β.

5.3 Development of the stimulus material

Stimulus material has been developed to meet the requirements of (1) readily discriminable and describable feature dimensions, (2) discrete and easily describable values on the dimensions, and (3) equally distributed values on the dimensions across the stimulus set (Regehr & Brooks, 1993).

Artificial but still familiar stimuli have been chosen in the form of houses (see Figure 40).

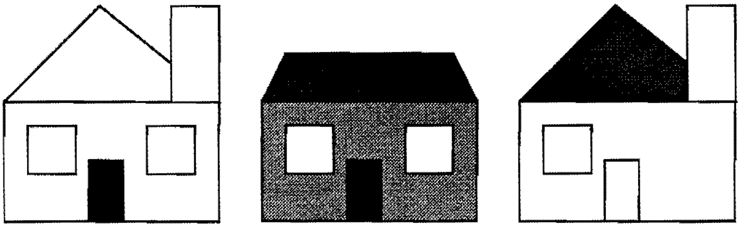


Figure 40: *Examples of stimuli used in Experiment 1*

The houses were constructed on the basis of six dimensions that can each have two values. The two values of the dimensions are coded by 0 and 1, so the stimulus is coded by a binary string of length six. The following dimensions were used: colour of the roof (0 = white, 1 = dark grey), colour of the facade (0 = white, 1 = grey), number of windows (0 = two, 1 = one), presence of a chimney (0 = no, 1 = yes), colour of the door (0 = white, 1 = black), and shape of the roof (0 = pyramidal, 1 = flat).

For the experiment in which the effectiveness of the Situated strategy in a concept learning task will be evaluated, we wanted approximately 30 stimuli for training, because that number had been used in the experiments on paired associates learning. Approximately 10 extra stimuli were needed for testing the transfer effect of the training. For that reason, six dimensions were needed: on the basis of six dimensions 64 houses can be constructed. Of these 64 houses, 40 houses have been selected based on a category structure as explained in Section 5.1. The initial selection is shown in Table 21.

Table 21: *Stimuli used in studies 1 and 2, and category structure used in Experiment 1. Stimuli are encoded by their value on the dimensions, with a=colour of the roof, b=colour of the facade, c=number of windows, d=presence of a chimney, e=colour of the door, and f=shape of the roof. Good transfer stimuli are indicated with G, bad transfer stimuli with B.*

Category A			Category B			
a b c d e f	a b c d e f		a b c d e f		a b c d e f	
0 0 0 1 1 1	0 0 0 1 0 0	G	0 0 1 1 1 1		0 0 1 1 0 0	
0 0 0 1 1 0	0 0 0 0 1 1	B	0 0 1 1 1 0		0 0 1 0 1 1	
0 0 0 1 0 1	0 0 0 0 1 0		0 0 1 1 0 1	B	0 0 1 0 1 0	G
0 0 0 0 0 1	0 0 0 0 0 0		0 0 1 0 0 1		0 0 1 0 0 0	
1 1 1 1 1 1	1 1 1 0 1 1	B	1 1 0 1 1 1		1 1 0 0 1 1	
1 1 1 1 1 0	1 1 1 0 1 0	G	1 1 0 1 1 0		1 1 0 0 1 0	G
1 1 1 1 0 1	1 1 1 0 0 1		1 1 0 1 0 1		1 1 0 0 0 1	B
1 1 1 1 0 0	1 1 1 0 0 0		1 1 0 1 0 0		1 1 0 0 0 0	
1 0 1 1 1 1	1 0 1 0 1 1		1 0 1 1 1 0		1 0 1 0 1 0	
1 0 1 1 0 1	1 0 1 0 0 1	G	1 0 1 1 0 0		1 0 1 0 0 0	G

5.3.1 Exploratory study 1: Perception of stimulus dimensions

The aim of this study was to discover if the analytic dimensions that we think are apparent in the stimuli are also the ones used by the subjects.

Method

Procedure. A similar procedure to that of Regehr and Brooks (1993, Experiment 1a) was used. Each of the 40 stimuli was placed on a separate 10 × 10 cm file card. The cards were laid in front of a subject in a random arrangement and the subject was told the following[1]:

> The 40 houses in front of you can be divided into two groups. If you were told that you had to figure out how the houses divide into the two categories, what would you do? What would you test as potential reasons for dividing the houses? Please name anything you can think of.

Subjects' responses were recorded. While responding, subjects were repeatedly prompted to name any additional methods they could think of for dividing the houses. The study took approximately ten minutes per subject. From

1. A Dutch version of these instruction was used.

the recordings, the number of dimensions mentioned by the subjects were counted.

Subjects. Eight subjects with university or higher vocational training participated voluntarily in the study. The average age was 23.

Materials. The 40 houses from Table 21 were used as stimuli. The width of the houses was 58mm, the height 45mm for houses with a flat roof and 57mm for houses with a pyramidal roof.

Results

Of the eight subjects, four identified all six dimensions of variation from the analytic structure. Two subjects failed to identify the colour of the roof as a potential dimension, two subjects failed to identify the shape of the roof, and one failed to identify the colour of the door. Five of the eight subject mentioned colour as a first dimension, two subjects mentioned the shape of the roof first, and one subject mentioned the presence of a chimney first.

Discussion

On the basis of these results, we can conclude that all dimensions are readily discriminable and describable. A possible explanation for the fact that colour of the roof was not mentioned by two subjects is that the colour of the roof changed independently of the colour of the facade in only a small number of the houses, namely 20%: all the other houses were either completely white or completely dark. The large proportion of subjects that started with colour as a dimension indicates that this is a salient feature. We will use these observations in determining the category structures and in the attention strengths in the simulations with ALCOVE.

5.3.2 Exploratory study 2: Salience of stimulus dimensions

The aim of this study was to discover which of the analytic dimensions is the most salient in the sense that it is the most obvious one subjects would use to classify houses. A spontaneous classification task was used, because in such a task subjects tend to focus on a single analytic dimension when forming categories (Medin, Wattenmaker, & Hampson, 1987; Regehr & Brooks, 1993, 1995).

Method

Procedure. A similar procedure to that of Regehr and Brooks (1993, Experiment 1b) and Medin et al. (1987) was used. Subjects received the 40 stimuli on file cards in a random order. They were asked to lay out the stimuli, look them over carefully, and place the houses into two categories in any way that seemed appropriate. No restriction was placed on the number of houses in each category. Following the categorizations, subjects were asked to describe the criteria by which they classified the stimuli. Next, they were asked why they had used these criteria. The study took approximately fifteen minutes per subject.

Subjects. Eight subjects with university or higher vocational training participated voluntarily in the study. The average age was 23. No subject had any prior experience with the stimuli.

Materials. The same stimuli were used as in exploratory study 1, namely the 40 houses from Table 21.

Results

Seven of the eight subjects reported using a single analytic dimension to classify the houses into the two categories. Six of these subjects divided the houses on the basis of the shape of the roof; the remaining subject used the presence of the chimney. All the subjects who made the categorization on the basis of one dimension reported as reason for using that dimension that it was the most salient, or presented the largest difference between the houses. One subject reported that colour also had been considered as a basis for categorization.

One subject used three dimensions for the categorization: the colour of the roof, colour of the facade, and colour of the door. This subject reported that an intuitive rule based on the amount of colour in the house as a whole has been used.

Discussion

On the basis of these results, we can conclude that the shape of the roof is a very salient dimension, as it is used by almost all subjects. Therefore, a classification rule based on only that dimension would probably be relatively easy to learn. This kind of information will be used in determining the category structures.

The use of three dimensions by one subject can be interpreted as the use of a single overall colour dimension. This illustrates that it is still possible that subjects will use dimensions in the categorization that do not correspond to the analytical dimensions. This cannot, however, be prevented.

5.3.3 Conclusions

From these studies, we conclude that the analytic dimensions are readily discriminable, with the colour of the facade and shape of the roof as the most salient dimensions. As there are no profound reasons to change the stimulus set, the same set will be used in the concept learning task.

5.4 A quest for gradually increasing learning curves

For the concept learning task, the set of stimuli had to be divided into two equally large sets, called category A and B. It was decided to use a classification rule that assigns houses unambiguously to one of the two categories. A possible categorization is shown in Table 21, with the stimuli of category A on the left, and the stimuli of category B on the right.

The kind of classification rules considered were disjunctions of conjunctions. This means that in order to belong to a certain category a stimulus should meet one of the requirements of that category, the requirement being that the stimulus should have a several features, i.e., certain values on certain dimensions. For instance, for the category structure as shown in Table 21, a house belongs to category A whenever it has

a white roof, a white facade, and two windows (stimuli starting with 000),
or a gray roof, a gray facade, and one window (stimuli starting with 111),
or a gray roof, a white facade, one window, and a pyramidal roof (stimuli starting with 101 and ending with 1).

In the remainder of this chapter, this kind of classification rule will be denoted by A = 000xxx + 111xxx + 101xx1. So, in the case of the category structure of Table 21, category B is given by the classification rule: B = 001xxx + 110xxx + 101xx0.

The problem addressed in this section is which classification rules should be used when comparing item sequencing strategies, as has already been done in the previous chapter for paired-associates learning. The classification rules should be neither too easy nor too difficult: when subjects learn very fast, the Situated strategy has no time to adapt; when subjects hardly learn at all, the Situated strategy has nothing to adapt to. In both cases, no effect of strategy is to be expected. So, the main goal of the pilot studies is to construct classification rules that result in gradually increasing learning curves.

Though both Random Recycling and Situated will be used as item sequencing strategies in the pilot studies, there is no intent to compare both strategies already. Therefore, no statistics will be presented, but the focus will be merely on the learning curves. In addition, the transfer test will be used merely to test the complete experimental procedure, and the results of this test will not be discussed.

5.4.1 Pilot study 1: Classification rules of three disjunctions

Method

Design and Procedure. Comparisons were made between groups of subjects learning a categorization of houses in different experimental conditions. In each condition, the subjects were presented with alternate test and practice phases. In both phases, houses were presented on the screen (see Figure 41) with the request to select the correct response category as accurately and quickly as possible. In the instructions no indication was given with respect to the existence of an underlying rule: the only kind of instruction given was that there were two sets of houses and that the subjects had to learn to categorize the houses.

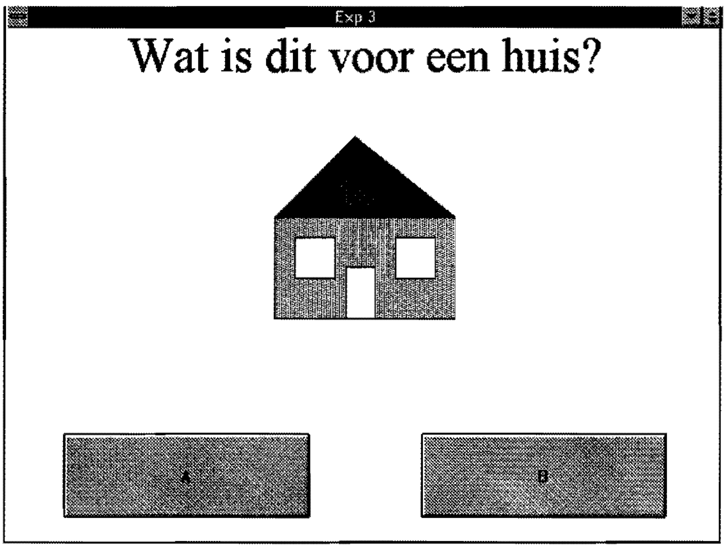


Figure 41: *Screen layout of the concept learning experiments. The text reads: "What kind of house is this?".*

Subjects selected a response category by pressing on the keyboard the z-key for category A and the /-key for category B. On the screen (see Figure 41) two buttons were displayed for the categories corresponding to the left-right orientation of the keyboard keys used. Answers and response times were recorded. In test phases, all 30 houses were presented to the subjects in a random order. No feedback was given with respect to the correctness of the response. The first test phase enabled the subject to get acquainted with the kind of stimuli used in the experiment.

In practice phases, depending on the condition assigned to the subject, Random recycling or Situated was used to determine the sequence of 30 houses presented to the subjects. So, depending on the condition, it could occur that in a practice phase the same house was presented more than once, and other houses not at all. Feedback was given with respect to the correctness of the response, and the correct response was displayed on the screen for 3 seconds.

The experiment ended with a transfer test after sixteen test phases or a score of 100% correct on two subsequent test phases. The maximal number of test phases was determined in a pilot study in such a way that the experiment took no more than an hour. The transfer test was similar to a regular test phase, except that in addition to the 30 houses practised, 10 transfer houses, with which the subject had no experience were presented as well.

As good transfers, stimuli were chosen that differed from some other stimuli of the same category on only one dimension which was irrelevant for the classification. As bad transfers, stimuli were chosen that differed from some other stimuli of the other category on only one dimension which was relevant for the classification. For instance, in the category structure of Table 21, stimulus 000100 of category A is a good transfer, as it differs on only one dimension from stimulus 000101which also belongs to category A. Stimulus 000011 of category A is a bad transfer as it differs on only one dimension from stimulus 001011 of category B.

Subjects. Twelve university students participated voluntarily in the pilot study. The average age was 21. Subjects were randomly assigned to one of the two experimental conditions. Five subjects were assigned to the Situated condition, seven to Random Recycling.[1]

Materials. The same stimuli were used as in the exploratory studies, namely the 40 houses from Table 21. The same category structure as in Table 21 was used. So, A = 000xxx + 111xxx + 101xx1, and
B = 001xxx + 110xxx + 101xx0. The houses indicated in the table with a G or B were not used in the ordinary test or practice phases, but as the additional houses for the transfer test.

Results and Discussion

The results of the pilot study are shown in Figure 42. A substantial number of the subjects remained at a guessing score of around 50% correct during most of the experimental session. Some learning took place only after test 8, but,

1. The unequal distribution over the conditions is due to the fact that two potential subjects failed to turn up.

with the exception of three subjects, this learning was rather limited. All three subjects who did eventually reach scores of above 90% correct were in the Situated condition. The fact that the relatively large increase in scores of these subjects only occurred at a moment when the Random Recycling subjects also showed some, though limited, learning supports the view that an advantage of the Situated strategy can only occur when some learning takes place.

We concluded that the task is too difficult. A probable cause is that the classification rules are too difficult. As subjects learn conjunctions faster than disjunctions (Holland, Holyoak, Nisbett & Thagard, 1987), it may well be that the three disjunctions used in the classification rules were mainly responsible for the difficulty. Therefore, the complexity of the classification rules was reduced by eliminating one of the three disjunctions of each rule.

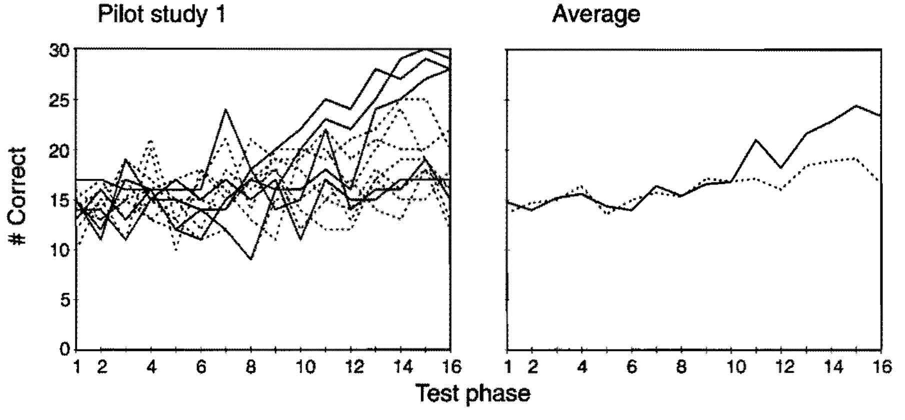


Figure 42: *Results of Pilot study 1. Each line in the left-hand graph represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line) and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

5.4.2 Pilot study 2: Effect of category simplification

Method

Design and Procedure. The same design and procedure were used as in Pilot study 1, except that in this case the number of stimuli used in test and practice phases was restricted to 24, and the number of stimuli in the transfer test was restricted to 32.

Subjects. Twelve university students participated voluntarily in this pilot study. The average age was 21. Subjects were randomly assigned to one of the two experimental conditions.

Materials. The category structure as used in Pilot study 1 was simplified by removing the last disjunctions from the classification rules.
So, A = 000xxx + 111xxx, and B = 001xxx + 110xxx. The houses indicated in Table 21 with a G or B were not used in the ordinary test or practice phases but as the additional houses for the transfer test.

Results and Discussion.
The results of the pilot study are shown in Figure 43. Only two subjects of each condition reached a score of over 90% correct. Half the subjects remained on guessing level. So, contrary to expectations, the simplification of the classification rules did not lead to a large improvement in the average scores. The only apparent effects of the simplification were an acceleration of the moment at which a rise occurred in the learning curves of the few subjects that did learn, and an increase in the steepness of the rise. The latter can be explained by the fact that understanding one additional disjunction of a classification rule leads to a larger additional percentage of correct classifications in case of rules consisting of two disjunctions than in case of rules of three disjunctions.

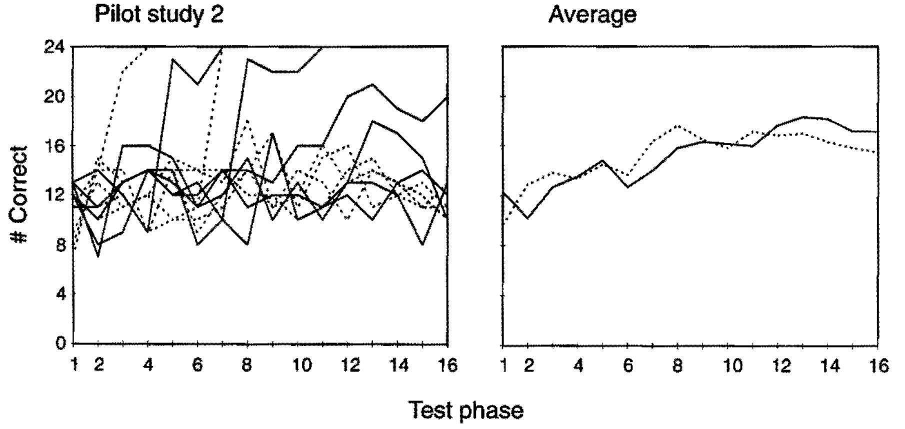


Figure 43: *Results of Pilot study 2. Each line in the left-hand graph represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line) and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

We conclude that the difficulty of the task is not due to the structure of the classification rules. A possible explanation for most subjects staying at a guessing level is that the discovery of a correct disjunction of the classification rules depends on the subject's attention to the colour of the roof and facade, combined with the number of windows. When subjects direct their attention to these dimensions right from the start, they are likely to discover the correct disjunctions very quickly. On the other hand, when subjects direct their attention to irrelevant dimensions like the presence of a chimney, they are unlikely to find the correct disjunctions, especially as subjects tend to maintain their hypothesis even when confronted with evidence that it is incorrect (Klahr & Dunbar, 1988; Livingston & Andrews, 1995).

This reformulates the problem of obtaining increasing learning curves to the problem of ensuring that most subjects direct their attention to the relevant dimensions of the stimuli and neglect the irrelevant ones in an early phase of the experiment.

Livingston and Andrews (1995) showed that individual differences in the sequence of encounters with a set of category exemplars can generate individual differences in hypotheses about category structure, as reflected in both performance measures and feature salience assignments.Thus, the initial sequence in which stimuli are presented is very important for category learning. So far, in the pilot studies, this initial sequence was random in the first test phase, and almost random (though determined by the strategy) in the first practice phase. So, each subject got another initial sequence. This may explain the variation between subjects. Hence, we decided to make the experimental conditions more homogeneous by using the same order of presentation for all subjects in the first test and practice phase. It is expected that this will reduce the variance between subjects.

By choosing a good initial sequence, it may be possible to direct the attention of all subjects to the relevant features. Elio and Anderson (1984) found an effect of the moment of exposure to total category variation on the performance of the subjects. We decided to divide the stimuli into two groups of equal size, one of which has a lower category variation than the whole stimulus set. The first group contained only stimuli with a chimney (top row of Figure 44) and the other group contained mostly stimuli without a chimney (bottom row of Figure 44). A random order was determined for the stimuli in both groups (this is the order of the stimuli in both rows of Figure 44), and the stimuli of the first group were presented before the stimuli of the second group. We expected that this order of presentation would implicitly clarify the irrelevance of the chimney and would also lead to a higher probability in the first practice phase of discovering which other dimensions are irrelevant (because of the diminished category variation within the groups).

5.4.3 Pilot study 3: Effect of a fixed initial sequence

Method

Design and Procedure. The same design and procedure were used as in Pilot study 2, except that in this case the order in which stimuli were presented during the first test and practice phase was determined beforehand and was the same for all subjects. This order is shown in Figure 44.

Subjects. Five university students participated voluntarily in the pilot study. The average age was 22. Subjects were randomly assigned to one of the two experimental conditions: two to Random Recycling and three to Situated.

Materials. The same category structure was used as in Pilot study 2. So, A = 000xxx + 111xxx, and B = 001xxx + 110xxx. The houses indicated in Table 21 with a G or B were not used in the ordinary test or practice phases, but as the additional houses for the transfer test.

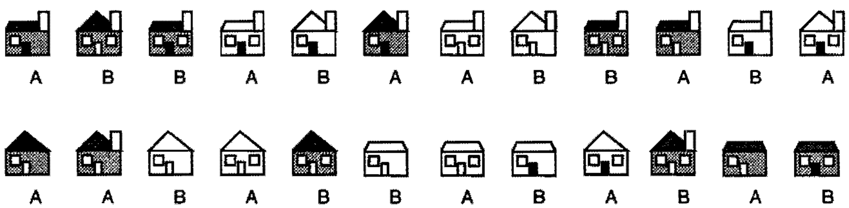


Figure 44: *The order in which the stimuli were presented during the first test and practice phase of Pilot study 3 (from left to right, from top to bottom). The letters below the houses indicate the category to which they belong.*

Results and Discussion

The results of the pilot study are shown in Figure 45. All subjects reached a score of 100% correct, three subjects even reached that score as early as the fifth test phase. Though the number of subjects was very limited, this indicates a positive effect on the performance of the chosen fixed initial sequence. However, this effect seemed too large: a consequence of the fast learning is that the Situated strategy does not get much time to adapt to the subjects.

To reduce the fast learning to some extent, the fixed sequence in the first test and practice phase was altered in such a way that more category variation was presented to the subjects in the first group of stimuli. For that purpose, two stimuli without a chimney from the second group replaced two stimuli of the first group. These stimuli were chosen such that they differed on only one dimension from a stimulus that remained in the first group. Again the order within the groups was determined randomly. The resulting order is shown in Figure 46. The main experiment was performed with these settings.

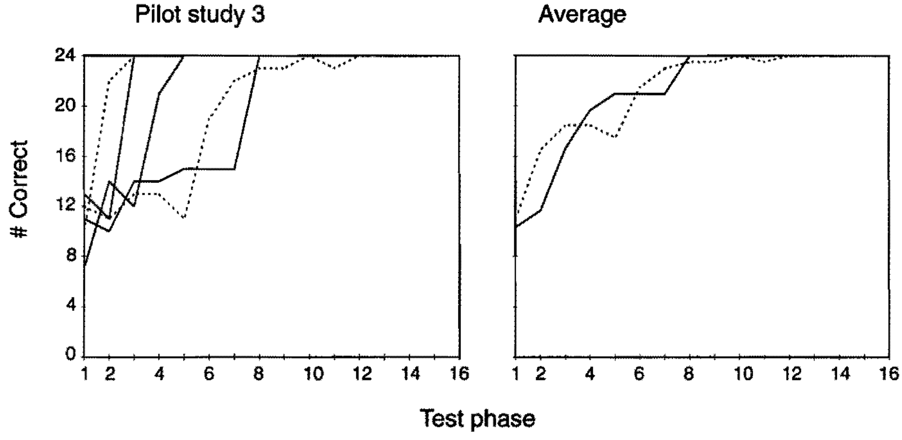


Figure 45: *Results of Pilot study 3. Each line in the left-hand graph represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line) and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

5.5 Experiment 1

5.5.1 Method

Design and Procedure

The same design and procedure were used as in Pilot study 3. Again, the order in which stimuli were presented during the first test and practice phase was determined beforehand and was the same for all subjects. This order differed from the order used in Pilot study 3, and is shown in Figure 46.

Subjects

Twenty-seven university students, with an average age of 22, participated voluntarily in the experiment. None of them had any prior experience with the stimuli and this kind of task. Subjects were randomly assigned to one of the two experimental conditions: 14 to Random Recycling and 13 to Situated.

Equipment

The experiment was run on PC's (486, 14 inch screen) under Windows in a network configuration. In the experiment room, 14 PC's were located, at a distance of at least one meter from each other. The subjects used an ordinary keyboard to respond, use of the mouse was not needed. No audio was used. All instructions and feedback were given in the form of text on the screen.

Materials

The same category structure was used as in Pilot study 2 and 3. So, A = 000xxx + 111xxx, and B = 001xxx + 110xxx. The houses indicated in Table 21 with a G or B were not used in the ordinary test or practice phases but as the additional houses for the transfer test.

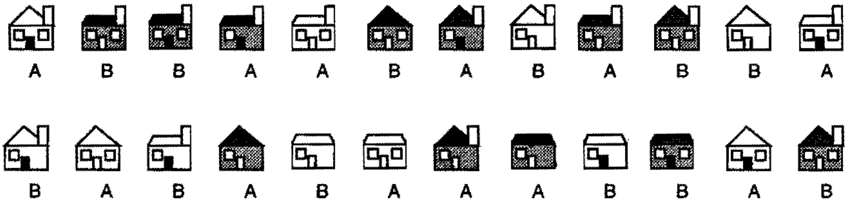


***Figure 46:** The order in which the stimuli were presented during the first test and practice phase of Experiment 1 (from left to right, from top to bottom). The letters below the houses indicate the category to which they belong.*

5.5.2 Results: general

The results of the experiment are shown in Figure 47.

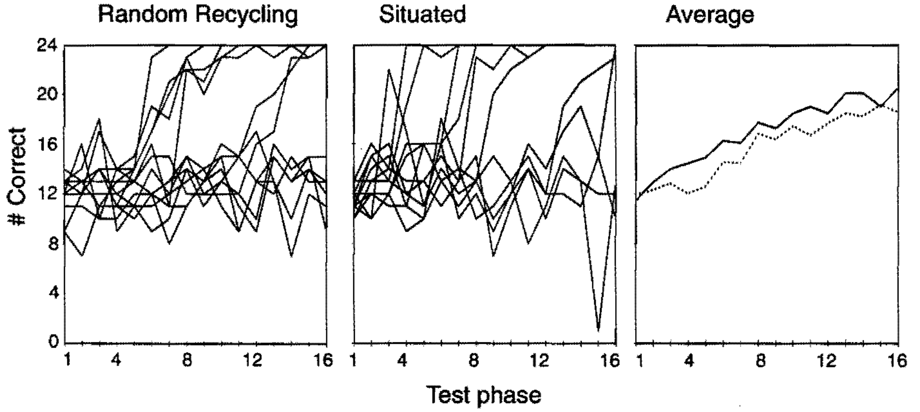


***Figure 47:** Results of Experiment 1. Each line in the two left-hand graphs represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line) and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

The results of three subjects, two in the Situated condition and one in Random Recycling, were not included in the analysis. Two subjects were excluded because they had not participated seriously in the experiment as their response times were extremely low (sometimes even zero). One subject was excluded because he had not completed the experiment.

A MANOVA was performed on logit-transformed proportions of correct responses, with the test phase as a within-subjects repeated measures factor and the experimental condition (strategy used in the practice phases) as a between-subjects factor. When a subject was presented with the transfer test due to a score of 100% on two sequential test phases, a score of 100% correct was assigned to the skipped test phases. The results of the analysis are summarized in Table 22.

Table 22: Results of the MANOVA on the data of Experiment 1.

Source	Num DF	Den DF	F
		Between subjects	
Strategy	1	1	1.11
		Within subjects	
Test	15	8	2.59
Test × Strategy	15	8	0.92

**p* <.05, ***p* <.01, ****p* <.001

The effects of test phase and strategy were not significant. The interaction between test phase and strategy was also not significant.

5.5.3 Discussion: general

As can be seen in Figure 47, a substantial number of the subjects in both conditions remained at a guessing level for most or even the total range of test phases. This explains why there are no significant effects, not even of test phase. We decided to perform a post-hoc analysis to determine whether there is an effect of strategy for the high performers.

5.5.4 Results: high performers versus low performers

A post-hoc analysis was performed in which subjects were divided into two groups per strategy: one group for the high performers and a second group for the low performers. The median per strategy was used as a criterion: if the number of correct responses of a subject in most test phases was above or equal to the median of that test phase, the subject was assigned to the group of high performers. On the basis of this criterion, in both strategies five subjects were assigned to the high performers, and in Random Recycling eight subjects were assigned to the low performers and in Situated six.The learning curves and average learning curves per group and per strategy are shown in Figure 48.

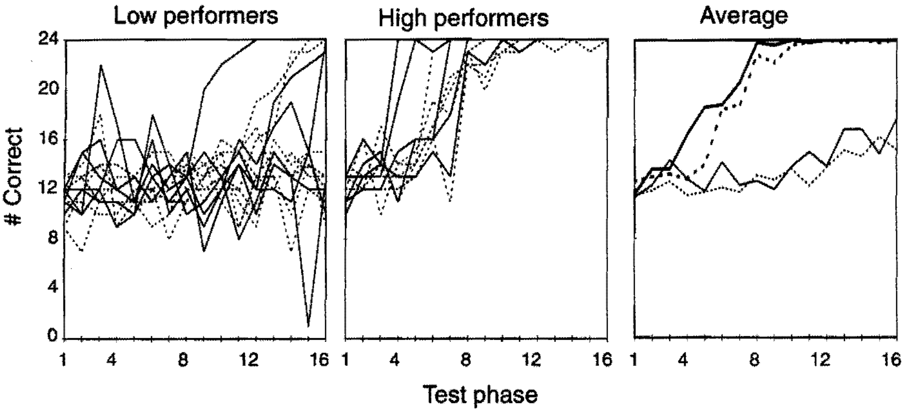


Figure 48: Results of Experiment 1 for low and high performers, respectively. Each line in the two left-hand graphs represents a subject, and each line in the right-hand graph represents the average over subjects for the low performers of Random Recycling (dashed line) and Situated (solid line), and the high performers of Random Recycling (bold dashed line) and Situated (bold solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.

For each group, a MANOVA was performed on logit-transformed proportions of correct responses, with the test phase as a within-subjects factor and the experimental condition (strategy used in the practice phases) as a between-subjects factor. Because of insufficient error degrees of freedom, in both groups only the between-subjects effects are available. In both groups the effect of strategy was not significant.

5.5.5 Discussion: high performers versus low performers

From Figure 48, it can already be induced that there is no difference for the low performers of the strategy used. This is not surprising as the Situated strategy can only have an effect once a student has learned something. So, for subjects who remain at change level no effect can be expected.

Figure 48 seems to indicate an advantage of using the Situated strategy for the high performers, as the rise in the learning curves of the subjects in the Situated condition tends to occur earlier. This is also reflected in the average learning curves. However, due to the small number of high performing subjects and the variance between subjects this effect is not significant and may well be due to coincidence.

In general, the kind of learning curves obtained indicate that the circumstances under which the Situated strategy was tested were not ideal, as a lot of subjects remained on guessing level, which gives the strategy nothing to adapt to, while the other subjects showed very steep rises in the learning curves, which gives the strategy no time to adapt. More gradually increasing learning curves are needed.

5.5.6 Fit of the models

The concept learning models were fitted to the data in the same way as models were fitted to the data in the case of paired associates learning, as described in Section 4.1.8. Each model functioned as an artificial subject, getting the same experimental procedure as the real subjects. For each combination of parameter values, the average χ^2 was calculated over a hundred runs of the model and those parameter values were determined for which this average χ^2 was minimal.

In contrast to the case of paired associates learning, the models were not fitted to each individual subject, but to the average learning curves of the two groups of high performers and low performers in both experimental conditions. This decision was based on the variation of the individual learning curves, which makes it rather senseless to fit to individual learning curves, and the large variance between subjects due to the difference between the high and low performers, which makes it rather senseless to fit to the average learning curves over all subjects in an experimental condition.

Fit of ALCOVE

ALCOVE was fitted to the data of Experiment 1 by using six input units (one for each stimulus dimension), 24 hidden units (one for each training exemplar), and two output units (one for each category). It was assumed that the physical dimensions of the stimuli had corresponding psychological dimensions.

We used the same value of the specificity parameter for all hidden units, and this value was chosen as 6.5, in correspondence with the value used by Kruschke (1992). We tried various other values for this parameter, but it hardly affected the learning curves. The attention strengths and association weights were initialized at 1. The response mapping constant was set at ϕ=2.

As λ_w, the learning rate for the weights, had the largest effect on the learning curves, the value of this parameter was determined in most detail.

The value of λ_a, the learning rate for the attentional strengths, had less impact on the learning curves and was therefore determined more coarsely.

The minimal χ^2, the level of significance, and the parameter values for which this best fit was obtained for each group are shown in Table 23. The learning curves corresponding to these parameter values are shown in Figure 49.

Table 23: *Results of the fit of Alcove on the data of Experiment 1.*

Data of the fit	Parameter values	Min χ^2
Random Recycling: high performers	λ_w=.074 λ_a=.4 α_1=1 α_2=1 α_3=1.33 α_4=1.15 α_5=1.15 α_6=1.16	3.8**
Situated: high performers	λ_w=.143 λ_a=0 α_1=1 α_2=1 α_3=1 α_4=1 α_5=1 α_6=1	3.0***
Random Recycling: low performers	λ_w=.007 λ_a=1 α_1=1.01 α_2=1.01 α_3=1.53 α_4=1.5 α_5=1.5 α_6=1.51	0.9****
Situated: low performers	λ_w=.02 λ_a=.2 α_1=1 α_2=1 α_3=1.29 α_4=1.21 α_5=1.26 α_6=1.27	2.1****

****p* <.05, ***p* <.01, ****p* <.001, *****p* <.0001; 15 degrees of freedom**

A second fit of Alcove was done with varying initial values of the attention strengths. In this way, we tried to capture the initial relative salience of the different dimensions, as discovered in the exploratory studies described in Section 5.3. The attentional strengths α_1, α_2, and α_6, associated with the input units for the colour of the facade, colour of the roof, and shape of the roof, respectively, were initialized at 0.1. The other attentional strengths, α_3, α_4, and α_5, were initialized at 0.033. Given the classification rules used in the experiment, it was expected that Alcove would increase the relative attentional strength of the number of windows, and decrease the relative attentional strength of the shape of the roof. The results of these fits are shown in Table 24 and Figure 49.

Table 24: Results of the fit of Alcove with differing α's on the data of Experiment 1.

Data of the fit	Parameter values	Min χ^2
Random Recycling: high performers	λ_w=.04 λ_a=.9 α_1=1.52 α_2=1.52 α_3=3.66 α_4=1.97 α_5=1.76 α_6=1.79	4.4**
Situated: high performers	λ_w=.039 λ_a=.1 α_1=.51 α_2=.51 α_3=.9 α_4=.29 α_5=.39 α_6=.47	3.8***
Random Recycling: low performers	λ_w=.018 λ_a=0 α_1=.1 α_2=.1 α_3=.033 α_4=.033 α_5=.033 α_6=.1	1.2****
Situated: low performers	λ_w=.012 λ_a=.5 α_1=22.02 α_2=22.02 α_3=44.87 α_4=42.22 α_5=43.13 α_6=34.26	1.7****

****p* <.05, ***p* <.01, ****p* <.001, *****p* <.0001; 15 degrees of freedom**

The resulting values of the attention strengths are also shown in Table 23 and 24. The values of α_1 and α_2 are always equal, as they are initially the same and the values of the corresponding dimensions, namely the colour of the facade and colour of the roof, always change together in the stimuli used in this experiment.

In two cases, namely the high performers of Situated in Table 23 and the low performers of Random Recycling in Table 24, the learning rate of the attention strengths, λ_a, equalled zero, so the values of the attention strengths did not change during learning. For the second case, this is not strange as subjects hardly learn, but for the first case it is surprising if changes in attention strengths are viewed as the way to change hypotheses.

As expected, the value of α_3, corresponding to the number of windows (very relevant for the classification), is relatively high compared to the other attentional strengths for the high performers. This is especially true for the version of Alcove with varying initial attention strengths (see Table 24): it is approximately twice as large as the other ones. This is particularly nice as its value was initially one third of that of α_1, α_2, and α_6.

As expected, the values of the attention strengths α_4, α_5, and α_6,which correspond to the irrelevant dimensions, are relatively similar. The only exception is formed by the high performers of Situated from Table 24: their α_6 is relatively large. Nevertheless, considering that the value of α_6 was initially three times as large as that of the other two, this difference has certainly diminished in the course of learning.

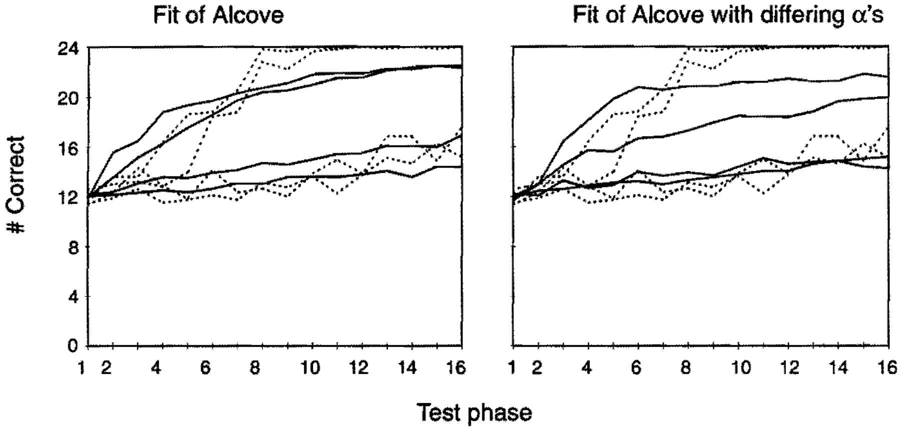


Figure 49: *Fits of Alcove with equal and varying initial values of the attention strengths on the data of Experiment 1. Each line in the graphs represents the average over subjects for the low and high performers of Random Recycling and Situated (dashed lines), respectively, and the fits of Alcove with equal and varying initial values of the attention strengths on these average learning curves (solid lines). The test phases are given on the x-axis and the number of correct responses on the y-axis.*

Fit of the Configural Cue Model

The Configural Cue model was applied to the data of Experiment 1 by using six input units (one for each stimulus dimension), 24 hidden units (one for each training exemplar), and two output units (one for each category). It was assumed that the physical dimensions of the stimuli had corresponding psychological dimensions.

We used a fixed value of parameter α, which reflects the conditionability of the configural units, and this value was chosen as 1 corresponding to the value used by Pearce (1994). The association weights were initialized at 0. The response mapping constant was set at ϕ=2.

The minimal χ^2, the level of significance, and the parameter values for which this best fit was obtained for each group are shown in Table 25. The learning curves corresponding to these parameter values are shown in Figure 50.

Table 25: Results of the fit of Configural cue on the data of Experiment 1.

Data of the fit	Parameter values	Min χ^2
Random Recycling: high performers	β=.09	4.0**
Situated: high performers	β=.197	2.2****
Random Recycling: low performers	β=.007	1.0****
Situated: low performers	β=.021	1.8****

****p* <.05, ***p* <.01, ****p* <.001, *****p* <.0001; 15 degrees of freedom**

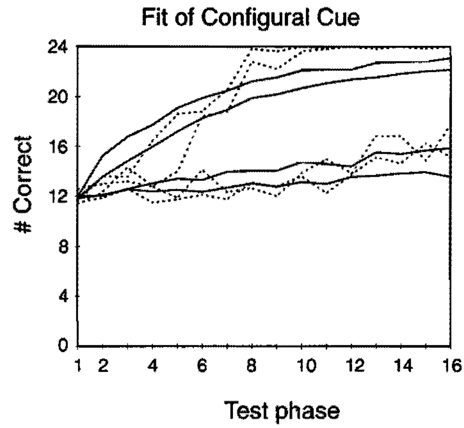


Figure 50: *Fit of Configural Cue on the data of Experiment 1. Each line in the graph represents the average over subjects for the low and high performers of Random Recycling and Situated (dashed lines), respectively, and the fit of Configural Cue on these average learning curves (solid lines). The test phases are given on the x-axis and the number of correct responses on the y-axis.*

5.5.7 Discussion: fit of the models

As shown in Table 23, 24, and 25, both models fit the data of Experiment 1 very well. The extra parameters in the case of Alcove, particularly the attention strengths which are learned during run time, do not, however, result in a better fit of Alcove on the data than Configural Cue. Nor does the use of varying initial attention strengths, which are supposed to reflect differences in the initial salience of the dimensions, result in a better fit compared to the use of equal initial attention strengths.

For both models, the difference between low and high performers is reflected in different values for the learning rates. For Alcove, the learning rate of the association weights rather than that of the attention strengths seems to be important in explaining the variance between the groups of subjects. This also indicates that the use of attention strengths is not as powerful as expected. Nevertheless, the effect of salience on learning clearly indicates a need for something like attention strengths, as will become even clearer in the experiments discussed below.

As shown in Figure 49 and 50, the learning curves produced by both models differ in one major aspect from those produced by the high performance subjects. In the first place, the learning curves of the models tend to rise very early in the learning process, but with a slope that decreases quite fast. The curves of the real subjects rise some test phases later, but with a steeper and more constant slope.

Another defect of the models is that the parameter values found for both strategies differ. For both Alcove with equal initial attention strengths, and for Configural Cue, much higher parameter values were found for the Situated strategy than for Random Recycling: λ_w and β were found at least twice as large for Situated. This suggests that the models would predict a negative effect of using the Situated strategy, which is certainly not reflected in the experimental results.

Strikingly, this does not hold for Alcove with varying initial attention strengths: approximately the same values of λ_w were found for both strategies for the high performers, and the value of λ_w was even somewhat smaller for Situated in the case of the low performers. A possible explanation may be that the variation in initial attention strengths makes some exemplars (houses) more difficult to learn than others, while equal initial attention strengths makes all houses equally difficult. Unequal difficulties are needed for the Situated strategy to have an effect, and in the case of equal difficulties the Random Recycling strategy should be preferred. This result of obtaining approximately equal values of λ_w for both strategies indicates that it may be preferable to use Alcove with initial attention strengths reflecting the initial salience of the dimensions, and that the possibility to do this makes Alcove preferable to Configural Cue.

Nevertheless, the parameter values found with that variant of Alcove still differ over the strategies. Further analyses revealed that in this variant of Alcove the value of λ_a had a very great effect on the learning curves obtained: with the same parameter settings as found for Situated for the high performers a much higher learning curve was found when using Random Recycling as the strategy. So, in this particular case a higher learning rate of the attention strengths reduces learning, as it produces lower learning curves.

5.6 Use of the Situated strategy: Another view on items

Up to now, we have regarded every house as an item. So, the Situated strategy has been working on the set of houses, presenting houses that had been categorized incorrectly more frequently than houses that had been categorized correctly on the last presentation. However, this may not be a good approach for concept learning. In the case of concept learning, the student has to learn something more general than the classification of individual houses, namely concepts. Therefore, learning to classify an individual house may depend on learning to classify other houses. For instance, it is probably easier to remember that a sparrow is a bird and not a mammal when being confronted repeatedly with other animals similar to sparrows that are also birds.

Each disjunction of a classification rule can be associated with a subset of the houses, namely with those houses that meet the disjunction. Given this hierarchy of a set of houses consisting of a number of subsets that consist in their turn of individual houses, the selection of a house for presentation can be done in two phases. First, a subset is selected. Next, from that subset a house is selected for presentation. The advantage of this division into two phases is that the selection process can take account of the internal structure of the set of houses.

For each of these two phases a sequencing strategy can be used. So, there are four possible combinations of using Random Recycling and Situated in such a setting. We decided to focus first on the effect of using the Situated strategy as a subset selection strategy. Therefore, only two combinations were considered, namely those with Random Recycling as a strategy of selecting a house given a subset. It remains interesting, however, to investigate the other two possibilities as well.

With this choice of combinations, we are actually not changing the Situated strategy, but merely changing the view on items: in this case, subsets are viewed as the items on which the Situated strategy works. That this is a reasonable choice can be seen in the following example. Consider, for instance, a case in which clock times have to be learned. Once a student shows to know "two o'clock", it seems reasonable to present full hours less often, because there is a high likelihood that the subject will also know "three o'clock" etc. Hence, in this example, the Situated strategy should work with "full hours", "half hours", etc., as items rather than with "two o'clock", "three o'clock", etc., as items. It could be given as a guideline that the Situated strategy should work on items which are learned independently of each other.

5.7 The quest continued

In order to be able to compare both item sequencing strategies, we still need gradually increasing learning curves. The approach used above did lead to increasing learning curves, but the increase was too steep and, besides, too many subjects still performed at a relatively low level during most of the experimental session, which was very demotivating for them. Though the use of a well-chosen and fixed initial sequence clearly reduced the variance between subjects and influenced the moment of the rise in performance, something else was needed to obtain a *gradual* rise.

A gradual rise is obtained when disjunctions of the classification rules are learned independently of each other. In the classification rules as used above (A = 000xxx + 111xxx, and B = 001xxx + 110xxx) this is not the case. For instance, once the subjects discover that the white houses belong to A when they have two windows and to B when they have one window, it is very straightforward to discover that it is precisely the reverse in the case of the coloured houses. So, this category structure was not very well suited for gradual learning.

Another argument against that category structure is that the classification rules only contain two disjunctions. So, the learning can only contain one intermediate stage: a stage in which one disjunction is mastered and the other one is still unmastered. Though we could evaluate sequencing strategies with such a category structure, it seems preferable to widen the scope on which the strategies may work, hence to increase the number of disjunctions from two to three again.

Given that the classification rules must consist of three disjunctions and that the learning of each disjunction must be independent of the learning of the other disjunctions, this raises the question of how to make sure that the learning curves increase. Three disjunctions are even harder to learn than two disjunctions, and the independence of the learning of the disjunctions will also increase the difficulty.

A possible solution is to ensure that at least one of the disjunctions is relatively easy to learn by using very salient dimensions as a classification criterion. Therefore, we decided to use the colour (of the roof and facade) and shape of the roof as relevant dimensions, in accordance with their salience as discovered in the exploratory studies reported in Section 5.3.

The colour (of the roof and facade) is used to distinguish between the disjunctions: colour is constant within each disjunction, and varies between the disjunctions of the same category. As can be seen in Figure 51, houses with the same colour pattern are very readily distinguishable from the other houses. The first disjunctions of the rules have been chosen such that they are easy to learn: all houses with a coloured facade and roof belong to A, and all houses with a white facade and roof belong to B. The houses of both other colour patterns are divided equally over the categories.

The shape of the roof is used to classify houses with a white facade and coloured roof: houses with a flat roof belong to A, and those with a pyramidal roof belong to B. The number of windows is used to classify houses with a coloured facade and white roof: houses with two windows belong to A, and those with one window belong to B. This leads to the classification rules
A = 11xxxx + 10xxx1 + 010xxx, B = 00xxxx + 10xxx0 + 011xxx.

As the shape of the roof is a very salient dimension, it is expected that the disjunctions which use it as a criterion for classification will be easier to learn than those that use the number of windows. However, this is not very important: it suffices that subjects who attend to the shape of the roof will have more difficulty with the disjunctions using the number of windows and vice versa. So, the learning of the disjunctions is independent and gradual learning is possible.

In order to enlarge the natural salience of the colour even further and to give the subjects a hint as to the kind of classification rules to look for, the following sentence was added to the instructions: "The colour of the houses (facade and roof) determine which other dimensions you should pay attention to.". This addition may also lead the subjects to use an analytic learning strategy, but this is not considered a problem as the previous experiment already indicated that subjects tend to use an analytic strategy anyway.

5.7.1 Pilot study 4: Effect of salience (three disjunctions)

Method

Design and Procedure. The same design and procedure were used as in Experiment 1, except that in this case the order in the first test phase was random again, and the order in the first practice phase was determined by the strategy.

Subjects. Six subjects with university or higher vocational training participated voluntarily in the pilot study. Subjects were randomly assigned to one of the two experimental conditions.

Materials. The same kind of houses were used as in Experiment 1. The following classification rules were used:
A = 11xxxx + 10xxx1 + 010xxx, B = 00xxxx + 10xxx0 + 011xxx.
Not all 64 houses that meet these rules were used, but a selection was made such that, with the exception of the presence of a chimney, all irrelevant dimensions were balanced within each disjunction. For instance, for the first disjunction of A (which contains houses with a grey facade and roof), houses

were selected such that an equal number of houses had one window and two windows, a dark door and a white door, and a flat roof and a pyramidal roof. The selected houses are shown in Figure 51.

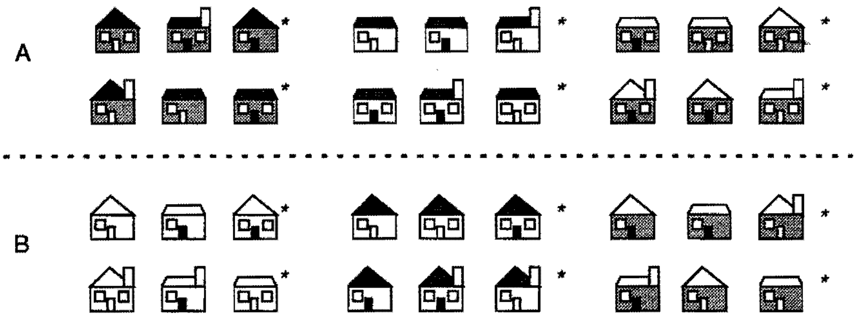


Figure 51: Stimuli used in Pilot study 4 and 5. Transfer stimuli are marked with an asterisk.

Results and Discussion

The results of the pilot study are shown in Figure 52. The learning curves look very promising in the sense that they gradually increase for all subjects.

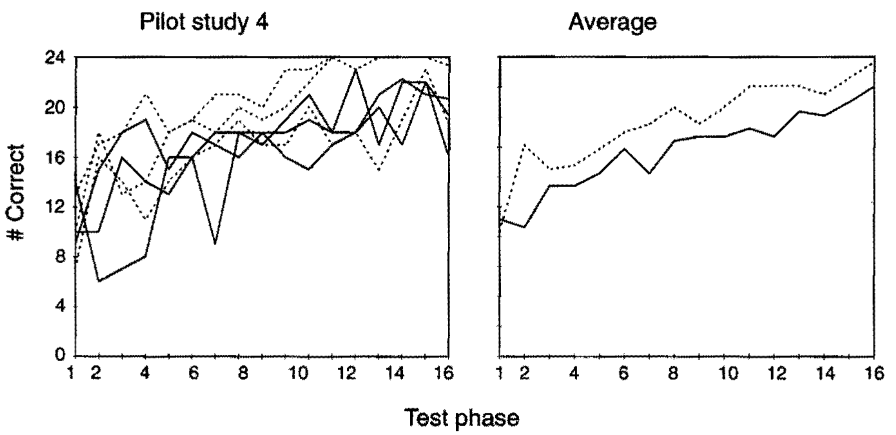


Figure 52: Results of Pilot study 4. Each line in the left-hand graph represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line) and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.

However, the effect of strategy is not what was expected: the subjects with Random Recycling clearly outperform those with Situated. The large difference in the second test phase is striking. A possible explanation for this difference is the order in which stimuli were presented during the first practice phase. In the case of Random Recycling, all houses are presented in such a way that houses from the various disjunctions are distributed homogeneously over the sequence. On the other hand, in the case of Situated, it may well be that houses of one disjunction are presented much more frequently than houses of the corresponding disjunction of the other category. The guessing probability of a correct classification of 50%, in particular, makes it very likely that this will occur. This may lead to confusion of the subjects, who are biased to expect as much houses in both categories. Subjects may start to pay attention to irrelevant dimensions as the relevant dimension does not vary in most of the houses (Elio & Anderson, 1984).

This is confirmed by the loggings of the subjects in the Situated condition. Two out of three subjects had a rather unequal distribution of the presentation of houses from the different disjunctions: on average, four houses were presented more as belonging to a disjunction of one category than to the corresponding disjunction of the other category. For instance, six houses were presented with a white facade and coloured roof that belong to A, and only two that belong to B. These two subjects performed very badly on the second test phase. The third subject only had such an unequal distribution for one of the disjunctions. This subject performed almost as well in the second test phase as the subjects in the Random Recycling condition.

We decided to use a fixed initial sequence during the first test and practice phase, as we had done before (cf., discussion Pilot study 2). However, in this case the initial sequence was chosen such that it corresponds to the kind of initial sequences produced by Random Recycling: a house of a certain disjunction was only presented when a house of all other disjunctions had been presented. In correspondence with the initial sequence as used in Pilot study 3 and Experiment 1, the variance within the first part of the sequence was reduced by presenting all houses with a chimney in the second part. The initial sequence chosen is shown in Figure 53.

5.7.2 Pilot study 5: Effect of a fixed initial sequence on Situated

Method

Design and Procedure. The same design and procedure were used as in Pilot study 4, except that in this case the order in which stimuli were presented during the first test and practice phase was determined beforehand and was the same for all subjects. This order is shown in Figure 53.

Subjects. Three subjects with higher education participated voluntarily in the pilot study. All subjects were assigned to the Situated condition.

Materials. The same stimuli were used as in Pilot study 4 (see Figure 51), with classification rules: A = 11xxxx + 10xxx1 + 010xxx, B = 00xxxx + 10xxx0 + 011xxx.

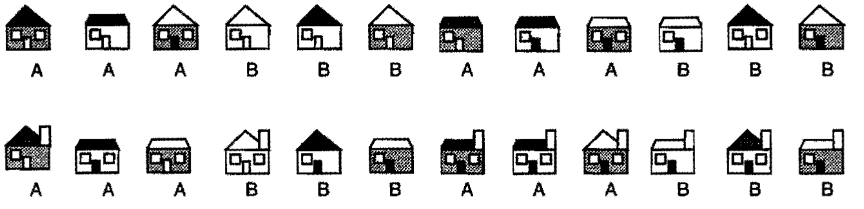


***Figure 53:** The order in which the stimuli were presented during the first test and practice phase of Pilot study 5 (from left to right, from top to bottom). The letters below the houses indicate the category to which they belong.*

Results and Discussion

The results of the pilot study are shown in Figure 54. There is no clear advantage of using this fixed initial sequence: though one subject performs very well, the other two remain at a relatively low level of performance, lower even than the subjects in the Situated condition of the previous pilot study. Nevertheless, the performance of these subjects on the second test phase is somewhat better than the performance of the subjects in Pilot study 4.

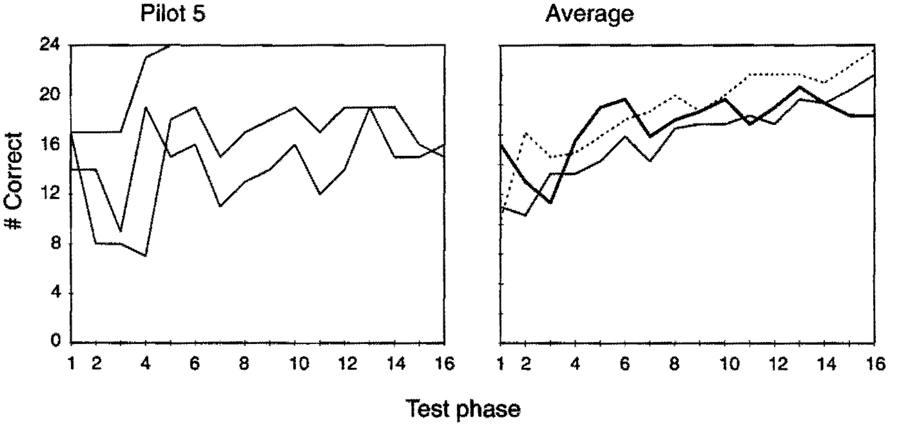


***Figure 54:** Results of Pilot study 5. Each line in the left-hand graph represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition of Pilot study 4 (dashed line), the Situated condition of Pilot study 4 (solid line), and the Situated condition of Pilot study 5 (bold solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.*

A possible explanation of the decrease in performance lies in the order of the initial sequence. Though this order implicitly indicates the irrelevance of the chimney, it also has the property that the houses from each disjunction of both rules are spread homogeneously over the sequence as a whole. One consequence of this may be that for the first disjunction of both rules (each of which differs from all other disjunctions in respect of very salient dimensions) it is less obvious that similar houses are classified in the same way than they would have been if the houses were presented closer to each other.

This kind of initial order (though not exactly the same) was also a natural consequence of the Random Recycling strategy in Pilot study 4. So, there must be some reason other than the lack of a good initial sequence for the Situated strategy performing worse than Random Recycling in Pilot study 4, and even worse in Pilot study 5 under a condition which was even closer to that of Random Recycling.

We conclude that the problem is internal to the Situated strategy and that some changes in the use of the strategy may be needed.

5.7.3 Changes in the Situated strategy

Value of the strategy's parameter

A first adjustment which can be made to the Situated strategy is the value of its parameter. As discussed in Chapter 3, this value was fixed at 10, based on an assumption that we would like the probability of an item from the bad set being presented to be approximately 50% when the number of items in the bad set is only 5% of the total number of items. This assumption was meant to guarantee that learning still takes place for the last 5% of the items, but also that forgetting is prevented by presenting members of the good set with an equal probability in the end of learning.

Though this may be true and, hence, 10 may be a reasonable value for larger sets of items, problems may arise when the set of items is fairly small. In the case of the above classification rules, the number of items on which the strategy works equals six. This implies that 5% of the items amounts to zero items when rounded to a whole number. So, in the end, when the last item has to be learned, the probability of presenting an item of the good set is still only 33%. When learning starts and only the first item has been learned, the likelihood of this item being presented again is only 2%. Hence, it may be a very long time before an item of the good set is presented again, and forgetting or an unjustified addition to the good set (likely with a high guessing probability) may affect performance very badly.

We decided to change the value of the parameter to 5, in order to obtain a probability of approximately 50% in the final phase of learning.

Criterion for transfers to the good set

In contrast to the tasks in the case of paired associates learning, there is a very high likelihood of transferring an item to the good set even though the subject does not know it yet, as the likelihood of classifying an item correctly by chance is 50%. In order to cope with this high guessing probability, we decided to change the criterion for a transfer to the good set. A subset is no longer transferred to the good set when a house of that subset has been classified correctly, but only when the classification was correct the last three times (a house of) that subset was presented.

5.7.4 Changes to reduce variance

In order to reduce the variance between subjects as far as possible, two additional changes were made in the conditions under which the experiment will be performed.

In the first place, it was decided to use only one shade of gray in the stimuli. In the pilot studies some subjects indicated that they believed that some dimensions could take more than two values, as various shades of gray were present in the stimuli. Therefore, all shades of gray were replaced by the same colour. The implication for the stimuli can be seen in Figure 55. A possible side effect of this change may be that the various parts of the house (facade, roof, and door) are less distinguishable from each other when they happen to have the same colour. However, we do not expect this effect to be harmful to the learning.

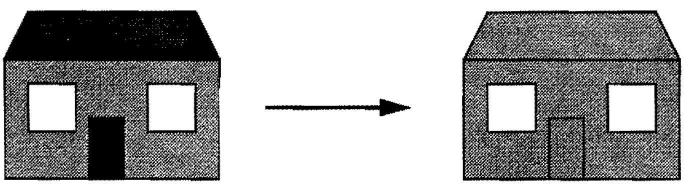


Figure 55: *Change of gray shades in the stimuli. Left a house as in Pilot study 4 and 5, to the right the same house after the transformation of color.*

In the second place, it was decided to restrict the use of houses with a chimney to the transfer test and to use only houses without a chimney in the practice and test phases. The presence of a chimney is irrelevant for the classification, but a subject still occasionally forms hypotheses on the basis of the presence of a chimney, sometimes even complicated hypotheses in which the number of windows is added to the number of chimneys.

5.8 Experiment 2

5.8.1 Method

Design and Procedure

The same design and procedure were used as in Pilot study 5. Again, the order in which stimuli were presented during the first test and practice phase was determined beforehand and was the same for all subjects. This order is shown in Figure 56.

Subjects

Eighteen subjects with university or higher vocational training participated voluntarily in the experiment. None of them had any prior experience with the stimuli and this kind of task. Subjects were randomly assigned to one of the two experimental conditions.

Equipment

The experiment was run on PC's (386 & 486, 14 inch screen) under Windows in a network configuration. The subjects used an ordinary keyboard to respond; use of the mouse was not needed. No audio was used. All instructions and feedback were given in the form of text on the screen.

Materials

The same category structure was used as in Pilot study 4 and 5. So,
A = 11xxxx + 10xxx1 + 010xxx, B = 00xxxx + 10xxx0 + 011xxx.

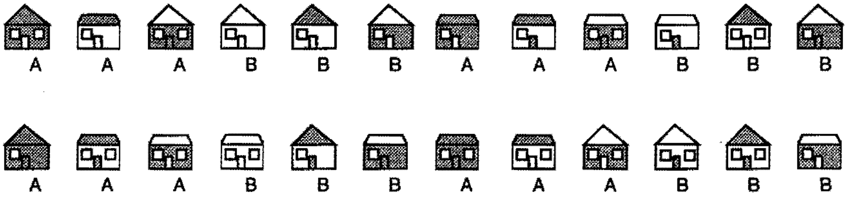


Figure 56: *The order in which the stimuli were presented during the first test and practice phase of Experiment 2 (from left to right, from top to bottom). The letters below the houses indicate the category to which they belong.*

5.8.2 Results

The results of the experiment are shown in Figure 57. A MANOVA was performed on logit-transformed proportions of correct responses, with the test phase as a within-subjects repeated measures factor and the experimental condition (strategy used in the practice phases) as a between-subjects factor. When a subject was presented with the transfer test due to a score of 100% on two sequential test phases, a score of 100% correct was assigned to the skipped test phases. The results of the analysis are summarized in Table 26.

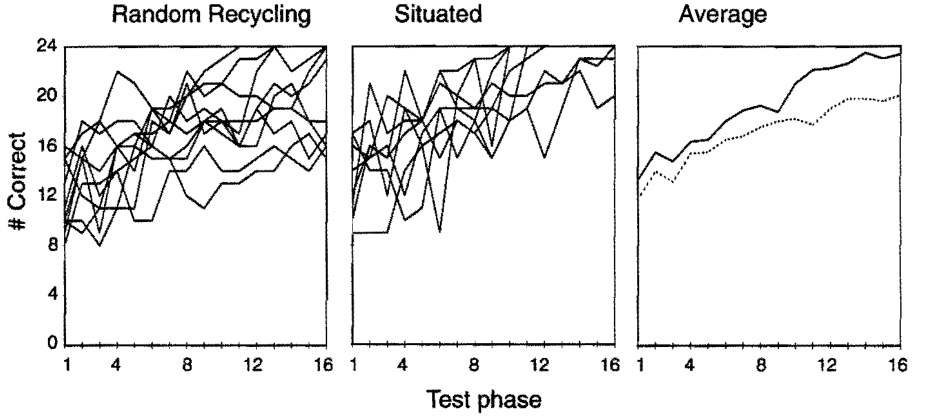


Figure 57: Results of Experiment 2. Each line in the two left-hand graphs represents a subject, and each line in the right-hand graph represents the average over subjects in the Random Recycling condition (dashed line), and the Situated condition (solid line), respectively. The test phases are given on the x-axis and the number of correct responses on the y-axis.

There is a significant main effect of strategy [$F(1,1)=6.44$, $p < .05$]. There is no main effect of test, and the interaction between test phase and strategy is not significant either.

Table 26: Results of the MANOVA on the data of Experiment 2.

Source	Num DF	Den DF	F	
		Between subjects		
Strategy	1	1	6.44	*
		Within subjects		
Test	15	2	2.68	
Test × Strategy	15	2	1.11	

$^*p<.05$, $^{**}p<.01$, $^{***}p<.001$

A MANOVA was also performed on the last eight test phases. The results of this analysis are summarized in Table 27. There are significant main effects of strategy [$F(1,1)=6.12$, $p < .05$] and test phase [$F(7,10)=9.45$, $p < .01$]. A significant interaction was found between test phase and strategy [$F(7,10)=3.24$, $p < .05$].

Table 27: *Results of the MANOVA on the data of Experiment 2 over the last 9 test phases.*

Source	Num DF	Den DF	F	
		Between subjects		
Strategy	1	1	6.12	*
		Within subjects		
Test	7	10	9.45	**
Test × Strategy	7	10	3.24	*

***p* <.05, ***p* <.01, ****p* <.001**

5.8.3 Discussion

The lack of a significant effect of test phase for the analysis over the total range of test phases may be due to the irregularity of the learning curves, especially in the beginning, which introduces a lot of variance. This irregularity can be explained by the high probability of classifying a house correctly by chance. The main effect of test over the last eight test phases indicates that subjects did indeed learn.

The main effect of strategy, both over the total range and over the last eight test phases, indicates that the use of the Situated strategy increased the subjects' performance. However, the difference in the first eight test phases is rather minimal, so the effect of the Situated strategy only becomes prominent in the last eight test phases. This is reflected by the fact that the interaction between strategy and test phase is significant over the last eight test phases, while it is not significant over the total range. So, the increase in performance due to the Situated strategy produces steeper learning curves in the second half of the range. This can be explained by the fact that the Situated strategy needs time to adapt to the subjects. The Situated strategy can only produce an advantage once the subjects have learned something.

The group subjects in the Situated condition perform more homogeneously than the group subjects in the Random Recycling condition. The average standard deviation over the last eight test phases is 14.2% for the Random Recycling condition and 9% for the Situated condition. The subjects in the Random Recycling condition can be divided into two groups: one group with a score of between 65% and 70% correct on the last test phase and a second group with a score between 96% and 100% correct. The subjects in the Situated condition all have scores which are between 96% and 100% correct, with the exception of one subject with a score of 83% correct. This indicates that the advantage of using the Situated strategy is mostly due to an increase in the performance of the low performing subjects, reducing the difference between high and low performers.

5.9 Conclusions

As far as the effectiveness of the Situated item sequencing strategy is concerned, the results indicate an advantage of using the Situated strategy compared to Random Recycling in a concept learning task. This advantage was due to an increase in the performance of the low performing subjects. These results are similar to those found for paired associates learning. A different view of items was needed to obtain these results: not every exemplar (house) was regarded as an item, but the item sequencing took place on subsets. Future research should investigate the effect of using the Situated strategy also within the subsets.

The applicability of the above to interactive instruction may be challenged in several ways. In the first place, one might argue that concept learning mainly occurs in daily life, in which an item sequencing strategy cannot be used. However, concept learning plays an important role in various instruction situations, including the classroom. Examples include biology (e.g., mammals, birds, insects), chemistry (e.g., metals, acids, salts), and art history (e.g., Impressionism, Baroque, Jugendstil). In all these cases, interactive instruction can be used and, hence, an item sequencing strategy.

In the second place, one might argue that classification rules were used in the experiments that were explicitly constructed as disjunctions of subsets, and that it may be more difficult to identify what to use as items in real life. However, the subsets used above reflected similarity rather than subrules, and it seems undeniable that within concepts in real life it is often possible to identify subgroups of exemplars with salient features that are very similar to each other. These subgroups can then be used as items. For instance, when designing a lesson on the concept of mammals, subgroups like whales, felines, and rodents could be distinguished.

The application of the results to interactive instruction is not restricted to concept learning. One outcome is that in order for the Situated strategy to be effective, a different view of items is sometimes needed. This can be applied to other kinds of learning as well. For instance, it has been used in the Appeal prototype (Dutch for English speaking persons) for learning clock times. Instead of using items for “5 o’clock”, “6 o’clock”, “5 minutes past 11”, etc., items have been used for "full hours", "half hours", "minutes between the full hour and a quarter past the full hour", etc.

As far as the validity of concept learning models is concerned, a comparison of the fits of ALCOVE and Configural Cue to the data of the first experiment showed two major shortcomings of the models. In the first place, the learning curves of the models rose very early in the learning process, but with a slope that decreased quite fast. The curves of the high performance subjects rose some test phases later, but with a steeper and more constant slope. In the second place, the parameter values found for both strategies differed: much higher parameter values were found for the Situated strategy than for Random Recycling. The use of varying initial attention strengths in Alcove reduced this effect considerably. This suggests that it may be preferable to use Alcove with initial attention strengths that reflect the initial salience of the dimensions, and that the possibility of doing this makes Alcove preferable to Configural Cue.

References

Ashby, F.G., & Gott, R.E. (1988). Decision rules in the perception and categorization of multidimensional stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*, 33-53.

Elio, R., & Anderson, J.R. (1984). The effects of information order and learning mode on schema abstraction. *Memory and Cognition, 12*, 20-30.

Friedman, D., Massaro, D.W., Kitzis, S.N., & Cohen, M.M. (1995). A comparison of learning models. *Journal of Mathematical Psychology, 39*, 164-178.

Gelman, S.A., & Markman, E. (1986). Categories and induction in young children. *Cognition, 23*, 183-209.

Gluck, M., & Bower, G. (1988a). Evaluating an adaptive network model of human learning. *Journal of Memory and Language, 27*, 166-195.

Gluck, M., & Bower, G. (1988b). From conditioning to category learning: an adaptive network model. *Journal of Experimental Psychology: General, 117*, 227-247.

Goldstone, R.L. (1994). The role of similarity in categorization: providing a groundwork. *Cognition, 52*, 125-157.

Holland, J.H., Holyoak, K.J., Nisbett, R.E., & Thagard, P.R. (1987). *Induction: Processes of inference, learning, and discovery.* London: The MIT Press.

Katz, J.J., & Postal, P.M. (1964). *An integrated theory of linguistic descriptions.* Cambridge, MA: The MIT Press.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12*, 1-48.

Kruschke, J.K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review, 99*, 22-44.

Kruschke, J.K. (1993). Three principles for models of category learning. In G.V. Nakamura, D.L. Medin, and R. Taraban (Eds.), *The psychology of learning and motivation: Categorization by humans and machines* (Volume 29). New York: Academic press.

Livingston, K.R., & Andrews, J.K. (1995). On the interaction of prior knowledge and stimulus structure in category learning. *The Quarterly Journal of Experimental Psychology, 48A*, 208-236.

Luce, R.D. (1963). Detection and recognition. In R.D. Luce, R.R. Bush, and E. Galanter (Eds.), *Handbook of mathematical psychology.* New York: Wiley.

Medin, D.L., & Schaffer, M.M. (1978). Context theory of classification learning. *Psychological Review, 85*, 207-238.

Medin, D.L., & Smith, E.E. (1981). Strategies and classification learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 7*, 241-253.

Medin, D.L., Wattenmaker, W.D., & Hampson, S.E. (1987). Family resemblance, conceptual cohesiveness, and category construction. *Cognitive Psychology, 19*, 242-279.

Nosofsky, R.M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 115*, 39-57.

Nosofsky, R.M. (1988). Similarity, frequency, and category representations. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*, 54-65.

Nosofsky, R.M. (1992). Similarity scaling and cognitive process models. *Annual Review of Psychology, 43*, 25-53.

Nosofsky, R.M., & Kruschke, J.K. (1992). Investigations of an exemplar-based connectionist model of category learning. In D.L. Medin (Ed.), *The psychology of learning and motivation: Categorization by humans and machines* (Volume 28). New York: Academic press.

Nosofsky, R.M., Kruschke, J.K., & McKinley, S.C. (1992). Combining exemplar-based category representations and connectionist learning rules. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 18*, 211-233.

Pearce, J.M. (1987). A model of stimulus generalization for Pavlovian conditioning. *Psychological Review, 94*, 61-73.

Pearce, J.M. (1994). Similarity and discrimination: A selective review and a connectionist model. *Psychological Review, 101*, 587-607.

Posner, M.I., & Keele, S.W. (1968). On the genesis of abstract ideas. *Journal of Experimental Psychology, 77*, 353-363.

Regehr, G. & Brooks, L.R. (1993). Perceptual manifestations of an analytic structure: The priority of holistic individuation. *Journal of Experimental Psychology: General, 122*, 92-114.

Regehr, G. & Brooks, L.R. (1995). Category organization in free classification: The organizing effect of an array of stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 21*, 347-363.

Roberts, P.L.& MacLeod, C. (1995). Representational consequences of two modes of learning. *The Quarterly Journal of Experimental Psychology, 48A*, 296-319.

Rosch, E., & Mervis, C.B. (1975). Family resemblances: Studies in the internal structure of categories. *Cognitive Psychology, 7*, 573-605.

Shepard, R.N. (1987). Toward a universal law of generalization for psychological science. *Science, 237*, 1317-1323.

Tversky, A. (1977). Features of similarity. *Psychological Review, 84*, 327-352.

Vandierendonck, A. (1993). *Knowledge and category learning*. Paper presented at the "Sixth Conference of the European Society for Cognitive Psychology", Elsinore, September 11-15, 1993.

Vandierendonck, A. (1995). A parallel rule activation and rule synthesis model for generalization in category learning. *Psychonomic Bulletin & Review, 2*, 442-459.

Chapter 6

The Navigation Agent: implementing a mixed locus of control

Abstract

The function of the Navigation Agent is to determine a path through the course material adapted to the interests, foreknowledge, and capabilities of the individual student. On the other hand, it should remain possible for the student to select topics himself. The moment at which a topic changes may also be determined both by the student and the agent. The agent should base this moment on the student's performance. A method for topic selection in a "mixed locus of control" setting is discussed. In two experiments, the effectivity of topic selection and initiative by the agent was explored. Topic selection by the agent showed an advantage for students who were unable to monitor their own learning process. Initiative by the agent reduced the time needed to finish the task by stimulating the subjects to study the lessons in a balanced way. The experiments provide evidence that a mixed locus of control is preferable to either the student or the agent taking the initiative and selecting other lessons. This implies that both the student and the agent can take the initiative and can select other lessons.

6.1 The task of the Navigation Agent: topic selection

In its most basic form, a course can be regarded as a set of items the student has to study. An interactive instruction system can help the student by determining the order in which these items are presented. This is what we have called item sequencing above, and have discussed in the previous three chapters as a task of the Practice Agent. No prior knowledge about the items and their relations was used in the item-sequencing algorithm discussed above .

For several reasons, this needs not, however, be sufficient for optimal learning. In the first place, the order in which items should be presented to the student may depend on content-specific information about the items. For instance, it seems preferable to present clock times when the student has mastered numbers, and not the other way around. In the second place, the set of items may be very large, imposing a high cognitive load on the student. It seems preferable to restrict the set of items the student is learning concurrently. Lastly, it seems preferable to have some semantic coherence in the sequence. For instance, when studying clock times it is more natural to do all the clock times and not to interrupt this process by, for instance, introducing the study of the past tense.

So, in order to keep item sequencing as simple as possible (so that we can use the algorithm described above), it has to take place on a set of items which are semantically close and independent of each other (in the sense that no item is prerequisite for any other item in the set). Therefore, we divide the set of items into subsets which meet these requirements. These subsets do not have to be disjoint. Given a particular subset, we use item sequencing to determine at any moment which item of that subset should be presented. This remains the task of the Practice Agent. We will use the term topic selection to indicate the process of determining the subset of items for the Practice Agent to employ. This will be the task of the Navigation Agent, together with determining when topic selection should take place. In this chapter, the task of this agent will be analysed in more depth, and an algorithm will be described, as well as its evaluation.

6.2 Two views of navigation: user navigation and guidance

Many applications enable the user to access a large database of information in all kinds of formats (like text, video, sound tracks), say presentations. Examples of such applications are CD-I's such as the Smithonian CD-I and CD-ROMs such as Encarta. An interactive instruction system can also be viewed as such an application. In this case, the presentations are often interactive and are usually called exercises. In the terms we have used above, presentations constitute the units on which the Navigation Agent operates, and a presentation may contain several items on which the Practice Agent operates.

In existing applications, there are two general approaches. In the first approach, the users navigate through the database themselves, using, for instance, menu structures. They can be aided in their task by various ways of visualizing the database, for instance with cone trees visualizing hierarchical information (Robertson, Mackinlay & Card, 1991), or with "subway lines" visualizing cohesion (Espinosa & Baggett, 1993). Nevertheless, navigation often remains a difficult task, particularly in the case of a large database.

In the second approach, 'guidance' is used: part of the application navigates for the user in the database. For instance, in the Smithonian CD-I, the user can get a kind of guided tour through the museum. In Laurel, Oren, and Don (1990) a system is described in which anthropomorphic agents guide the user through a database using a narration metaphor. In an interactive instruction system, guidance means that the successive lessons and exercises are selected by the system. As described above, this is a task of the Navigation Agent.

A possible danger of pure guidance may be that users feel a lack of control (Norcio & Stanley, 1988). As argued in the first chapter, we advocate a mixed locus of control (Gentner, 1992), in which both the user and the system (i.e., Navigation Agent) can navigate and the decision about who navigates depends on the interaction. In the case of an interactive instruction system, this decision may depend on the student's performance. Such a mixed locus of control in the area of navigation is also described by Takeuchi and Otsuki (1990). As argued in the first chapter, the system should take over control especially in case of low performance.

6.2.1 Problems with current approaches to guidance in IIS's

There are several problems associated with the current approaches to guidance in interactive instruction systems.

1. *Flexibility concerning the foreknowledge of the student is limited.* It is generally assumed that the foreknowledge of all students is equal, and that all students have exactly the foreknowledge needed for the course and nothing more. This is not necessarily the case, however.

Sometimes, user profiles or stereotypes are used, and the user is initially assigned to one of the disjoint classes (Rich, 1989). A classification often mentioned is that of novices and experts. Hensgens, Rosmalen, and Baaren (1995) describe a system in which students gain access to a subset of the course material with a pre-set path, depending on their profiles. Sometimes separate applications are developed for separate market segments. For instance, there may be a course in "Spanish for beginners", and a separate course in "Spanish for advanced learners". However, the use of a restricted number of discrete classes cannot really capture the continuum of the difference in foreknowledge.

2. *Flexibility concerning the learning objective is limited.* It is generally assumed that the learning objective of all students is equal, and that all students have to learn everything the course has to offer. This is not necessarily the case, however.

Sometimes, the student profiles mentioned before are used, which in this case should capture what the student should learn or wants to learn (Hensgens, Rosmalen, & Baaren, 1995). Again, separate applications are sometimes developed for separate market segments. For instance, there may be a course in "Spanish for business men", and a separate course in "Spanish for holiday-makers". Of course, the discrete classes form only an approximation of the continuum of the difference in learning objectives.

3. *Flexibility concerning the capabilities of the student is limited.* Even when the initial level and the learning objectives of two students are the same, it may be that the path towards the learning objective should be different for the two. This may be because one student has a quicker understanding than the other one: a somewhat slower student may reach the goal in smaller steps than a fast student. But it may also be because one student prefers videos while another prefers text, or because one student prefers formulas and another examples. So, different paths through the course material may be needed even when the same foreknowledge and learning objectives are assumed for all students.

4. *The character of guidance tends to be static and often also linear.* Even assuming a flexible use of foreknowledge, learning objectives, and capabilities, the problem remains that these are taken to be established facts that are initially true and do not change during the interaction with the application. Foreknowledge is, however, only an estimation of what a student knows, just like capabilities. A student may forget, which may lead to an apparent change of foreknowledge during the interaction. A student who is initially judged as slow may turn out to be faster than expected during the interaction with the application, and a student who was judged as fast may turn out to have more problems than expected. Also, the learning objectives may change during the interaction. During the interaction, a student may indicate an interest in achieving another learning objective in addition to (or perhaps instead of) the one chosen initially.

6.2.2 A combination of user navigation and adaptive guidance

In order to avoid the problems mentioned above, the Navigation Agent should determine a path through the course material at run time, taking the current estimation of the foreknowledge, goals, and capabilities of the student into account at any moment. Therefore, the course material should consist of separate modules which are described by attributes such as the foreknowledge needed to study the module, the goals accomplished by studying the module, and the difficulty of the module.

The idea to analyse course material in order to make the relations between concepts in the material explicit has been argued before in a different setting (see Pask, Kallikourdis & Scott 1975; Jochems, 1980). The idea of modularity is also advocated by Koehorst, Baaren, and Rosmalen (1991), but they stress the importance of modularity for the course design process. They mention casually that modularity can also be used to adapt a course to a group of students by a non-programmer.

The global structure of topic selection in the case of a mixed locus of control is shown in Figure 58. Both the agent and the student can select a topic. The selection by the agent is influenced by the estimated knowledge, goals and capabilities of the student, and the description of the course material. The estimations are changed on the basis of the interaction with the student, the topic selection by the student, or explicitly by the student.

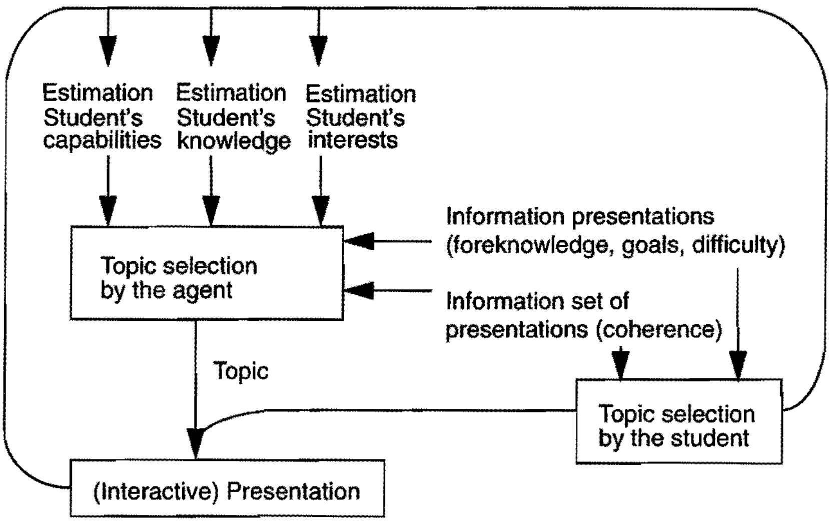


Figure 58: *Global structure of topic selection in the case of a mixed locus of control.*

6.3 A topic selection algorithm

A topic should be chosen that is sufficiently easy, contributes to reaching the student's goals (eventually), and allows the student to reach his goals in the most effective way. The foreknowledge, goals, and capabilities of the student as estimated by the agent may be adjusted during run time, either explicitly by the student or teacher, or implicitly because assumptions turn out to be incorrect. The global structure of the topic selection algorithm used by the agent is shown in Figure 59.

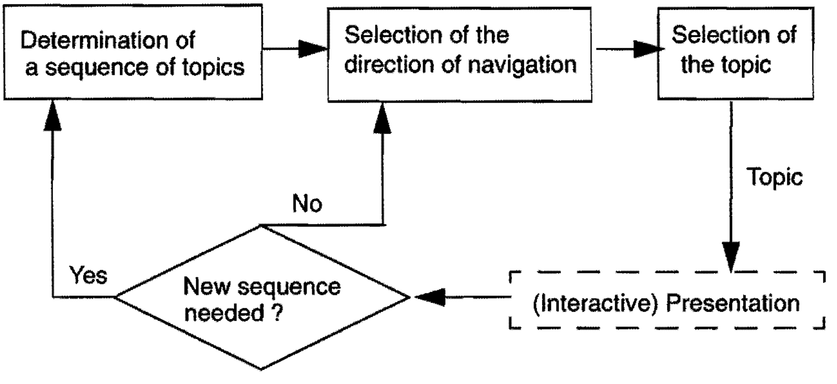


***Figure 59:** Global structure of the topic selection algorithm used by the agent.*

Determination of a sequence of topics
A sequence of topics is determined that meets the following criteria:

1. If the sequence is studied from beginning to end, the student has *sufficient foreknowledge* to study each subsequent topic, and eventually all the student's *goals are reached.*

2. The sequence *meets a certain criterion,* which may be related to the effort it takes the student to study a topic of the sequence, or, in other words, the inherent difficulty of the topics. For instance, this effort or difficulty should be smaller than a certain amount for every topic of the sequence.

The level of difficulty can be implemented in several ways. In the first place, the course designer can associate numbers with the topics or exercises, reflecting their relative difficulty. This may, however, be rather difficult and time-consuming. In the second place, the relative levels of difficulty can be determined experimentally. During prolonged use of the system, the levels of difficulty will increasingly reflect the relative difficulty as experienced by the average student. In the third place, a limited number of categories of difficulty can be used, indicated, for example, by a 3 for items suitable for very bright learners only, a 2 for items suitable for average learners, and a 1 for items suitable for even poor learners.

The criterion may also impose requirements on the coherence of the sequence: the semantic relation between successive topics. Coherence is partly a natural consequence of the sequence determination (see below): topics needed for the same goal are likely to be close together in the sequence. Furthermore, the semantic structure can be used which is mostly imposed on the topics by the course designer, like a division into themes, chapters, and paragraphs (possibly during post-processing of the sequence).

3. The sequence has approximately the *minimal costs* of the sequences that meet the criterion. We have used the word "approximately", because for efficiency reasons heuristics will be used instead of a complete search (which would lead to an exponential growth in calculation time). The costs are a measure of the effort needed for the individual student to study the complete sequence. A possible implementation is to sum up the difficulty of the topics of the sequence.

The implementation of a sequence that meets these criteria takes place as follows. A simple form of recursion is used. Departing from the goal of a particular student, topics are searched with which that goal can be reached. Only topics that are not too difficult for that student are considered. The foreknowledge associated with these topics should be achieved before they are studied, and the same procedure can be used to obtain a sequence resulting in that foreknowledge. During the recursion the costs of every sequence are calculated, and the cheapest sequence is chosen. When no sequence meets the criterion, the criterion may be changed.

Selection of the direction of navigation

Given this sequence, and a position in the sequence, navigation by the agent can be in two directions: either towards the end of the sequence, i.e., a more difficult lesson, or towards the beginning of the sequence, i.e., an easier lesson. An easier lesson should be selected if the student is performing poorly in the current lesson and a more difficult lesson should be selected if the student performs well. When an easier or more difficult lesson is not available, the current lesson should be selected.

There are several ways in which "poorly" and "well" can be defined, for instance by using the number of successive correct or incorrect responses. Inspired by the concept of a good set (see Chapter 3 for an explanation of the good set), we have chosen the following definitions.

The student's performance in a particular lesson is considered to be "poor" if the he or she has given an incorrect answer to the item last presented in the exercise of that lesson and less than 50% of the items of that lesson have been mastered, in the sense that they belong to the good set. The student's performance in a particular lesson is considered to be "good" in all other cases.

There are two situations in which the student's performance in the current lesson cannot be used.The first situation occurs when the student was not in any lesson at the moment of navigation, for instance at the beginning of the course or when trying a so-called final test (see below). In that case a more difficult lesson will be selected.

The second situation occurs when the student was reading the instructions of a lesson and had not started practising. There are three possible interpretations of the student choosing another lesson while reading the instructions of a lesson for the first time. Either the lesson seemed uninteresting (not leading towards a certain goal), or it seemed too easy (understandable without practising), or it seemed too difficult. In the first two alternatives, a more difficult lesson is needed. In the last alternative, an easier lesson is needed. It is hard to determine which alternative is true, but we hypothesized that it is more likely that it is one of the first two. So, we decided that a more difficult lesson would be selected always when the student was reading instructions for the first time. The experiment will explore whether this decision was correct.

Selection of the topic

Given the sequence and the direction of navigation, the selection of the topic is easy unless there is a mixed locus of control. When only the agent can select topics, the next topic should be the previous or next topic from the sequence for an easier or more difficult topic, respectively.

However, in the case of a mixed locus of control, the student could also have selected topics and, hence, the current topic may not be part of the sequence, or an easier or more difficult topic may require more than one step in the sequence. For instance, consider a sequence 1-2-3-6-12. When the student has selected 12 and next navigation towards an easier topic is required, the selection of that easier topic should depend on the past performance of the student on topics 1, 2, 3, and 6.

The last topic selected by the agent could be used as the index in the sequence from which the previous or next one is chosen. However, if, for instance, the student had selected topic 3 before and had performed well at it, it seems reasonable to use topic 3 as the index in the sequence rather than, for instance, topic 1. So, we decided to increase the index in the sequence to a topic selected by the student whenever the student performed well on that topic, "well" being defined as above.

When is a new sequence needed?
There are several situations in which a new sequence has to be determined. In the first place, a new sequence has to be determined when the capabilities of the student turn out to differ from the expectation. This may be because the student is performing very well and a faster, though more difficult, path through the course material can be chosen. Or it may be because the student performs very poorly and an easier, though longer, path through the course material is needed. There are several ways to determine when and how the level of difficulty that a student can handle should be changed. One possibility is to use the speed with which a student learns a topic in relation to a reference criterion. A second possibility is to use the interaction history, for instance raising the level of difficulty that the student can handle when the student has learned two topics with that particular level of difficulty without any problems. A third possibility is to let the student or a human teacher change the level himself or herself. A combination of these possibilities can be used also, in which the agent makes small adjustments and the student (or human teacher) can make larger adjustments.

In the second place, a new sequence has to be determined when the student's goals have changed. A change of goals may be announced explicitly to the agent by the student or a human teacher. The agent may also perceive a change in interests implicitly because the student keeps on navigating to other topics.

In the third place, a new sequence has to be determined when the foreknowledge of the student turns out to be different from the expectation. The agent can discover this, e.g., when the student performs very poorly on the first topic of the sequence, or when the student performs very well.

Finally, a new sequence has to be determined when the database with course material has changed.

6.4 Experiment 1: student-initiated topic selection via menu structures, guidance, or a mix of both

In the two experiments reported in this section and the next, the task of the Navigation Agent was relatively easy. The foreknowledge of all subjects with respect to the course material was the same and known beforehand, namely none. The goal of all subjects was the same and determined by the experimental setting. The course material was limited so that no alternative paths with different levels of difficulty could be constructed. In this first experiment, the Navigation Agent had only to decide the next topic; the timing of the navigation was determined by the student.

The purposes of this experiment were as follows. In the first place, the advantage of an efficiently-functioning topic selection mechanism should be determined. A simple, but regularly applied form of user navigation, namely a menu structure, was used as a baseline. The effect of combining user navigation and guidance should also be determined.

In the second place, the acceptance by the students of automatic topic selection should be determined. This acceptance may depend, among other things, on the correctness of the choice of an easier or more difficult lesson.We have adopted the hypothesis that automatic topic selection is more effective and acceptable when combined with the possibility of user navigation.

6.4.1 Method

Design and Procedure

Three conditions were used in a between-subjects design: a "Menu condition", a "Guidance condition", and a "Mixed condition". In all conditions, the subjects were instructed to study "Square Dance" lessons in such a way that they would learn to perform the "Slide through" as quickly as possible. They could try to pass the "Slide through" test as often as and whenever they wanted. The experiment ended when the test was completed correctly. No feedback was given during the test with respect to the correct responses.

Subjects could practise lessons for as long as they wanted. In the "Menu condition" and the "Guidance condition" they were instructed to press a "Quit lesson" button whenever they wanted to study another lesson. In the Menu condition, pressing this button resulted in the appearance of a menu screen as shown in Figure 60. From the menu, the subjects could select a lesson. In the Guidance condition, pressing the button resulted in the automatic selection of a lesson according to the algorithm described above. In the "Mixed condition" subjects could use both a menu and guidance. They were instructed to press a "Menu" button whenever they wanted to select a lesson through the menu, and to press a "Pick lesson" button whenever they wanted a lesson to be automatically selected for them. The effect of pressing the Menu button was exactly the same as pressing the Quit lesson button in the Menu condition, and the effect of pressing the Pick lesson button was exactly the same as pressing the Quit lesson button in the Guidance condition. In all conditions, the name and number of the current lesson was always shown in the left upper corner of the screen (as in Figure 61).

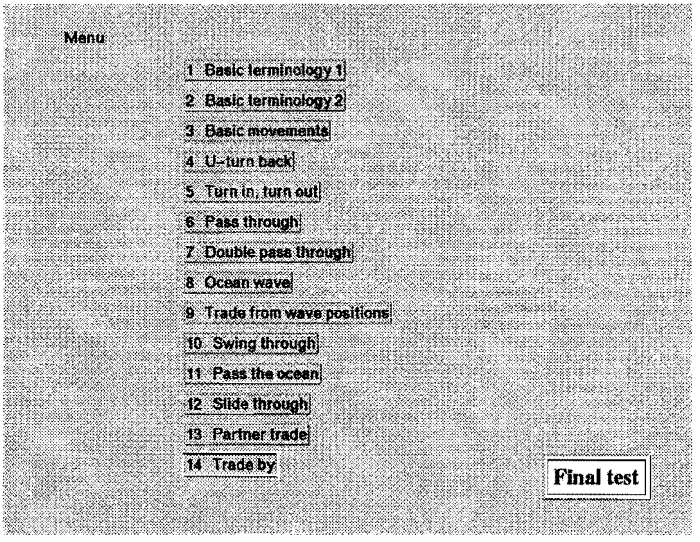


Figure 60: *Screen layout of the menu of the Square Dance course*

The times were recorded at which subjects decided to change lesson, selected a lesson, and selected the test. It was also recorded which lesson was selected by the subject or the agent, which exercise items were studied and whether the subject responded correctly to the exercise or test item.

Immediately after the experimental session, subjects had to fill out a questionnaire in which they could indicate any problems they had experienced during the course, how well they liked it, why they selected the lessons in that particular order (in the Menu and Mixed condition), how they liked the order selected for them (in the Guidance and Mixed condition), and when they used the menu and when the automatic selection (in the Mixed condition).

Subjects

Forty-two subjects with university or higher vocational training participated voluntarily in the experiment. The average age was 24. All subjects had some computer experience. None of the subjects had ever heard of Square Dancing before. Subjects were randomly assigned to one of the three experimental conditions.

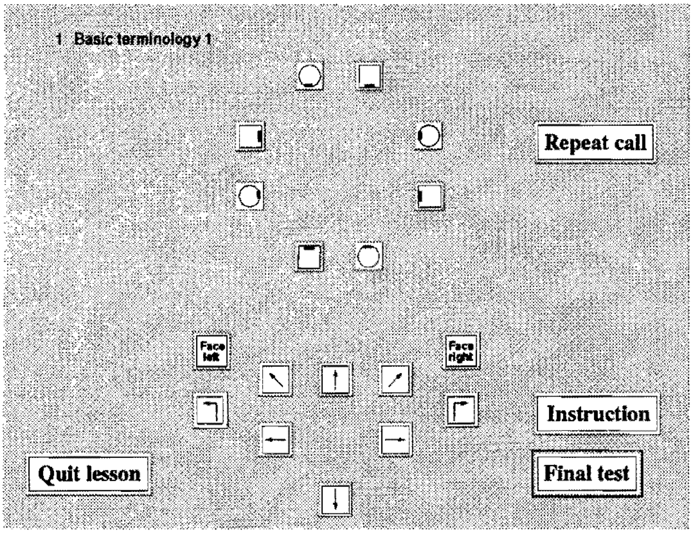


Figure 61: Screen layout of an exercise of the Square Dance course

Materials

Square Dance is an American dance that is performed by a group of eight dancers consisting of four couples. In the starting position, the couples stand around the sides of an imaginary square. Dancers move according to the instructions given by a "caller". In the computer course, subjects had to move the dancers by remote control according to the instructions they heard. The understandability of the interface and the course material had been tested in pilot studies and was improved accordingly (Claessens, 1996).

The course consisted of fourteen lessons. Each lesson comprised an instruction part and an exercise part. The first two lessons explained some basic terminology, e.g., that circles represent girls and squares boys, and that a dancer can be activated by clicking on it. In the third lesson, the remote control was explained and the basic movements of the dancers were practised. In the other lessons, more advanced dance movements were presented. According to an experienced square dancer, the order in which the lessons were presented in the menu is a reasonable order for a student who has to learn all the movements present in the course. The sequence determined by the topic selection algorithm (see Section 6.3) consisted of Lesson 1, 2, 3, 6, and 12. The instructions of Lesson 12 gave an indication that Lesson 6 was prerequisite.

The interface of an exercise is shown in Figure 61. The eight small buttons in the upper part of the screen represent the dancers, the ten small buttons in the lower part of the screen constitute the remote control.

6.4.2 Results: General

The results of the experiment are shown in Figure 62 and 63, for the 'total time', and the time spent on each lesson, respectively. The total time was calculated from the moment the subjects started the course after reading the general instructions till the moment they started the last test. The mean and standard deviations of the total time were m=33'42", sd=14'3" for the Menu condition, m=27'12", sd=9'11" for the Guidance condition, and m=27'24", sd=12'23" for the Mixed condition.

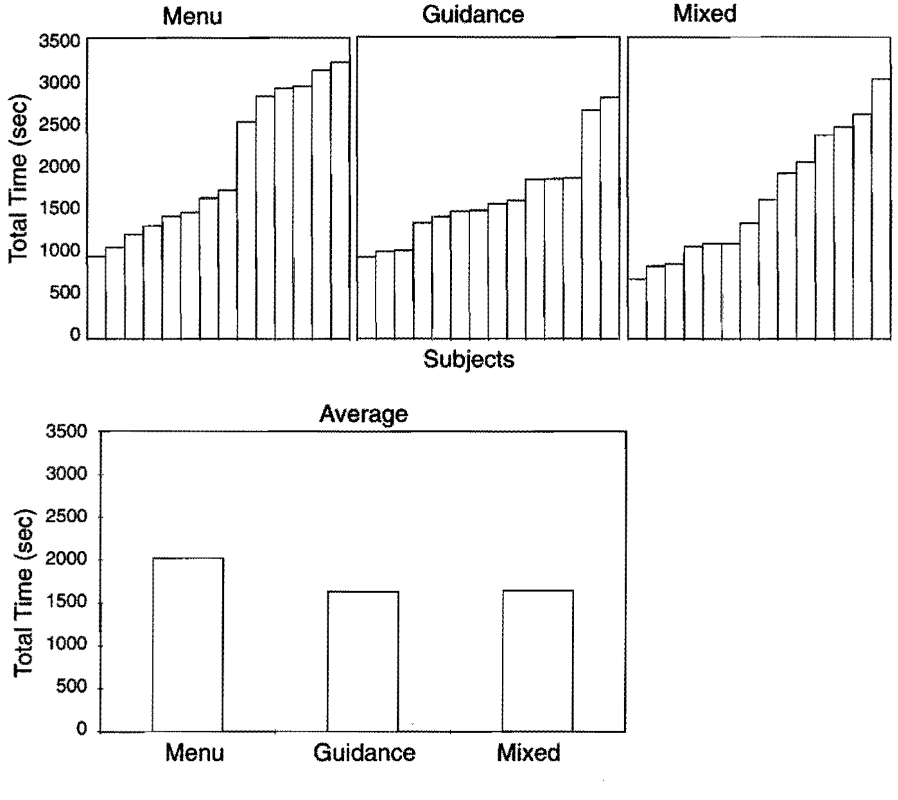


***Figure 62:** Total times in Experiment 1 per subject for the Menu condition, the Guidance condition, and the Mixed condition, respectively, and the average over the conditions. Each bar in the top graphs represents a subject. The bars in the bottom graph represent the average over subjects. The total time is given on the y-axis.*

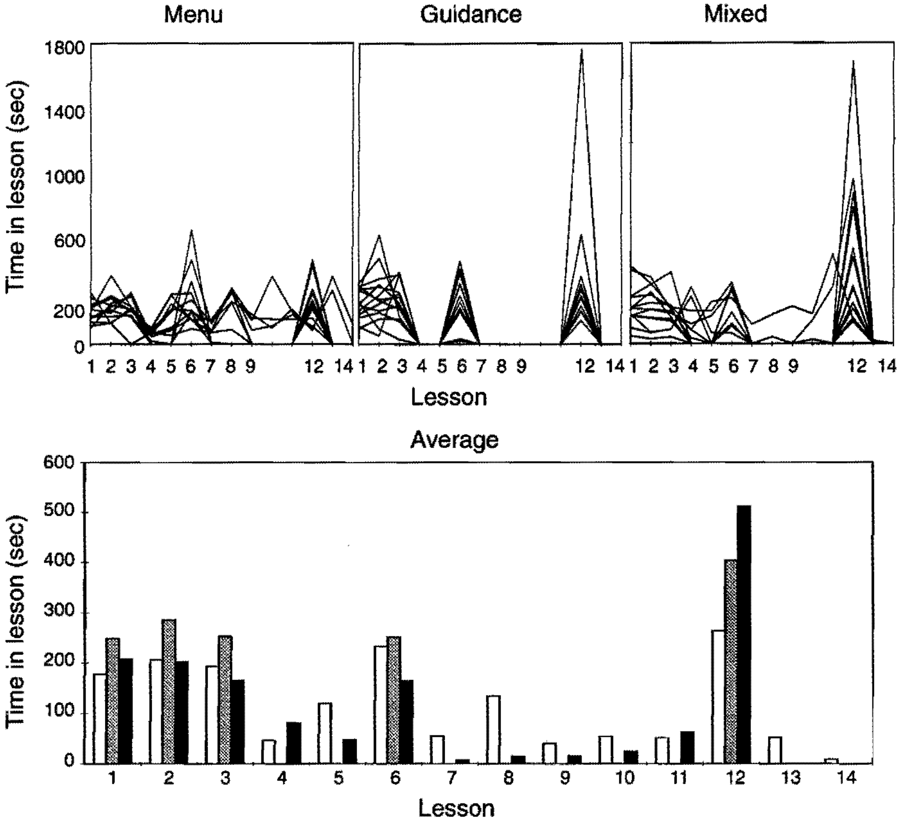


Figure 63: *Times per lesson in Experiment 1 per subject for the Menu condition, the Guidance condition, and the Mixed condition, respectively, and the average over both conditions. Each line in the top graphs represents a subject. The bars in the bottom graph represent the average over subjects in the Menu condition (white), the Guidance condition (gray), and the Mixed condition (black), respectively. The lessons are given on the x-axis and the time spent in that lesson is given on the y-axis.*

Some representative paths followed through the course material are shown in Figure 64. In the Menu condition, a lot of subjects followed the lessons in the order of the menu till they decided to select the lesson on the Slide through (e.g., see the third path for the Menu condition in Figure 64). Three subjects started with the lesson on the Slide through, like the subject of the first path for the Menu condition in Figure 64. The Quit lesson button was used on average 10.2 times, excluding the selection of the first lesson. On average 9 minutes and 23 seconds were spent on superfluous lessons, where "superfluous" was defined as lessons other than 1, 2, 3, 6, and 12.

In the Guidance condition, all the subjects followed the same path, which is shown in Figure 64. The Quit lesson button was used on average 6.6 times, excluding the selection of the first lesson.

In the Mixed condition, three subjects always used the Menu button, three subjects always used the Pick lesson button, and eight subjects used both. When both were used, the Menu button was used frequently to select an easier lesson (see the two paths shown in Figure 64). On average[1], the Menu button was used 4 times and the Pick lesson button 3.3 times, excluding the selection of the first lesson. On average, 4 minutes and 17 seconds were spent on superfluous lessons.

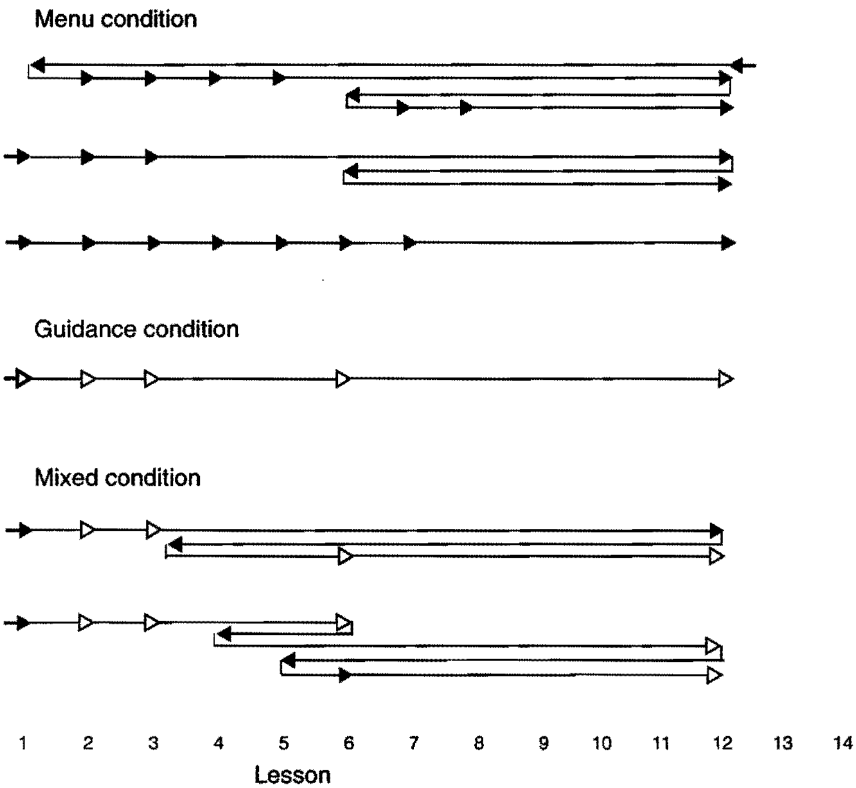


Figure 64: *Some paths through the course material for the Menu condition, the Guidance condition, and the Mixed condition, respectively. The heads of the arrows indicate whether the lesson was selected by the subject (filled) or by the agent (hollow).*

1. One subject is excluded who used the Menu button 36 times and the Pick lesson button 28 times is excluded from the average.

A MANOVA was performed on the total time and the time spent on each lesson, with the experimental condition (Menu, Guidance, Mixed) as a between-subjects factor. The three experimental conditions were contrasted pairwise. The results of this analysis are summarized in Table 28.

Table 28: *Results of the MANOVA on the data of Experiment 1, with Condition as the source, and Df=1.*[a]

Dep. var.	F		Dep. var.	F		Dep. var.	F
Menu Condition versus Guidance Condition							
Total time	2.02		Lesson 5	14.18	***	Lesson 10	3.83
Lesson 1	3.34		Lesson 6	0.09		Lesson 11	1.67
Lesson 2	2.50		Lesson 7	14.48	***	Lesson 12	1.06
Lesson 3	2.28		Lesson 8	15.05	***	Lesson 13	3.20
Lesson 4	3.13		Lesson 9	4.08		Lesson 14	1.58
Menu Condition versus Mixed Condition							
Total time	1.90		Lesson 5	5.03	*	Lesson 10	1.16
Lesson 1	0.60		Lesson 6	1.23		Lesson 11	0.07
Lesson 2	0.01		Lesson 7	10.47	**	Lesson 12	3.30
Lesson 3	0.51		Lesson 8	11.80	**	Lesson 13	3.04
Lesson 4	1.73		Lesson 9	1.48		Lesson 14	1.58
Guidance Condition versus Mixed Condition							
Total time	0.00		Lesson 5	2.32		Lesson 10	0.77
Lesson 1	1.11		Lesson 6	1.97		Lesson 11	2.40
Lesson 2	2.80		Lesson 7	0.32		Lesson 12	0.62
Lesson 3	4.96	*	Lesson 8	0.20		Lesson 13	0.00
Lesson 4	9.52	**	Lesson 9	0.64		Lesson 14	0.00

$^{*}p<.05$, $^{**}p<.01$, $^{***}p<.001$

a. With the Student-Newman-Keuls method, the same effects were found to be significant with a familywise α–level of .05 except the effect on the third lesson, which is not important for the discussion below. With the Bayesian approach of Waller and Duncan exactly the same effects were found to be significant as above.

There is no significant effect of the condition on the total time for any of the contrasts. For the contrast between the Menu condition and the Guidance condition, there are significant effects of the condition on the time spent on the fifth, seventh, and eighth lessons [$F(1,1)=14.18$, $p<.001$; $F(1,1)=14.48$, $p<.001$; $F(1,1)=15.05$, $p<.001$]. For the contrast between the Menu condition and the Mixed condition, there are also significant effects of the condition on the time spent on precisely these lessons [$F(1,1)=5.03$, $p<.05$; $F(1,1)=10.47$, $p<.01$; $F(1,1)=11.8$, $p<.01$]. For the contrasts between the Guidance condition and the Mixed condition, there are significant effects of the condition on the time spent on the third and the fourth lessons [$F(1,1)=4.96, p<.05$; $F(1,1)=9.52, p<.01$].

6.4.3 Discussion: General

There is no significant effect of the experimental condition on the total time. However, there is certainly a tendency in favour of both the Guidance condition and the Mixed condition (see means and Figure 62): on average, six and a half minutes more were needed in the Menu condition. The large variance between subjects may partly be due to the time measure we used: subjects had to decide when to try the final test for themselves.

The results of the individual subjects in the Guidance condition are much more homogeneous than in the other two conditions (see standard deviations and Figure 62). The top graphs of Figure 62 suggest that, especially in the Menu condition, the group of subjects can be divided into two groups: a slow group and a fast group. An explanation may be that in the Menu condition some subjects merely follow the sequence of the menu from top to bottom, while other subjects are much better at deciding for themselves which lessons are relevant, given the objective of learning the Slide through. The heterogeneous results in the Mixed condition may be explained by some subjects being more reluctant to use guidance than others. These explanations are confirmed by the navigation paths (such as shown in Figure 64), and by the time spent per lesson for each individual subject as shown in Figure 63. A post-hoc analysis was performed in which the subjects of both the Menu and the Mixed condition were divided into good and poor navigators (see below).

The Menu condition has the disadvantage that a great deal of time was spent on superfluous lessons: on average almost nine and a half minutes, which is one third of the total time in that condition. This is confirmed by the significant effect of condition on the fifth, seventh, and eighth lessons in the comparison with both of the other conditions. The absence of a significant effect on the fourth lesson is probably due to the relative simplicity of that lesson. The absence of a significant effect on the tenth, eleventh, thirteenth, and fourteenth lessons is due to the fact that only a few subjects spent any time on these lessons (see the top-left graph in Figure 63). This may be explained by the subjects realizing that it would take a long time to do all the lessons before trying the twelfth lesson or the test.

The Guidance condition has the advantage that only relevant lessons are studied as a natural consequence of the topic selection algorithm used. It is striking to note that, on average, in the Guidance condition subjects spent a relatively long time on the first three lessons (see the bottom graph in Figure 63). From Figure 63 (middle top graph) it can be concluded that this is due to only a few subjects. This explains why the effect is not significant (see Table

28). An explanation of the long time spent on the first lessons may be that subjects expect that the agent will not only select a new lesson, but can also take the initiative for the selection. The effect of the agent taking the initiative for the selection will be investigated in the next experiment.

The Mixed condition has the advantage that less time is spent on superfluous lessons than in the Menu condition (five minutes less) and also less time is spent on the first lessons than in the Guidance condition (three and a half minutes less). In the Mixed condition subjects may be more aware of the need to decide on the moment of navigation for themselves, because they also have the possibility to select the topic themselves, as in the Menu condition.

Strikingly, in both the Guidance condition and the Mixed condition more time was spent, on average, on the twelfth lesson than in the Menu condition. However, there was no significant effect of condition on that lesson. In the Guidance condition this effect is clearly due to one subject (see the middle top graph in Figure 63). The performance of this subject illustrates a problem which many subjects in this condition complained about in the questionnaire: it seemed impossible to them to obtain an easier lesson. This particular subject used the "Quit lesson" button in the instruction part of the lessons, without doing any exercises, till she reached the twelfth lesson and did not manage to quit that lesson. She did not realize that it would make a difference if she tried to quit the lesson after having done some of the exercises, and spent a long time doing exercises in this most difficult lesson. This problem suggests that it is incorrect always to select a more difficult lesson when topic selection is requested while the student is reading the instructions of a lesson for the first time.

In the Mixed condition, on the other hand, the long time spent on the twelfth lesson was caused by four of the subjects. This can be explained by the fact that these subjects experienced more difficulty with this lesson, because three of them did not spend any time on the sixth lesson. All of these subjects tended to use the Menu for navigation.

6.4.4 Results: Good navigators versus poor navigators

A post-hoc analysis was performed in which the subjects of both the Menu condition and the Mixed condition were divided into two groups: one group with subjects who studied at least two superfluous lessons, and one group with the remaining subjects. We will call the first group the poor navigators and the second group the good navigators. In the Menu condition nine subjects belonged to the poor navigators, and five subjects to the good navigators. In the Mixed condition four subjects belonged to the poor navigators, and ten subjects to the good navigators.

The average results per group are shown in Figure 65. The results of the MANOVA on these groups is summarized in Table 29.

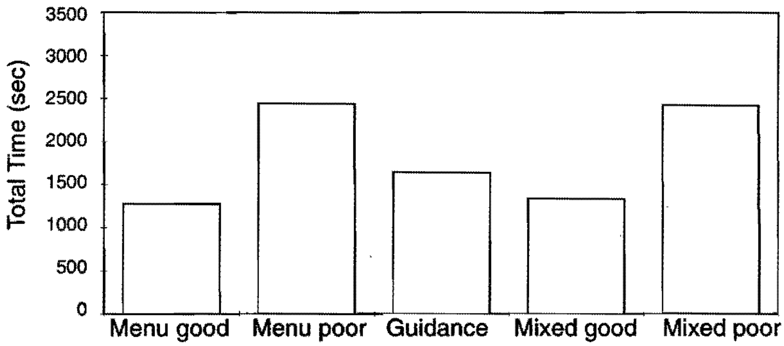


Figure 65: *Total times in Experiment 1 for the average over the conditions with the Menu condition and the Mixed condition divided into two groups. Each bar represents the average over subjects. The total time is given on the y-axis.*

There is a significant effect of condition on the total time for the contrasts between the poor navigators and the good navigators of the Menu and Mixed condition [$F(1,1)=22.05$, $p<.0033$], and the poor navigators of the Menu and Mixed condition and the Guidance condition [$F(1,1)=11.31$, $p<.00033$]. The contrast between the good navigators of the Menu and Mixed condition and the Guidance condition was not significant.

Table 29: *Results of the MANOVA on the total time in Experiment 1 with a division into two groups of the Menu condition and the Mixed condition.*[a]

Source	Num DF	Den DF	F
Poor navigators vs. Good navigators	1	1	22.05 **
Poor navigators vs. Guidance	1	1	11.31 ***
Good navigators vs. Guidance	1	1	2.12

***$p<.017$, **$p<.0033$, ***$p<.00033$**

a. According to the Bonferroni principle, effects are significant with a familywise α–level of .05, .01, or .001, respectively, when $p<.017$, $p<.0033$, or $p<.00033$, respectively.

6.4.5 Discussion: Good navigators versus poor navigators

The significant effect of condition for the two above-mentioned contrasts indicates, in combination with Figure 65, that the poor navigators of the Menu condition and the Mixed condition need more time than the good navigators and the subjects in the Guidance condition. This implies an advantage of using guidance for students who are unable to monitor their own learning process (like low performers or children). As nine of the fourteen subjects of the Menu condition are poor navigators, determining a good path through the course material seems a difficult task, so, guidance seems necessary.

It can be argued that subjects had only limited information available for the navigation. On the other hand, the number of lessons in this course was very limited (only 14), the foreknowledge relations between the lessons was limited, and the lessons needed to reach the goal were quite easy to find: Lesson 12 had the same name as the goal and mentioned the need for Lesson 6, and besides these two lessons only the first lessons were needed, which all had a name beginning with "basic". So, perhaps some improvement in performance can result from providing the subjects with more information, but on the other hand this information may be more difficult to provide and performance can be expected to decrease when the number of lessons increases.

The lack of a significant effect for the contrasts between the good navigators and the subjects of the Guidance condition suggests that the guidance works quite well, even though subjects complained that it was not possible to return to the easier lessons. Figure 65 suggests a small advantage for the Menu condition, but it is likely that the good navigators are also the better learners, so, this effect is negligible.

6.5 Timing of guidance

The Navigation Agent should take the initiative to navigate whenever the student is performing very well or very poorly on the current lesson. There are several ways in which "very well" and "very poorly" can be defined, for instance by using the number of successive correct or incorrect responses, or by looking at asymptotic performance. Inspired by the concept of a good set (see Chapter 3 for an explanation of the good set), we have chosen the following definitions.

The student's performance in a particular lesson is considered to be "very good" if the student has answered the item last presented in the exercise of that lesson correctly and at least 80% of the items in that lesson have been mastered, in the sense that they belong to the good set. The student's performance in a particular lesson is considered to be "very poor" if the student has made more than three mistakes in the exercise of that lesson, when the item last presented was answered incorrectly, and at most 20% of the items of that lesson have been mastered, in the sense that they belong to the good set.

As a consequence of these definitions, the agent will only take the initiative to navigate when the student is doing exercises, and not when the student is reading instructions. More sophisticated mechanisms might include the time spent on the instructions relative to the time spent on the exercise, and the number of times the instructions are consulted.

6.6 Experiment 2: timing of guidance

6.6.1 Method

Design and Procedure

The same design and procedure were used as in Experiment 1, except that there were two conditions: a Guidance condition and a Mixed condition. In contrast to the previous experiment, the initiative for the selection of another topic could be taken both by the Navigation Agent and the user. The agent used the criteria as described above to decide when guidance was needed. To be able to compare these conditions with the conditions of the previous experiment the Guidance condition of this experiment will be denoted by TGuidance, and the Mixed condition by TMixed.

Subjects

Fourteen subjects with university education participated voluntarily in the experiment. The average age was 26. All subjects had some computer experience. None of the subjects had ever heard of Square Dancing before. Subjects were randomly assigned to one of the two experimental conditions, seven to each condition.

Materials

The same materials were used as in Experiment 1, except that the instructions were changed to inform the subjects that the system might also take the initiative.

6.6.2 Results

The results of the experiment are shown in Figure 66 and 67 for the total time and the time spent in each lesson, respectively. The mean and standard deviations of the total time were m=22'50", sd=8'32" for the TGuidance condition, and m=22'3", sd=3'37" for the TMixed condition.

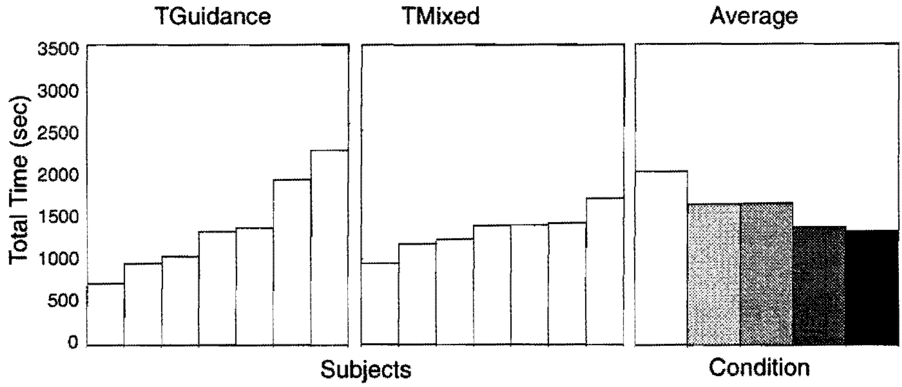


Figure 66: Total times in Experiment 2 per subject for the TGuidance condition and the TMixed condition, respectively, and the average over both these conditions and the conditions of Experiment 1. Each bar in the two left-hand graphs represents a subject. The bars in the right-hand graph represent the average over subjects in the Menu condition (white), the Guidance condition (light gray), the Mixed condition (gray), the TGuidance condition (dark gray), and the TMixed condition (black), respectively. The total time is given on the y-axis.

In the TGuidance condition, the Quit lesson button was used 1.7 times on average (excluding the selection of the first lesson), and the use of this button always resulted in the selection of the current lesson. The agent took the initiative for topic selection 4 times on average . All subjects followed the path 1-2-3-6-12 through the course material.

In the TMixed condition, the Menu button was used 1.7 times on average and the Pick lesson button 3.9 times (excluding the selection of the first lesson). The Menu button was mostly used to return to an easier lesson, and the use of the Pick lesson button often resulted in the selection of the current lesson. The agent took the initiative for topic selection 3.3 times on average. Only one subject spent some time on superfluous lessons.

The data of the previous experiment were included for the analysis of these data. A MANOVA was performed on the total time and the time spent on each lesson, with the experimental condition as a between-subjects factor. Three contrasts were performed: between the Menu condition and the TGuidance and TMixed conditions, between the Guidance and Mixed conditions and the TGuidance and TMixed conditions, and between the TGuidance condition and the TMixed condition. The results of this analysis are summarized in Table 30.

There is a significant effect of the condition on the total time for the contrast between the Menu condition and the TGuidance and TMixed condition

[*F*(1,1)=7.17, *p* <.017]. There is no significant effect of the condition on the total time for the other two contrasts.

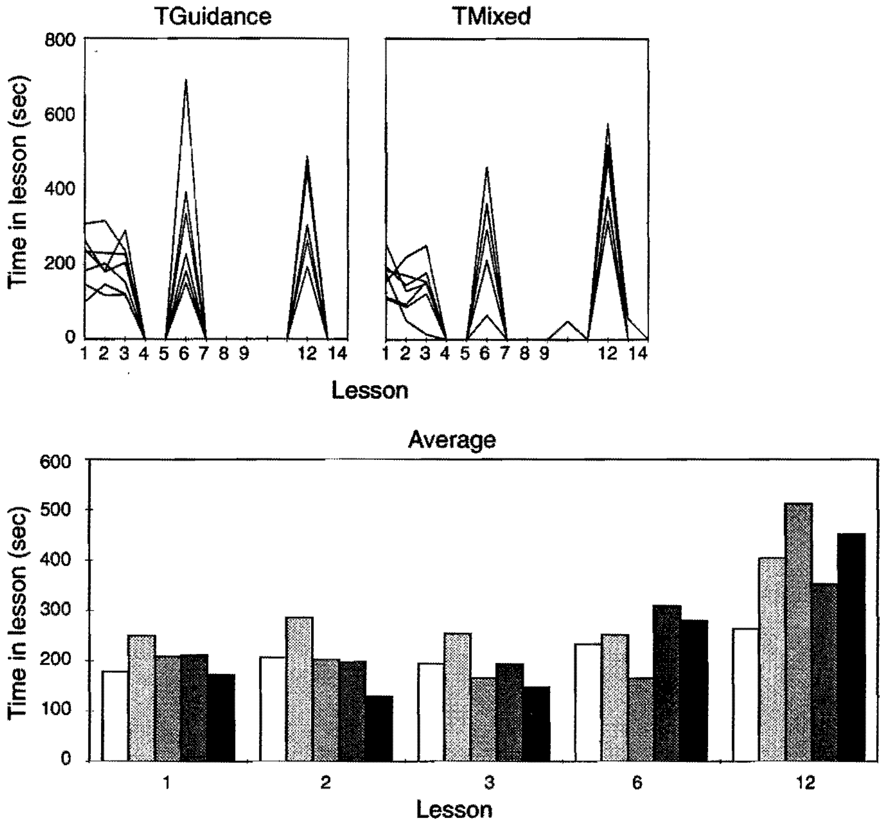


Figure 67: *Times per lesson in Experiment 2 per subject for the TGuidance condition and the TMixed condition, respectively, and the average over both conditions. Each line in the two top graphs represents a subject. The bars in the bottom graph represent the average over subjects in the Menu condition (white), the Guidance condition (light gray), the Mixed condition (gray), the TGuidance condition (dark gray), and the TMixed condition (black), respectively. The lessons are given on the x-axis, and the time spent on that lesson is given on the y-axis.*

For the contrast between the Menu condition and the TGuidance and TMixed conditions, there are significant effects of the condition on the time spent on the fifth, seventh, and eighth lessons [*F*(1,1)=18.55, *p* <.00033; *F*(1,1)=18.93, *p* <.00033; *F*(1,1)=19.69, *p* <.00033]. There are no significant effects of the condition on the time spent on any lesson for the other two contrasts.

Table 30: Results of the MANOVA on the data of Experiment 2, with Condition as the source, and Df=1.[a]

Dep. var.	F		Dep. var.	F		Dep. var.	F
Menu Condition versus TGuidance and TMixed Conditions							
Total time	7.17	*	Lesson 5	18.55	***	Lesson 10	4.34
Lesson 1	0.13		Lesson 6	1.00		Lesson 11	2.18
Lesson 2	0.03		Lesson 7	18.93	***	Lesson 12	1.30
Lesson 3	0.41		Lesson 8	19.69	***	Lesson 13	3.52
Lesson 4	4.10		Lesson 9	5.34		Lesson 14	2.07
Guidance and Mixed Conditions versus TGuidance and TMixed Conditions							
Total time	1.78		Lesson 5	1.01		Lesson 10	0.17
Lesson 1	1.47		Lesson 6	2.64		Lesson 11	1.05
Lesson 2	4.31		Lesson 7	0.14		Lesson 12	0.29
Lesson 3	1.52		Lesson 8	0.09		Lesson 13	0.02
Lesson 4	4.15		Lesson 9	0.28		Lesson 14	0.00
TGuidance Condition versus TMixed Condition							
Total time	0.02		Lesson 5	0.00		Lesson 10	0.04
Lesson 1	0.62		Lesson 6	0.11		Lesson 11	0.00
Lesson 2	1.14		Lesson 7	0.00		Lesson 12	0.33
Lesson 3	0.79		Lesson 8	0.00		Lesson 13	0.05
Lesson 4	0.00		Lesson 9	0.00		Lesson 14	0.00

***$p<.017$, **$p<.0033$, ***$p<.00033$**

a. According to the Bonferroni principle, effects are significant with a family-wise α–level of .05, .01, or .001, respectively, when $p<.017$, $p<.0033$, or $p<.00033$, respectively.

6.6.3 Discussion

The effect of condition on the total time is not significant for the contrast between the Guidance and Mixed conditions and the TGuidance and TMixed conditions. However, there is certainly a tendency in favour of the TGuidance and the TMixed conditions (see means and Figure 66): on average Guidance took 4'22" more than TGuidance, and Mixed took 5'21" more than TMixed. This is supported by the fact that there is a significant effect of condition on the total time for the contrast between the Menu condition and the TGuidance and TMixed conditions, while the contrasts between the Menu and Guidance conditions and between the Menu and Mixed conditions were not significant. So, the initiative of the Navigation Agent with respect to the moment of navigation seems to decrease the time needed to complete the task.

On average, less time was spent on the first three lessons in the TGuidance and Tmixed conditions than in the Guidance and Mixed conditions, respectively (see Figure 67): 10'1" compared to 13'8", and 7'26" compared to 9'35", respectively. There was no significant effect of condition on the time spent on any of these lessons for this contrast. However, it suggests that the problem posed by some of the subjects in the Guidance condition spending too much time on the first lessons has disappeared.

In the TMixed condition, in contrast to the Mixed condition, hardly any time was spent on superfluous lessons: only one subject spent some time on these. A possible explanation is that the agent frequently took the initiative for the selection of another lesson and that the subjects agreed with that selection. This is confirmed by the fact that the agent took the initiative 3.3 times on average, and that the Menu button was less often used than in the Mixed condition: 1.7 times on average in the TMixed condition compared to 4 times in the Mixed condition.

The significant effect of condition on the time spent on the fifth, seventh, and eighth lessons for the contrast between the Menu condition and the TGuidance and TMixed conditions merely illustrates that less time was spent on superfluous lessons in the last two conditions. This is the same effect as was found in the previous experiment for the contrasts between the Menu condition and the Guidance and Mixed condition.

On average, more time was spent on the sixth lesson in the TGuidance and TMixed condition than in the Guidance and Mixed condition respectively (see Figure 67): 5'9" compared to 3'53", and 4'40" compared to 2'44". There was no significant effect of condition on the time spent on this lesson for this contrast. However, it suggests that the cause of the difficulty which some of the subjects experienced with the twelfth lesson, especially in the Mixed condition –namely that they did not spend enough time on the sixth lesson– has disappeared. Again, a possible explanation may be that the agent frequently took the initiative for the selection of another lesson, like the sixth lesson, and that the subjects agreed with that selection. Another explanation may be that the subjects explicitly waited until the agent took the initiative to leave the sixth lesson, thereby showing an increased motivation for studying the lesson. Both explanations are supported by the data. Firstly, now all subjects spent time on the sixth lesson, while in the Mixed condition three subjects did not study that lesson at all. Secondly, the average number of exercises performed in that lesson has increased.

On average, less time was spent on the twelfth lesson in the TGuidance and TMixed conditions than in the Guidance and Mixed conditions, respectively. On the one hand, we expected this decrease in time, because the two main problems causing a long stay in that lesson were removed. Firstly, in the TGuidance and TMixed conditions, the agent could take the initiative to select an easier lesson. This did not occur, however. But in the TMixed condition the Pick lesson button was used relatively more frequently compared to the Menu button than in the Mixed condition, and an easier lesson was selected automatically three times. Secondly, the time spent on the sixth lesson had increased, thus decreasing the added complexity of the twelfth lesson.

On the other hand, we expected an increase in the time spent on the twelfth lesson, as the initiative of the agent may lead to the incorrect expectation that the agent can also determine the moment of the test. This may explain why the time spent on the twelfth lesson in the TGuidance and TMixed condition is still longer than in the Menu condition. If this is the cause, then it is a pure experimental effect, in the sense that it is not relevant for a real application.

The results of the individual subjects in the TMixed condition are much more homogeneous than in the other conditions (see standard deviations and Figure 62 and 66). The variance may have been reduced by a combination of two factors, namely a stimulation due to the initiative of the agent to spend enough time studying the relevant lessons (this seemed to be lacking in the Mixed condition) combined with a possibility to use the Menu button for fast consultation of a specific easier lesson (this is lacking in both the Guidance and TGuidance conditions). This suggests that the TMixed condition should be preferred to the TGuidance condition.

6.7 Conclusions

A new approach has been described in which initiative and topic selection by the Navigation agent can be combined with initiative and topic selection by a student. The topic selection of the agent is based on the foreknowledge, goals, and capabilities of the individual student. The moment of initiative is based on the student's performance.

In two experiments the effectivity of both topic selection and initiative by the agent was explored. Topic selection by the agent had an advantage for students who were unable to monitor their own learning process. Initiative by the agent reduced the time needed to finish the task by stimulating the subjects to spend enough time studying the relevant lessons.

Both guidance initiated by the agent and a mix of user navigation and guidance are interesting new functionalities as they lead to a reduction in learning time. The results of the experiments suggest that a mixed locus of control is preferable, in which both the student and the agent can take the initiative and select another lesson. This reduces the variance between students, thus making the learning process more predictable. The groups of subjects used in these experiments were already quite homogeneous. So, the effect will probably be even larger when the groups are less homogeneous.

There are several directions for further research. In the first place, research should be done on how quickly the Navigation Agent can discover deviations from the initially assumed foreknowledge and goals, and how fast it can adjust the difficulty of the path through the course material to the abilities of the individual student. For that purpose, simulations of students may also be used. In the second place, the algorithm for topic selection should be tested in more complex domains than the simple one used in the experiments. In the third place, research should be done to further improve the timing of initiative, especially with respect to the initiative to return to an easier lesson. Finally, the effect of providing the student with more information for self-navigation should be investigated.

References

Claessens, M. (1996). *Ergonomic evaluation of the Square Dance program.* Unpublished manuscript.

Espinosa, R. & Baggett, P. (1993). A design for accessing multimedia information using cohesion. In M.D. Brouwer-Janse and T.L. Harrington (Eds.), *Human-machine communication for educational systems design.* (Volume F129 of NATO ASI Series). Berlin: Springer.

Jochems, W. (1980). *Leerstofanalyse.* Delft: Delftse Unversitaire Pers.

Gentner, D. (1992). Interfaces for learning: Motivation and the locus of control. In F.L. Engel, D.G. Bouwhuis, T. Bösser, & G. d'Ydewalle (Eds.), *Cognitive modelling and interactive environments in language learning* (Volume F87 of NATO ASI Series). Berlin: Springer.

Hensgens, J., Rosmalen, van P., & Baaren, van der J. (1995). *Authoring for simulation-based learning. Instructional Science, 23,* 269-296.

Koehorst, A., Baaren, van der J., & Rosmalen, van P. (1991). Authoring planning for educational systems. *Proceedings of the sixth international PEG conference.* Genova: Instituto per la matematica applicata, CNR.

Laurel, B., Oren, T., & Don, A. (1990). Issues in multimedia interface design: media integration and interface agents. *Proceedings of the ACM CHI '90 conference on human factors in computing systems*, 133-139.

Norcio, A.F., & Stanley, J. (1988). Adaptive human-computer interfaces. *Naval Research Laboratory Report 9148.*

Pask, G., Kallikourdis, D., & Scott, B.C.E. (1975). The representation of knowables. *International Journal of Man-Machine Studies, 7,* 15-134.

Rich, E. (1989). Stereotypes and user modelling. In A. Kobsa & W. Wahlster (Eds.), *User models in dialogue systems*. Berlin: Springer Verlag.

Robertson, G.G., Mackinlay, J.D., & Card, S.K. (1991). Cone trees: Animated 3d visualizations of hierarchical information. *Proceedings of the ACM CHI '91 conference on human factors in computing systems*, 189-194.

Takeuchi, A., & Otsuki, S. (1990). Intelligent tutoring system: Coordinating mixed initiative with strategic dialogue by soft scenario. *The transactions of the IEICE, E73*, 308-314.

Chapter 7

The Explanation Agent, Feedback Agent, and Presentation Agent

Abstract

Based on examples of the interaction between students and the current Appeal prototype, further insight is given into the behaviour of the Explanation Agent, the Feedback Agent, and the Presentation Agent, and the theoretical foundation of that behaviour. These three simple agents maintain the instruction dialogue with the student, and are perceived by the student as one person. No experiments have been performed with these Agents as yet, but some directions in which experiments should be performed are indicated.

7.1 The instruction dialogue

The previous two agents, namely the Practice Agent and the Navigation Agent, were both concerned with sequencing: the topics and the items within the topics were presented to the students in an order adapted to their interests and capabilities (see Chapter 3,4, 5, and 6). The three agents described in this chapter, namely the Explanation Agent, the Feedback Agent, and the Presentation Agent, maintain the instruction dialogue with the student. From these three different, simple agents a dialogue partner for the student emerges, which ensures a sensible level of communication. The student does not perceive the different agents, but only one person who is a kind of teacher and gives a dynamic and understanding impression.

The communication takes place regardless of whether the student learns or not. The Explanation Agent determines the amount and timing of the information to be presented to the student. The Feedback Agent determines the feedback on actions of the student. The Presentation Agent determines the form in which the information is presented, for instance a formula or a picture.

To avoid schizophrenic actions of the emerging teacher, the Feedback Agent has been designed to react faster than the other two agents. In its turn, the Presentation Agent has been designed to react faster than the Instruction Agent.

7.2 The Explanation Agent

As described in Chapter 2 (Section 2.6.2), the main task of the Explanation Agent is to give explanations adapted to the student whenever the student needs them. The behaviour of the Explanation Agent should be contingent on the performance of the student (cf., situatedness principle); the most appropriate intervention is chosen on the basis of the student's success.

7.2.1 Theory underlying the Explanation Agent

In Chapter 1 (Section 1.2.3), an overview was given of research indicating aspects of explanation that should be adapted to the student. This included the amount and timing of explanations (for references see Chapter 1).

According to Wood, Wood, and Middleton (1978), the content of explanation should be based on a layered model of intervention varying between the "general verbal encouragement" level and the "demonstration" level, whereby each layer represents a different level of control from the side of the agent. So, the amount of explanation should depend on the overall performance of the student.

For the purpose of explanation, communication is needed. Layered protocol theory (Taylor, 1988) provides a model for the structuring of communication. According to layered protocol theory information should be provided at the appropriate layer of abstraction: always providing the information needed at that point of the communication. It implies that the instruction dialogue is finished successfully.

In any case, the explanation should satisfy the Gricean maxim of quantity (Grice, 1975): the minimal amount of information should be provided as needed for successful communication.

The application of these ideas is, however, no easy task. A distinction has to be made between several kinds of information. In the first place, there is *content information* which is learning-domain specific. This category encompasses, for instance, explanations on how clock times are expressed in Dutch, or which word can be put into which frame in a jigsaw exercise.

In the second place, there is *interface information* which explains the proper use of the user interface and is not necessarily learning-domain specific. This category encompasses, for instance, explanations on how to select an alternative in a multiple choice exercise, or how to perform a transformation exercise.

In the third place, there is *meta information* which is an explanation about obtaining information or the type of information that will be provided. This category encompasses, for instance, explanations on how to get help, or how to see an example again.

For all types of information, an adequate level of concreteness has to be determined. For instance, a student may understand the interface of an exercise, but may still need explanation on the subject matter. On the other hand, it may also happen that a student understands the subject matter, but does not understand the interface of an exercise. Information is omitted when it has been provided before and the student has demonstrated an understanding of the information successively, or not much time has passed since the last presentation of the information.

As far as the timing of the explanation is concerned, this depends on changes of situation, such as the navigation to a new topic, which are caused by the other agents or the student, and on the performance of the student. A lack of action from the student is also considered as a situation in which instruction is needed.

The choice between instruction strategies –like, for instance, the choice between first giving the rules on how clock times are expressed in Dutch and then letting the students practise, and letting the students discover for themselves how clock times are expressed in Dutch– is currently not made by the

Explanation Agent, but a natural consequence of the information provided by the course designer.

7.2.2 Example of the Explanation Agent's behaviour

The following fragments show interactions between a student and the Appeal prototype system for learning Dutch by English-speaking persons. They illustrate the behaviour of the Explanation Agent as it is currently implemented. E denotes the Explanation Agent, S denotes the student, and P denotes the Practice Agent.

In the first fragment the student is presented with a jigsaw exercise on the use of the passive form in Dutch: a sentence is presented, and the student has to transform it into the passive form.

(E1) *"The subject of the following exercises is verb forms."*
(E2) *"The type of this exercise is transformation."*
(E3) *"Look carefully at the following example."*

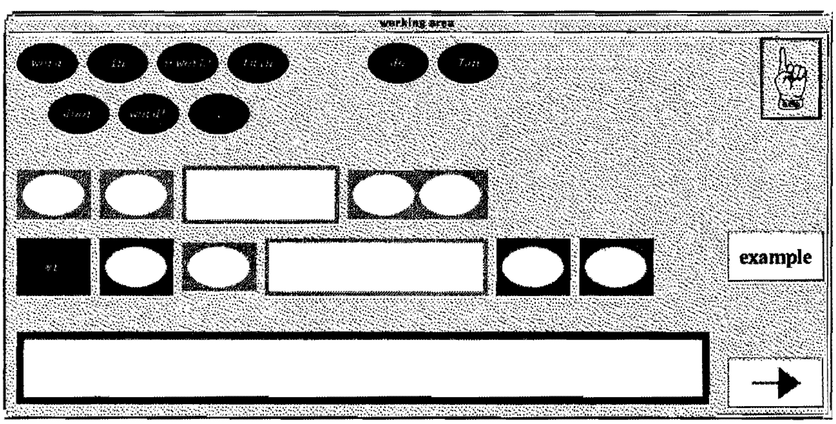


Figure 68: *Interface of the jigsaw exercise.*

(E4) An example is shown, consisting of an assignment ("Transform the following sentence into the passive form"), a small piece of video of a real life situation, a sentence which describes the situation ("Jan werkt in de tuin") both in text and audio, and a collection of words which are put into frames and consequently into bigger frames in order to construct the passive form of the sentence.
(E5) *"So, look at the sentence in the Presentation Area and transform it as required. Start with the smallest parts and put them into the frames. Finally put the largest parts from left to right into the sentence frame."*
(E6) "Transform the following sentence into the passive form".
(P1) An exercise is displayed, and the student can start constructing the sentence.
(S1) The student does not do anything. Time passes.
(E7) *"Raise your finger to get help."*

(S2) The student does not do anything. Time passes.
(E8) *"I will help you a bit"*
(E9) A word, or frame containing multiple words, is put on the right spot, thereby making the exercise easier.

In this fragment, actions E1 and E6 are typically meant to provide content information, actions E2 and E5 are meant to provide interface information, and actions E3, E7, and E8 are meant to provide meta information. It is also possible to provide different kinds of information in one action. For instance, action E4 (showing an example) provides both interface information, namely how to perform the exercise, as content information, namely a correct transformation of a certain sentence. Similarly, action E9 (putting a block on the right spot) provides both content and interface information.

When the student has already seen an example of that type of exercise, or no example is available, actions E3 and E4 are omitted. When the student shows an inability to perform the exercise, the example may be repeated. The student can also ask for the example.

When the student has already demonstrated an understanding of the user interface of the exercise, action E7 ("Raise your finger to get help.") is replaced by an incentive to act ("You must do something to proceed.").

Actions like E8 and E9 depend on the material provided by the course designer. For some exercises, these actions may not be possible. For other exercises, more levels of hints may be distinguished

In the second fragment the student is presented with an exploration exercise on clock times, in which he is supposed to study full hours, half hours and quarters of an hour. The student has to set the hands of a clock at a time, and can successively hear and see a Dutch sentence describing this time.

(S1) The student sets the hands of the clock to one o'clock and obtains as the result "het is een uur"[1].
(S2) The student sets the hands of the clock to five o'clock and obtains as the result "het is vijf uur".
(S3) The student sets the hands of the clock to eleven o'clock and obtains as the result "het is elf uur".
(E1) *"Now try something different. Try a half hour."*

In this fragment, action E1 illustrates that the Explanation Agent can take over control whenever necessary. As the student keeps on trying full hours, while he also has to learn half hours and quarters of an hour, the Explanation Agent reminds him to study the other subjects as well. So, on the one

1. This kind of feedback is handled by the domain-dependent module of Appeal.

hand, students have the opportunity to discover for themselves how clock times are expressed in Dutch, but on the other hand the learning process is still monitored.

In the Appeal prototype system, information is not only provided explicitly, but also implicitly. The students know when it is their turn to act when the artificial teacher is reading his book (see Figure 69). As a matter of fact, this is an emergent property of how the video of the artificial teacher was constructed. Because the messages of the teacher have to be adapted to the student, it was not clear beforehand what the teacher would have to say successively. Sentences and parts of sentences were recorded beforehand, and concatenated at run time. For the concatenation process to succeed the teacher needed to return to a standard position at the end of each message, which was reading the book.

Figure 69: Snapshots of the artificial teacher. In the left-hand snapshot, the student is supposed to do something, such as doing an exercise. In the middle snapshot, the teacher is talking. In the right-hand snapshot, the student is instructed to ask for help when he needs it.

7.2.3 Directions for experimental research

As yet no experiments have been performed to evaluate the effectiveness of the behaviour of the Explanation Agent. However, its behavioural repetoire turned out to be sufficiently expressive to deal with all kinds of different exercise formats, and the interaction emerging from its behaviours is felt by students to be quite natural and transparent.

In the first place, experiments should be performed on the effectivity and acceptability for the students of receiving information in the form of hints during the exercises. A condition in which hints are given spontaneously by the agent should be compared with a condition in which no hints are given, a condition in which hints can be requested by the student, and a mixed condition in which hints can both be requested by the student and be spontaneously given by the agent. The Square Dance course (see Chapter 6) can be used as a possible task domain.

In the second place, evidence should be collected to support the validity of the rules used by the agent to omit information. The domain-independence of these rules should also be explored further.

7.3 The Feedback Agent

As described in Chapter 2 (Section 2.6.2), the main task of the Feedback Agent is to give feedback adapted to the student. This includes providing the students at the right moment with information about the correctness of their responses and with the correct responses themselves.

7.3.1 Theory underlying the Feedback Agent

In Chapter 1 (Section 1.2.3), an overview was given of research indicating aspects of feedback that should be adapted to the student. This included the amount and timing of feedback, and the degree of enthusiasm and disappointment (for references see Chapter 1).

The *degree of enthusiasm and disappointment* mainly depends on the student's performance and is fairly independent of the topic. Therefore, we focused on that issue in the first design of the Feedback Agent. The adaptation of the degree of enthusiasm was based on the idea that a student who has made a mistake should be additionally motivated when answering correctly afterwards. In explorative studies aimed at optimizing the interface of the Square Dance course (described in Chapter 6), subjects indicated that they also expected the artificial teacher to become enthusiastic when they had made a number of correct responses. They showed a tendency to make mistakes on purpose, just to get the teacher enthusiastic when answering correctly afterwards. Therefore, an extra behaviour was added in which the Feedback Agent increased the degree of enthusiasm after a certain number of successive correct responses.

With respect to the *timing of feedback*, we opted for immediate feedback instead of delayed feedback. Immediate feedback seems more readily applicable in the case of domain-independence: it is hard to determine in a domain-independent way how much feedback should be delayed.

The agent chooses between single-try feedback (feedback after one response) and multiple-try feedback (feedback after several responses) depending on the student's performance. When the student gives an incorrect answer to an item that has been answered correctly before, a second opportunity is given before feedback is provided.

Like the Practice Agent and the Navigation Agent, the Feedback Agent uses a good set (see Chapter 3) to record which items have been answered correctly the last time they were presented. In a sense, the good set gives an indication of the prior knowledge of the student when doing a particular exercise. So, in this simple way the results of Clariana (1993) are used, which indicated that low prior-knowledge students benefited more from single-try feedback and high prior-knowledge students benefited more from multiple-try feedback.

If feedback is displayed visually, the agent should determine the duration of the display. The most effective display duration of the correct response –say "het is half een" in Dutch expresses "it is half past twelve"– depends very much on the topic and the specific exercise, since it depends on the complexity of the response. Adaptation could be performed on the basis of a display duration provided by the course designer, increasing it when the student is performing very poorly. Another possibility is to provide the student with an opportunity to adjust the "patience" of the Feedback Agent. The possibility for this kind of run-time parameter adjustment has been incorporated in the Appeal prototype.

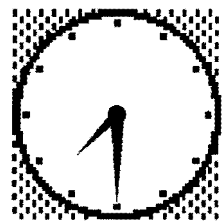
The *amount of feedback* can vary between no feedback at all, feedback indicating the correctness of the response, presentation of the correct response, and more elaborative forms of feedback with the reason why a certain answer is incorrect (Kulhavy, 1977). The amount of feedback used by the agent currently depends merely on the specific exercise involved. For instance, no feedback has to be given in explorative exercises in which the answer is always correct. The correctness of the response does not have be indicated by the agent in exercises in which the students can judge the correctness themselves, as in the jigsaw exercise discussed above where the words and frames bounce back automatically when put at a place where they do not fit. The correct answer does not have to be shown in exercises in which the students automatically know the correct answer when they see that their response was incorrect.This kind of information about the exercises is currently provided by the course designer.

The use of no feedback for a correct response for the high performing students, as suggested by Bouwhuis and Bunt (1993), is not needed. Because of the functioning of the Practice and Navigation Agents, the students are mostly confronted with problems that lie just beyond their current ability. So, it is not very likely that a situation will occur in which students have to endure a long sequence of items to which they can respond correctly. However, this argumentation does not apply when the student is training on speed of response instead of on correctness.

The reason why a certain response is incorrect can only be provided by the agent to the student when the domain-specific part of the application supplies this information. Adaptation could take place with respect to whether this kind of elaborate feedback is presented. This may, for instance, depend on the perceived confidence a student has in his answer (Kulhavy & Stock, 1989).

7.3.2 Example of the Feedback Agent's behaviour

The following fragment shows an interaction between a student and the Appeal prototype system for learning Dutch by English-speaking persons. It illustrates the behaviour of the Feedback Agent as it is currently implemented. In the fragment the student is doing an exercise on the expression of clock times in Dutch: a clock time is expressed in Dutch, and the student is supposed to set the hands of a clock to that time. P denotes the Practice Agent, S denotes the student, and F denotes the Feedback Agent.



(P1) *"Het is zes uur."* (It is six o' clock.)
(S1) The student sets the hands of the clock to six 'o clock.
(F1) *"Correct."*
(P2) *"Het is half acht."* (It is half past seven.)
(S2) The student sets the hands of the clock to half past eight.
(F2) *"That is not quite right."*
(F3) *"The correct answer is"*
(F4) The correct answer is shown.
(P3) *"Het is kwart over twee."* (It is a quarter past two.)
(S3) The student sets the hands of the clock to a quarter past two.
(F5) *"Excellent!"*
(P4) *"Het is half vijf."* (It is half past four.)
(S4) The student sets the hands of the clock to half past four.
(F6) *"That is correct."*
(P5) *"Het is tien uur."* (It is ten o' clock.)
(S5) The student sets the hands of the clock to eleven o' clock.
(F7) *"That is not correct."*
(F8) *"Try again!"*
(P6) The same exercise is presented again.
(S6) The student sets the hands of the clock to nine o' clock.
(F9) *"No, that is not correct."*
(F10) *"I will show you the correct answer."*
(F11) The correct answer is shown.

In this fragment, actions F1, F2, F5, F6, and F7 are intended to inform the student about the correctness of the response. They illustrate an adaptation of the degree of enthusiasm and disappointment to the student's performance. The degree of enthusiasm increases from F1("Correct.") to F5 ("Excellent!"), because the student has made an error (S2) just before the correct response (S3) leading to F5. The degree of enthusiasm decreases again from F5 ("Excellent!") to F6 ("That is correct."). Different degrees of disappointment, like F2 ("That is not quite right.") versus F7 ("That is not correct."), require information coming from the domain specific part of the application about the degree of severity of the error.

Actions F3, F4, F10, and F11 are intended to inform the student about the correct response. These kinds of actions are not needed when students can see the correct response immediately themselves, knowing that their own response was incorrect. This occurs, for instance, when the student has to choose between two alternatives.

Action F8 ("Try again") illustrates the distinction that the Feedback Agent tries to make between errors and slips. Because the student answered correctly the last time a full hour was presented (see S1), the agent assumes that the student made a slip when answering incorrectly on the full hour (S5), and the student gets a second opportunity to produce the correct answer.

Incidentally, this fragment also illustrates the item sequencing behaviour of the Practice Agent as discussed in Chapter 3, 4, and 5: a half hour is presented again very soon (P4) because the student had given an incorrect answer to a half hour (S2) before.

7.3.3 Directions for experimental research and extensions

No experiments have so far been performed to evaluate the effectiveness of the behaviour of the Feedback Agent. In the first place, experiments should be performed in which the effect of adaptation of the degree of enthusiasm and disappointment is tested. This adaptation is supposed to have a positive impact on the student's motivation and the student's feelings towards the artificial teacher. So, it would only influence the student's learning process indirectly. This indirect influence is unlikely to be found in an experimental setting (also due to the Hawthorn effect). However, motivation and feelings towards the artificial teacher can be measured with scales, and the number of trials a subject would like to practise can perhaps also be measured.

In the second place, experiments should be performed in which the effect of the "try again" is tested. The number of times when the student answers an item correctly after the "try again" option can be measured. When this number is relatively high, say above 50%, slips occur frequently, so, the "try again" behaviour is needed. Of course, this may depend on the instruction domain and the individual. Therefore, it may be preferable to let the agent learn this itself, so that the agent represses this behaviour whenever it is not necessary.

The behaviour of the Feedback Agent should be extended to include the capability of adapting the display duration of a correct response, as the display duration is an important factor in determining the tempo at which the student can study. The possibility of adapting the use of elaborate forms of feedback to the student should also be explored.

7.4 The Presentation Agent

As described in Chapter 2 (Section 2.6.2), the main task of the Presentation Agent is to adapt the form in which information is presented to the student. This includes, for instance, choosing whether to use formulas or pictures, and whether to use audio or text.

7.4.1 Theory underlying the Presentation Agent

In Chapter 1 (Section 1.2.3), an overview was given of research indicating that the presentation of information should be adapted to the student. Two aspects seem to be important in the adaptation of the presentation format to the student.

In the first place, the presentation format should be such that information can be represented as effectively as possible in memory. According to the dual coding theory of Paivio (1971, 1978), there are at least two mental representation forms: a verbal representation which is modality-independent and encodes linguistic information, and a non-verbal representation which is modality-dependent and encodes, for instance, pictorial information. Whereas text information is usually encoded only verbally, pictorial information is encoded both verbally and pictorially. According to the mental model theory (Johnson-Laird, 1983), which is an extension to the dual coding theory, a runnable mental model is constructed by making cross-referential connections between verbal and pictorial encodings.

Information is recalled better when it has been encoded both verbally and pictorially (Kulhavy, Lee, and Caterino, 1985). So, dual coding theory predicts that recall is best when information has been presented both verbally and pictorially (not necessarily simultaneously). This has been confirmed by empirical research (Paivio, 1986), though there is also some research that does not support this prediction (Kirby, 1993). Presenting information pictorially does not only lead to better recall, but also provides a better understanding of the subject matter (Schnotz, Picard, and Hron, 1993).

In the second place, the presentation format should be adapted to the student's learning strategy. Some students may have a higher spatial ability than others, in the sense that they have a higher ability to generate, retain, and transform abstract visual images (Lohman, 1979). This may lead to the use of different learning strategies such as an analytic strategy or a spatial strategy in a spatial visualization task (Kyllonen, Lohman, and Snow, 1984). Some students may pay more attention to text information, and others more to pictorial information.

These two aspects may conflict: the learning strategy of the student may not always be the most effective. The learning strategy can be influenced by training, but this may lead to a decrease in the performance of the able students (Kyllonen, Lohman, and Snow, 1984). Nevertheless, Kyllonen, Lohman, and Snow (1984) found, for the spatial visualization task, that even though verbal-analytic training could disrupt the performance on less difficult problems, verbal-analytic skills were required to process some more difficult problems. They suggested that a strategy shift was required for able students to meet aspects of increasing item difficulty. This suggests that though pictorial information may be very important for the initial model building, text information may be very important for abstraction.

Taking this into consideration, the Presentation Agent can use Dale's "Cone of Experience" (Dale, 1969) (see Chapter 1, Section 1.2.3) in several ways. Two strategies have been implemented. In one strategy, the agent starts with the most pictorial presentation form to support model building, and then goes progressively higher on the scale towards verbal information. In the other strategy, the agent starts with the most abstract (verbal) presentation form, and then goes progressively lower on the scale to a level the student can manage.

For both strategies, the step size can be adapted to the student, in the sense that it is larger when going in the direction of the student's preference, that it is larger when going to a higher level if the student is a high performer, and that it is larger when going to a lower level if the student has a history of low performance. The moment when the form of presentation changes depends on the student's performance. The following three general rules apply to a step to a lower level. The student should have answered incorrectly to the item last presented, the percentage of items in the good set (see Chapter 3) should be small, and the student should have had the opportunity to practise for a while with the current presentation style. The same kind of general rules apply to a step to a higher level.

Both strategies require the course designer to order the possible presentations with respect to the level of abstractness. Another possibility is that the agent has some knowledge about different types of presentations, and that the course designer should only provide the type of presentation, e.g., video or text.

Apart from using the level of abstractness, the Presentation Agent could also adapt the modality of the presentation form to students with perceptual handicaps. For instance, more use of audio could be made for students with low vision.

7.4.2 Example of the Presentation Agent's behaviour

The following gives an example of the Presentation Agent's behaviour. Suppose a child is learning addition. P denotes the Practice Agent, S denotes the student, PR denotes the Presentation Agent, and E denotes the Explanation Agent.

(P1) An exercise is presented in which two small numbers have to be added.
(PR1) The numbers are represented by a corresponding number of apples.
(S1) The student does not do anything. Time passes
(E1) *"Let me give you a hint."*
(E2) It is shown how an addition can be performed.
(PR2) The instruction is represented by the group of apples being joined together.
.... [After a number of correct responses.]
(PR3) *"Let's make it somewhat more difficult."*
(P2) An exercise is presented in which two small numbers have to be added.
(PR4) The numbers are represented by text.

In this fragment, actions PR1, PR2, and PR4 show that the Presentation Agent determines the form in which an exercise and instruction are given to the student. Action PR3 shows a step to a higher level, because of the student's good performance.

Actions E1 and E2 of the Explanation Agent may not have taken place if the presentation level had been different. The level of presentation as determined by the Presentation Agent limits the other agents in the actions they can perform. For instance, if the course designer has not provided instruction at the appropriate presentation level, no instruction can be given. The fact of whether instruction has been given is used by the Presentation Agent in determining the moment of a change (for instance to a lower level) in the form of presentation. So, if no instruction is given and the student is not doing anything, this can trigger the Presentation Agent to change the form of presentation to a lower level.

7.4.3 Directions for experimental research

Preliminary versions of the Presentation Agent have been implemented and an exploratory experiment has been performed. However, more research is needed to develop a good experimental task. An artificial task should be constructed in which learning occurs gradually, and different representation forms make sense. The degree to which different learning tasks may require different representations should also be explored.

Another issue is that the effectiveness of the Presentation Agent largely depends on the effectiveness of the Practice Agent, Navigation Agent, and Explanation Agent. In combination with the fact that course designers are not used to providing different versions of presentations at various levels of abstractness, and therefore may be somewhat reluctant to do so, this gives research on this agent a lower priority than research on the other agents.

7.5 Conclusions

A short overview has been given of the current status of the Explanation Agent, Feedback Agent, and Presentation Agent, and the connection of their behaviour with the theory. As no experiments on the behaviour of these agents have so far been performed, it is not possible to draw any conclusions regarding their effectiveness.

References

Bouwhuis, D.G., & Bunt, H. (1993). Interactive instructional systems as dialogue systems.In L. Verhoeven (Ed.) *Training for literacy.* Dordrecht: Foris.

Clariana, R.B. (1993). A review of multiple-try feedback in traditional and computer-based instruction. *Journal of computer-based instruction, 20,* 67-74.

Johnson-Laird, P.N. (1983). *Mental models: Towards a cognitive science of language, inference and consciousness.* Cambridge: Cambridge Press.

Grice, H.P. (1975). Logic and conversation. In: P. Cole and J. Morgan (Eds.) *Syntax and semantics, Volume 3: Speech acts.* New York: Academic Press.

Kirby, J.R. (1993). Collaborative and competitive effects of verbal and spatial processes. *Learning and Instruction, 3,* 201-214.

Kulhavy, R.W. (1977). Feedback in written instruction. *Review of Educational Research, 47,* 211-232.

Kulhavy, R.W., Lee, J.B., & Caterino, L.C. (1985). Conjoint retention of maps and related discourse. *Contemporary Educational Psychology, 10,* 28-37.

Kulhavy, R.W., & Stock, W.A. (1989). Feedback in written instruction: the place of response certitude. *Educational psychology review, 1,* 279-308.

Kyllonen, P.C., Lohman, D.F., & Snow, R.E. (1984). Effects of aptitudes, strategy training, and task facets on spatial task performance. *Journal of Educational Psychology, 76,* 130-145.

Lohman, D.F. (1979). *Spatial ability: A review and reanalysis of the correlational literature* (Tech. Rep. No.8). Stanford, CA: Stanford University, School of Education, Aptitude Research Project.

Paivio, A. (1971). *Imagery and verbal processes.* New York: Holt, Rinehart & Winston.

Paivio, A. (1978). A dual coding approach to perception and cognition. In H.L. Pick & E. Saltzman (Eds.), *Modes of perceiving and processing information*. Hillsdale, NJ: Erlbaum.

Paivio, A. (1986). *Mental representations: A dual coding approach.* Oxford: Oxford University Press.

Schnotz, W., Picard, E., & Hron, A. (1993). How do successful and unsuccessful learners use text and graphics? *Learning and Instruction, 3*, 181-199.

Taylor, M.M. (1988a). Layered protocols for computer-human dialogue I: principles.*International journal of man-machine studies, 28*, 175-218.

Wood, D., Wood, H., & Middleton, D. (1978). An experimental evaluation of four face to face teaching strategies. *International journal of behavioral development, 1*, 131-147.

Chapter 8

Conclusions

8.1 Introduction

The main goal of this dissertation was to determine how instruction can best be adapted to the individual student. A completely different approach has been chosen than the ITS approach: the focus has been on the instruction dialogue, hence, on domain-independent adaptation, with a mixed locus of control as an important issue. Instead of more traditional artificial intelligence, situated agents have been used to attain adaptivity. While the present research was being carried out, a prototype of an interactive instruction system, called Appeal, was developed. The knowledge acquired in this research served as a basis for the implementation of a domain-independent teaching module in the Appeal prototype. This domain-independent teaching module has successively been applied in various instruction domains.

Research presented in this dissertation has shown that several aspects of instruction can be adapted to the individual student without using domain-knowledge. Furthermore, it has been shown that it is possible to construct such a domain-independent teaching module as the interaction of a number of highly autonomous agents which, in their turn, can be designed as the interaction of a collection of relatively simple rules. Consequently, the design effort is limited compared to that needed in the case of ITS's.

The agent design process (see Chapter 2) has been illustrated as an iteration of using knowledge of the human learning process (see Chapter 3, 5, 6, and 7) and experimental research (see Chapter 4, 5, and 6). It has been shown that well-controlled experiments can be used to evaluate adaptivity.

Parts of the behaviour of the Practice Agent and Navigation Agent have been evaluated empirically. Future research should be aimed at extending these evaluations, and also at evaluating the behaviour of the other agents (see Chapter 7), and adjusting the behaviours of the agents accordingly.

8.2 Summary of the research results

8.2.1 The learning process

An important general finding from the research presented in this thesis is that the adaptive behaviour of the agents reduced the variance between students. The performance of the poor learners increased, while the performance of the good learners stayed at the same level. This was found for item sequencing in paired associates learning (Experiment 1, Chapter 4), item sequencing in concept learning (Experiment 2, Chapter 5), and navigation through the course material (Experiment 2, Chapter 6).

Another finding is that initiative of the agents (as in navigation, see Chapter 6) is particularly essential when students are unable to monitor their own learning process. Initiative of the agents also reduces the variance, especially in a situation with a mixed locus of control (see Chapter 6).

8.2.2 Guidelines

On the basis of the research presented in this thesis, guidelines can be given with respect to the kind of interactive instruction systems needed, the design of such systems, and the evaluation.

Guidelines with respect to the kind of IIS's needed

1. *Adapt to the individual students.*
Adaptivity reduces the differences in performance between students, which makes it easier to predict the effect of instruction (see Chapter 4, 5, and 6).

2. *Use a mixed locus of control.*
Control and initiative of the system is needed to monitor the learning process of the poor performers. Student control is suitable for high performers and enables the students to intervene when suboptimal choices are made by the system (see Chapter 6).

Guidelines with respect to the design of such systems

1. *Use an architecture of simple, independent agents.*
Adaptivity and a mixed locus of control can emerge from the interaction of a collection of simple agents, each of which represents a task of the teacher. These independent modules can be developed in parallel (see Chapter 2).

2. *Start with simple behaviours and extend these gradually.*
Not all functionality has to be developed simultaneously: it is better to start with very simple behaviours and extend these gradually, using a differentiation process (see Chapter 2) and the results of experiments (see Chapter 4, 5, and 6).

3. *Base behaviour on existing knowledge of the human learning process.*
In the design of initial behaviour, existing knowledge of the human learning process can be used (see Chapter 3 and 7). The main idea is not to use executable student models, but to use the functional properties of such models in combination with direct observation (see Chapter 3).

4. *Focus on domain-independent aspects.*
It is already possible to obtain a considerable degree of adaptation without using domain knowledge. Focusing on domain-independent aspects of instruction leads to the design of modules that can be reused in various learning domains (see Chapter 3, 4, 5, 6, and 7).

Guidelines with respect to the evaluation of such systems

1. *Incorporate experiments in the design process.*
Experiments can be used to evaluate an agent's behaviour and adjust it accordingly. As a result of experiments, extra behaviours can be added, or existing behaviour can be changed (see Chapter 4, 5, and 6).

2. *Look at the learning process rather than asymptotic performance.*
Asymptotic performance does not give much insight into the learning process: the way in which learning occurs is more important for measuring the effectiveness of instruction, and why it is or is not effective (see Chapter 4, 5, and 6).

3. *Look at individuals instead of averages.*
In the evaluation of adaptivity it is important to look at the results of individual subjects rather than to focus on averages. A typical result of adaptivity is a reduction of variance (see Chapter 4, 5, and 6).

4. *Use models.*

It takes a lot of effort and time to perform user experiments. Models can be used as simulations of students in order to obtain predictions of the effectivity of the behaviour of an agent, and to explore the effect of different parameter settings of the behaviour (see Chapter 3). However, in the end real-user experiments still have to be performed as the models are not always correct (see Chapter 4, and 5).

8.3 Questions and limitations

8.3.1 What is the benefit of using agents?

One might argue that the use of the term "agent" does not contribute much; that the algorithms described in, for instance, Chapter 3 and 6 do not need this term. However, it is not the use of the term we advocate, but the different mind set, the other perspective it gave on the problems we were dealing with. The approach of designing very simple behaviours with the observable behaviour of the student as the main input proved to be very useful. Problems that seemed very difficult initially, turned out to be relatively simple when viewed from that perspective. The notion of autonomy also naturally led to the idea of the agents taking initiative.

The notion of agents is also important because of the modularity it implies. It turned out to be very easy to add, remove, and change agents. Adaptivity on different aspects could be implemented and tested gradually

8.3.2 Is this the answer to the problem of introducing interactive instruction systems in schools?

The added value of interactive instruction systems for regular education is more obvious when they can adapt instruction to the individual student. However, this in itself does not solve the problem of introducing interactive instruction systems in schools.

The main problem for publishers seems to be the high costs of producing a high-quality interactive title. These costs only make it profitable to produce titles that can be used for a very large group of students, as in elementary schools, but then the profit margin is generally rather small. Of course, it can be argued that more adaptivity makes a title suitable for a larger group of students. However, the adaptive power is limited by the material provided by the course designer. So, the introduction of adaptivity certainly requires extra effort on the part of the course designer. Under these conditions, publishers are not very eager to introduce more functionality -and hence extra costs- into their titles.

The high costs of producing a title seem mainly to be due to

- the current development process, which is difficult to control
- expensive assets and complex software that are hard to reuse.

Methods should be developed to control the complexity of creating and maintaining multimedia software better. This is one of the goals of the Multimedia & Education project which is currently taking place as a cooperation between Philips Research and the Institute for Perception Research. The focus in this project is on explicit structuring of course material and modularity.

The general idea is that the gap between current approaches to interactive instruction design and the higher end ideas as presented in this thesis should be bridged gradually. The architecture should, however, be chosen such that a gradual addition of more adaptivity is possible. The agent-based architecture as presented in this thesis is very suitable for this purpose.

8.3.3 What kind of adaptivity should be introduced first?

Given that it is unrealistic to expect that publishing firms would want to add fully fledged adaptivity at once, the question arises as to what kind of adaptivity should be realized first. Two aspects seem to be important.

In the first place, the kind of adaptivity should require as little effort as possible from the course designer. This reduces the suitability of adaptivity with respect to the form of presentation, as this would involve the production of many extra assets (for different variants of the same presentation). For the same reason, adaptivity with respect to explanation is less suitable.

In the second place, the kind of adaptivity should have the greatest possible effect on the learning process of the students. This reduces the suitability of adaptivity in feedback, as the effect of domain-independent adaptation of feedback is certainly not as great as adaptivity in navigation.

Considering these two aspects, the kind of adaptivity that we would add first is adaptivity with respect to navigation. This requires explicit structuring of the course material and modularity, but these are also important for controlling the development process.

8.3.4 Is domain knowledge really not necessary?

One might argue that for really high level adaptivity it is not sufficient to use only domain-independent instruction rules. The student should, for instance, be able to ask questions about the domain. The goal of this research was, however, not to make domain knowledge completely superfluous, but to show that a lot of adaptivity can already be obtained in a domain-independent way. Indeed, for high level applications the domain-independent teaching module should be combined with a domain expert module. This has already been done in the Appeal prototype system, though in that case the domain expert module was still rather primitive.

We are currently exploring the possibility of combining the domain-independent teaching module of Appeal with the DenK system (Ahn, Beun, Borghuis, Bunt, and VanOverveld, 1994). DenK offers a cooperative assistant that has knowledge about the domain, can reason about that knowledge and can communicate with a user in natural language. In a combination of Appeal and DenK, the assistant of DenK could handle the symbolic interaction with the student (like typed natural language), and could answer various questions from the student about the domain. Appeal would offer initiative, which is currently lacking in DenK, and a higher degree of adaptivity. The integration of both approaches would lead to a system that is very suitable for special user groups.

References

Ahn, R.M.C., Beun, R.J., Borghuis, T., Bunt, H.C., & VanOverveld, C.W.A.M. (1994). The DenK-architecture: A fundamental approach to user-interfaces. Artificial Intelligence Review, 8, 431-445.

Samenvatting (Summary in Dutch)

Dit proefschrift beschrijft het ontwerp van een interactief instructiesysteem. dat zich aanpast aan de prestaties van de individuele leerling. Centraal in het proefschrift staat niet het systeem zelf, maar een manier waarop zo'n systeem ontworpen kan worden: met gebruikmaking van, enerzijds, bestaande modellen over leergedrag en, anderzijds, experimentele toetsing.

Het ontwerpen van een interactief instructiesysteem kan beschouwd worden als het ontwerpen van een artificiële leraar. Het verschil tussen een goede en een slechte leraar wordt nauwelijks bepaald door een verschil in domeinkennis, maar vooral door een verschil in het vermogen zich aan te passen aan de leerlingen, een vaardigheid die onafhankelijk is van het domein. Tot dusverre heeft het onderzoek naar interactieve instructiesystemen zich echter voornamelijk gericht op domeinafhankelijke aspecten van instructie, zoals het soort fouten dat op kan treden bij het optellen van twee getallen. De doelstelling van dit project was juist die algemene, domeinonafhankelijke aspecten van instructie te modelleren. Daartoe is onderzocht hoe een artificiële leraar geconstrueerd kan worden die zich aanpast aan de individuele leerling en bruikbaar is in verschillende instructiedomeinen.

Een benadering is gekozen waarbij de domeinonafhankelijke leraar gezien wordt als een zeer autonome agent die in staat is in te spelen op de (moeilijk voorspelbare) gedragingen van de leerling. Deze agent bestaat uit een collectie van simpelere agenten, die elk een bepaalde competentie van de leraar vertegenwoordigen. Iedere agent wordt gekenmerkt door een verzameling gedragingen die gebaseerd zijn op kennis van en experimenteel onderzoek naar menselijke leerprocessen. De agenten opereren parallel aan elkaar en vertonen een nauwe koppeling tussen hun acties en hun perceptie van de omgeving (bijv. acties van de leerling en acties van andere agenten). De interactie tussen de agenten onderling en tussen de agenten en hun omgeving resulteert in de adaptieve functionaliteit van een domeinonafhankelijke leraar.

De belangrijkste taak van de "Oefen Agent" is te bepalen welk item te presenteren aan de leerling, uit een verzameling van items die geleerd moeten worden en die allemaal te maken hebben met een en hetzelfde onderwerp. Onderzocht is hoe modellen van menselijke leerprocessen gebruikt kunnen worden voor het ontwerp van een adequate selectiestrategie voor deze agent, en een nieuwe selectiestrategie is ontwikkeld. Het effect van deze strategie is empirisch getest op twee soorten leergedrag waarnaar veel onderzoek gedaan is, namelijk het leren van zogeheten "gepaarde associaties" en het conceptleren. De resultaten van deze experimenten hebben implicaties zowel voor het gedrag van de agent als voor de modellen. Dit illustreert het gebruik van empirisch onderzoek en cognitieve modellen in het ontwerpproces. Tevens illustreert het wat de moeilijkheden zijn bij het ontwerp van adaptief gedrag van een agent en hoe zulk gedrag geëvalueerd kan worden.

De "Navigatie Agent" bepaalt, afhankelijk van de interesses, kennis en capaciteiten van de leerling, wanneer een ander onderwerp (d.w.z. een andere verzameling items voor de "Oefen Agent") aan bod komt en wat dat onderwerp dient te zijn. Een belangrijke vraag die in dit verband opkomt is, hoeveel initiatief de leraar (d.w.z. het systeem) bij het bepalen van de lesvolgorde moet nemen, respectievelijk hoeveel initiatief aan de leerling zelf moet worden gelaten. Door middel van empirisch onderzoek zijn de objectieve en subjectieve voordelen vastgesteld van een zogeheten 'mixed locus of control' situatie, waarin zowel de leraar als de leerling initiatief kan nemen met betrekking tot de keuze van het lesonderwerp.

Naast de tot hiertoe besproken agenten zijn drie andere agenten te onderscheiden die zorg dragen voor een zinnige dialoog met de leerling, en door deze waargenomen worden als een 'persoon'. Dit zijn de zogeheten "Uitleg agent", de "Feedback Agent" en de "Presentatie Agent". Afhankelijk van het leergedrag van de leerling bepaalt de "Uitleg Agent" de hoeveelheid en timing van uitleg, de "Feedback Agent" de hoeveelheid en aard van feedback (d.w.z. correctie of bevestiging van het antwoord), en de "Presentatie Agent" de manier waarop de informatie gepresenteerd wordt, bijvoorbeeld door middel van plaatjes of formules. Bij het ontwerp van deze agenten ligt de nadruk op het gebruik van bestaande kennis van het menselijke leerproces.

De belangrijkste conclusies van dit onderzoek zijn dat het mogelijk is met een verzameling simpele gedragingen adaptief en ogenschijnlijk intelligent gedrag van een domeinonafhankelijke leraar te realiseren, en dat het hierbij wezenlijk is empirisch onderzoek te incorporeren in het ontwerptraject.

Curriculum Vitae

17 januari 1970	Born in Geleen, The Netherlands
1982 - 1988	Gymnasium β (cum laude) Serviam Scholengemeenschap, Sittard
1988 - 1992	Computer science (cum laude) Major: Mathematics of programming Eindhoven University of Technology
1993 - 1997	Graduate student at the Graduate School "Perception and Technology", Institute for Perception Research (IPO), Eindhoven

Stellingen

behorende bij het proefschrift

An Agent-based Interactive Instruction System

van J. Masthoff

I

Om te voorkomen dat een groot deel van de bevolking uitgesloten wordt van een van techniek doordrenkte maatschappij, is onderzoek naar de bediening van apparaten nodig.

II

In plaats van de *gebruiker* te dwingen zich aan te passen aan complexe apparaten, dient de *ontwerper* gedwongen te worden apparaten te maken die zich aanpassen aan de gebruiker.

III

Bij complexe apparatuur met veel functionaliteit is het zinvol de gebruiker slechts geleidelijk te confronteren met de bedieningsmogelijkheden, door aanvankelijk veel initiatief van het apparaat uit te laten gaan en stapsgewijs de controle naar de gebruiker te verleggen.

IV

Als systemen initiatief nemen is het van groot belang dat gebruikers deze initiatieven kunnen negeren. Tactiele media (Keyson, 1996) bieden een uitgelezen mogelijkheid om aan deze eis te voldoen.

V

Het is mogelijk om een adaptieve, domein-onafhankelijke, artificiële leraar te maken die gebaseerd is op eenvoudige principes over hoe mensen leren (zie proefschrift).

VI

Het gebruik van hyperlinks zou voor de gebruiker afgeschermd moeten worden en vervangen door een combinatie van een ruimtelijke metafoor met op agent-technologie gebaseerde gidsen.

VII

Bij dialoogsystemen is niet het ontwerpen van een efficient en correct programma op basis van een gegeven specificatie het grootste probleem, maar veeleer het komen tot een goede specificatie. Hier zou in de informaticaopleidingen meer aandacht aan besteed moeten worden.

VIII

Voor hoogwaardige adaptiviteit in de dialoog tussen een systeem en een gebruiker is taalgeneratie noodzakelijk.

IX

Het idee van een strakke planning van onderzoeksprojecten is in tegenspraak met het principe van 'gesitueerdheid' (zie hds 2): de voortgang van interessant onderzoek is namelijk per definitie onvoorspelbaar.

X

Het belangrijkste doel van een aio-project is niet het verkrijgen van een proefschrift, maar het opleiden tot onderzoeker. Aio-projecten dienen derhalve meer als een mogelijkheid tot verbreding dan tot specialisatie gezien te worden.