

The ADAPT design model: towards instructional control of transfer

OTTO JELSMA, JEROEN J. G. VAN MERRIËNBOER
& JIM P. BIJLSTRA

Department of Education, University of Twente, 7500 AE Enschede, The Netherlands

Abstract. This paper presents a detailed description of the ADAPT (Apply Delayed Automatization for Positive Transfer) design model. ADAPT is based upon production system models of learning and provides guidelines for developing instructional systems that offer transfer of learned skills. The model suggests that transfer of training can be attributed to procedure overlap between the original training task and the transfer task, as well as to analogy between new problem solving situations and acquired cognitive schemata. More specifically, the role of schemata in transfer is thought to increase as the transfer task becomes more different from the original training task. Several instructional tactics are suggested to optimize transfer of training. *Declarative* tactics pertain to the instructional design for acquiring knowledge which is relevant to performance of the skill; such tactics include demonstrating the skill, verbal instruction, the encouragement to paraphrase particular pieces of information, the application of advance organizers and mnemonic systems, and the presentation of concrete models and examples. *Procedural* tactics refer to the instructional design for acquiring the skill, that is, to the design of practice; such tactics include the encouragement to imitate the skill, the application of variability of practice and contextual interference, and the presentation of annotated examples. The relevance of ADAPT is evaluated and implications for future research are presented.

Introduction

Transfer is a central purpose of many training programs. In essence, those training programs are based upon the assumption that knowledge and skills, which are learned to perform a particular task, will positively influence performance on other more-or-less related tasks. This effect, called “positive transfer”, can range from specific (i.e., content dependent) to general (i.e., content independent): specific transfer affects only particular knowledge and skills within a circumscribed subject matter; general transfer affects a wide range of new knowledge and skills. The concepts of specific and general transfer are highly related to Mayer and Greeno’s (1972) distinction between near and far transfer which they introduced to indicate the extent of similarity between the new setting and the original training setting.

The goal of this paper is to present a cognitive psychologically based instructional design model, called ADAPT (Apply Delayed Automation for Positive Transfer), which offers guidelines for the development of instructional systems that require positive transfer. We define an instructional design model as a coherent set of rules prescribing how instructional systems must be designed. Further,

in accordance with Van Merriënboer and Krammer (1987), we distinguish between instructional tactics, as *specific* design plans that prescribe methods to reach desired learning outcomes under given circumstances or conditions, and instructional strategies, as *general* design plans that mainly differ in their control of a learner's processing load: such strategies usually include a set of tactics.

The theoretical framework for ADAPT is formed by production system models of learning (see for an overview, Klahr, Langley and Neches, 1987). Production systems are a class of computer simulation models that can be used to describe the cognitive architecture and learning processes. As we will show, these models offer various points of contact for an explanation for transfer of training based on cognitive psychology. Within this framework, we will stress the importance of organization of declarative knowledge for positive transfer and we suggest that both *declarative* and *procedural* instructional tactics may be used to optimize that organization. This suggestion should be seen as an extension of Van Merriënboer and Krammer's (1987) original interpretations of declarative and procedural instruction.

The structure of our discourse is as follows. In the first section, we outline two historically different points of view concerning the phenomenon of transfer of training, namely the theory of *Identical Elements* and the *Gestalt Approach*, and we indicate that these different approaches are still present in contemporary conceptualizations of transfer of training. In the second section, we present the theoretical framework for the ADAPT design model. First, we outline the cognitive architecture; then we discuss the process of skill acquisition. In the third section, we present the ADAPT design model. First, we present the assumptions of ADAPT; second, we formulate ADAPT's guidelines for instructional control of transfer, and finally, we present two illustrative applications of ADAPT. One example pertains to the domain of supervisory control in the process industry; the second example concerns the domain of computer programming. In the concluding section, we offer a discussion of ADAPT and we present some suggestions for future research.

History of research on transfer

The study of transfer has a long history (see for reviews e.g., Adams, 1987; Annett and Sparrow, 1985; Ellis, 1965; Osgood, 1949; Royer, 1979). Until the end of the nineteenth century, the prevailing opinion concerning transfer was governed by the *Doctrine of Formal Discipline*. This doctrine asserted that, on analogy of the relationship between physical exercises and muscle power, the power of the mind could be increased by mental exercises such as the study of "difficult" subjects as Latin and geometry which, in turn, was expected to improve performance for all other subjects in the curriculum. At the turn of this

century, two influential lines of research came up which are still of importance to the issue of transfer of training, namely research based on the Associationist tradition, in particular Thorndike and Woodworth's (1901) theory of *Identical Elements*, and research based on the *Gestalt Approach* which had its origin in Selz's fundamental conceptions of intellectual processes (cited in Mandler and Mandler, 1964).

The Associationists consider the process of thinking simply as covert doing: thinking involves mentally trying all the likely responses until the one that will work is found. Since this form of trial and error process cannot be seen, the solution therefore appears to be achieved suddenly, as by a "flash of insight." Within this associationist tradition, the Identical Elements Theory of Thorndike and Woodworth (1901) claimed that transfer from one task to another would occur only when both tasks shared identical elements. Although it never became clear what exactly was meant by identical elements, this concept is usually interpreted to mean something like stimulus-response pairs (Orata, 1928); sometimes it is conceptualized much more broadly in that similar mental actions related to stimuli or habits are included as well (Cox, Valsiner and Ornstein, 1987). In general, it was assumed that the greater the number of identical elements, the greater the amount of transfer. Osgood (1949) synthesized the first 50 years of research on transfer of training in the twentieth century in his influential article on the "transfer surface." This construct indicated that transfer is functionally related to the similarity and difference relationships between stimuli and responses in an original and transfer task.

Unlike the Associationists who consider the concept of identical elements as the determining factor for transfer of training, the Gestaltists rely on mental structures. They consider the process of thinking as reorganizing or relating one aspect of a problem situation to another which may result in structural understanding. This involves reorganizing the elements of a problem situation in a new way so that they may solve the problem. In this process, it may be important to give hints to the problem solver because this may help to break out of old ways of organizing the situation. The new way of looking at the problem is accompanied by "insight," that is, the "magical" flash that occurs when the solution suddenly falls into place. The Gestaltists held that transfer from one task to another is achieved by arranging learning situations so that a learner can gain insight into the problem to be solved. This type of learning is thought to be permanent and to transfer to new situations.

Many of the Gestalt ideas are closely tied to a slightly different view of thinking which may be referred to as "Meaning Theory." The meaning view of thinking of which Bartlett's (1932) schema conception is a well-known example, involves finding how the present problem relates to those concepts and ideas that already exist in memory, that is, relations between external elements and

schemata. In his view: "... a schema, refers to an active organization of past experiences which must always be supposed to be operating in any well-adapted organic response" (p. 201). According to the Meaning Theory, thinking is mainly a process of figuring out which schema or set of past experiences the new problem should be related to, and then interpreting and reorganizing the new situation in accordance with the particular schema that is selected. With respect to transfer of training, it is argued that positive transfer is likely to occur when a problem situation is assimilated to useful schemata.

With the shift to more cognitively-based accounts of learning and performance, interest in transfer of training research declined during the 1960s and 1970s because most effort was invested in the study of structures and processes involved in the encoding and retrieval of information during initial task acquisition and retention (e.g., Battig, 1966, 1972; Neisser, 1967; Postman, 1971; Postman and Underwood, 1973; Sternberg, 1969). However, Cormier and Hagman (1987) indicate that during the past decade there has been a new upsurge in interest in transfer of training among researchers with a cognitive or information processing approach to human learning. Central to this approach is the assumption that learners are conceptualized as active constructors of knowledge rather than passive recipients of information; they actively seek to make sense of the environment, imposing structure and order on stimuli encountered through experience. On the one side, with respect to transfer of training, this cognitive view tackles transfer problems by carefully analyzing the stimulus and response properties of the learning events.

Recently, these stimulus-response pairs have been interpreted in terms of productions (e.g., Singley and Anderson, 1985, 1988, 1989). This is clearly in the tradition of the Associationist point of view, because it is assumed that events which share stimulus properties will be recognized by the learner as being similar and that the response learned to the first event can be generalized to the second. On the other side, the cognitive view bears on the Gestaltist point of view and in particular on the Meaning Theory by making strong assumptions about the nature of underlying memory representations. Memory is conceptualized as a highly structured storage system in which information is both stored and retrieved in a systematic manner. The critical step in transfer is the retrieval of a relevant bit of information when a particular problem is encountered. According to Royer (1979), the emphasis on memory search processes specifically makes the cognitive approach not limited to near transfer problems, but also relevant to far transfer problems. Thus, it seems that both the older Associationist and Gestaltist points of view concerning the phenomenon of transfer of training, in particular the Identical Elements Theory and the Meaning Theory, are in fact still present in current cognitive views on transfer of training.

Theoretical framework for ADAPT

Recent cognitive views on transfer of training offer various points of contact for the development of instructional design models, which provide guidelines for developing instructional systems that offer transfer of learned skills to new situations. Specifically, production system models of learning are a powerful basis for the development of such instructional design models. The ADAPT (Apply Delayed Automatization for Positive Transfer) design model is based upon this theoretical framework. In this section, we will outline the nature of production system models of learning; in particular, we will discuss the ACT* (Adaptive Control of Thought) theory of skill acquisition (Anderson, 1982, 1983, 1987).

Cognitive architecture

Production systems are a class of computer simulation models that can be used to describe the human cognitive architecture and learning processes (Klahr, Langley and Neches, 1987). When modeling human cognition, a production system consists, in its most basic form, of two memories that interact via a simple processing cycle. One memory is conceptualized as procedural memory in which a set of condition-action (IF-THEN) pairs, called productions, are represented. The other memory is referred to as working memory which contains information that is basically declarative in nature. The processing cycle consists of a three-stage recognize-act cycle. In a parallel process, the condition sides of all productions from procedural memory are matched with the contents of working memory. When the condition sides of more than one production match, conflict resolution principles apply to select a single production for application. Finally, the selected production is fired. Firing a production may result in behavior as well as in changes to the contents of working memory which implies that the processing cycle can continue indefinitely. In practice, it continues until a certain prespecified goal state, such as solving a particular problem, is achieved.

Anderson's ACT* theory (Adaptive Control of Thought; 1982, 1983, 1987) extends the general cognitive architecture of production systems by making a fundamental distinction between declarative and procedural knowledge. In addition to a working memory and a procedural memory, the cognitive architecture according to ACT* includes a declarative memory in which a set of facts are represented in an inert, propositional, network-like structure. A spreading activation mechanism, which is considered a parallel process, may affect many parts of the network simultaneously. This activated part, which in fact defines the working memory, is likely to be stored in declarative memory. Activation of declarative knowledge structures may result either from perceptual processes or from firing productions of procedural memory. Thus, productions operate on declarative knowledge that is currently active in working memory.

Skill acquisition

Skill learning in fact means learning what procedures exist for accomplishing various goals, learning how to carry them out, and learning under what circumstances to apply them (Charney and Reder, 1986). To achieve a goal, it may also be necessary that new facts, new concepts and new principles be learned (Merrill, 1983, 1987). For instance, in the case of operator training, trainees have to learn how and when they should intervene in a particular automated production process; they have to learn the characteristics of the device they are using; and they have to learn the principles of the production process they are controlling.

Learned skills can usually be performed quickly, fluently, effortlessly and relatively automatically (Hasher and Zacks, 1979; Jonides, 1981; LaBerge and Samuels, 1974; Logan, 1985, 1988; MacKay, 1982; Posner and Snyder, 1975; Schneider and Shiffrin, 1977; Shiffrin and Schneider, 1977). Indeed, skill seems to be closely related to automaticity. Yet, as Logan (1985, 1988) has argued convincingly, skill and automaticity are not the same thing. Automaticity is just a component of skill; skill is more than the sum of its automatic parts. If unskilled performance were simply less automatic than skilled performance, unskilled performers would be able to do whatever skilled performers could do, only less automatically. Clearly this is wrong. Logan argues that highly skilled performers must have a lot of declarative knowledge about their skill that allows them to make better use of their procedures. For example, Kieras (1982) has shown that skilled operators know a lot about complex engineered systems such as radar sets, process control installations, or airplanes. They know the function and purpose of the device, the controls and indicators, and the inputs, outputs and connections. They also know how to operate the device to accomplish a diversity of goals. They know procedures for fault management and maintenance, and they even may know something about the internal structure and mechanisms of the equipment. Using this knowledge enables them to operate the device adequately and efficiently.

Contrary to this seems the well-known phenomenon that skilled performers do not invariably demonstrate extensive declarative knowledge (e.g., Annett, 1985; Berry and Broadbent, 1984; Gallwey, 1975). For instance, Berry and Broadbent (1984) explored the relationship between performance on a cognitive task and the explicit or reportable knowledge associated with that performance. They showed that practice significantly improved performance, but had no effect on the ability to answer related questions. In contrast, verbal instruction significantly improved the ability to answer questions, but had no effect on task performance. Thus, it seems that we cannot always describe all that we can do, and, conversely, we cannot always do all that we can describe. Although this may be true, it then remains to be seen if the term 'expertise' is appropriate.

In agreement with Logan (1985, 1988), Olsen and Rasmussen (1989) recently argued that expertise is not sufficiently described in terms of automatized skilled performance. Whereas such performance surely is a part of expert behavior, it does not completely characterize expertise. In addition to the ability to demonstrate skilled performance, according to Olsen and Rasmussen, real expertise is characterized by the ability to advise and instruct, and to reason about and evaluate novice and expert performance in a particular field. They suggest the term 'reflective expert' for this kind of expert behavior. From this observation it is only a small step to the conclusion that both the development of automaticity and the acquisition of declarative knowledge play a significant role in gaining expertise.

Automatization

Anderson (1982, 1983, 1987) uses an adaptive production system to model the development of automaticity. His ACT* theory breaks down automatization into two major stages: a *declarative stage* and a *procedural* or automatic stage. In ACT*, automatization is conceptualized as the transition from the declarative to the automatic stage by *knowledge compilation*.

In the declarative stage, the learner receives information about a skill via instruction. Rehearsal is used to maintain the information in an active state. Subsequently, the contents of working memory may be encoded as a set of facts in declarative memory. The information provided with instruction is used by general interpretive productions to generate behavior. The generality of those productions refers to the fact that they make no specific reference to any particular knowledge domain. During the declarative stage, verbal mediation is frequently observed because the facts have to be rehearsed in working memory to keep them available for the general interpretive productions. This interpretive use of declarative knowledge is an example of what Schneider and Shiffrin (1977; Shiffrin and Schneider, 1977) characterized as "controlled processing", which has the advantage of flexibility in that a learner can be circumspect about the behavioral implications of the declarative knowledge. However, it has the disadvantage that it works slowly and may lead to serious costs in terms of processing load because interpreting declarative knowledge requires continuous retrieval of facts from declarative memory and because the individual interpretive steps are generally small.

With practice, compilative processes produce domain-specific productions that may be directly applied without the intercession of general interpretive productions. In ACT*, skills can only be acquired by doing them because the compilative processes are thought to operate on the traces of production *applications* in order to create new productions. There are two distinct types of processes involved in knowledge compilation. One is *composition*: a sequence of

productions that follow each other in solving a particular problem is collapsed into a single production that has the same effect as the sequence. This can produce a considerable speedup. The other is *proceduralization*: declarative knowledge to which general interpretive productions apply, is built into new domain-specific productions. This implies a reduction of processing load because the declarative knowledge need no longer be retrieved from declarative memory and held active in working memory. Anderson (1982, 1983, 1987) emphasizes that proceduralization does not necessarily imply the loss of the declarative representation nor the ability to use it interpretively. Further, composing smaller productions into larger productions does not eliminate the smaller productions. The original productions may remain available to apply in situations in which the composed productions cannot.

In the automatic stage, behavior may be produced by direct application of domain-specific productions without interpreting declarative knowledge. Learning may proceed further through a strengthening process which increases the strength of a production with every successful application. The direct application of domain-specific productions without interpreting declarative knowledge is an example of what Schneider and Shiffrin (1977; Shiffrin and Schneider, 1977) referred to as “automatic processing.” Automatic processing has the advantage that it works fast and with low demands on working memory. However, the fact that this type of processing can directly control behavior may be disadvantageous as well; for example, there is the ever-present danger for “action slips” (Norman, 1981) to occur if particular stimulus input triggers non-intended domain-specific productions.

Schematization

In addition to the development of automaticity, that is, the building of domain-specific productions, the acquisition of declarative knowledge that productions operate upon plays a significant role in achieving skilfulness. Declarative knowledge refers to static, inert structures that contain fact-like information. The information can be encodings of examples of instructions, encodings of properties of objects, encodings of situations, encodings of procedures, and so on. In ACT*, this fact-like declarative information is represented in a propositional network. The network also includes general knowledge which allows the system to understand the facts. The general knowledge is represented in cognitive structures that have been variously labelled schemata (Bartlett, 1932; Bobrow and Winograd, 1977; Rumelhart and Ortony, 1976; Thorndyke and Hayes-Roth, 1979), frames (Minsky, 1975), scripts (Schank and Abelson, 1977), or PUPS structures (PenUltimate Production System; Anderson, Boyle, Corbett and Lewis, 1986; Anderson and Thompson, 1987).

In accordance with the view that declarative knowledge is represented in a propositional network, a schema thus can be thought of as a cluster of general propositions attached to a node which specifies the general properties of a type of object or event or activity and leaves out any specification of details that are irrelevant to the type. For instance, a schema for process control devices may specify that they consist of a special keyboard and some kind of Visual Display Unit (VDU), that they are used to control production processes, and that it is difficult to learn how to operate them. The schema leaves out many characteristics of individual process control devices, such as the precise details on the information they present and on their keyboard lay-outs; where precisely they are located in the process control rooms; where they are manufactured; and for exactly what production processes they are used. Thus, a schema is a cognitive structure that allows particular objects, events or activities to be assigned to general categories. General knowledge of the category can then be applied to the particular case. It should be noted that, like all declarative knowledge, schemata may be compiled into domain-specific productions. The content and form of schemata are not fixed, but change as skill-level is growing. We refer to the result of processes which are responsible for that change as *schematization*. In our view, there are two distinct types of processes involved in schematization. One type involves *elaborative processes*; the other type involves *inductive processes*.

Elaborative processes are encoding processes that use schemata that already exist in memory to embellish the information that is provided with the instruction. Such elaboration of information is important because, at least in skill learning, one frequently observed characteristic of instruction is that it seldom if ever directly and exactly specifies what procedures should be applied to attain particular goals, how they should be carried out, and under what conditions they should be applied (e.g., Anderson, 1982; Charney and Reder, 1986; Kieras, 1987a, 1987b). Elaborative processes connect schemata with the new information and they infer information from the schemata that was not presented in the instruction. The result of elaboration of instructional material is an increase of interconnections in the schemata. This is thought to facilitate memory for new information because multiple retrieval routes to particular information become available (e.g., Anderson and Reder, 1979; Bransford, 1979; Mayer, 1975, 1981, 1982; Reder, 1979; Reder, Charney and Morgan, 1986). According to Van Merriënboer and Krammer (1987), the use of elaborative processes may be seen as a form of meaningful learning because new information is assimilated to knowledge (schemata) that already exists in memory. As opposed to elaborative processes that make maximal reference to previous knowledge in encoding new information, we refer to restrictive processes as encoding processes through which new information is stored with minimal reference to previous knowledge. Such restrictive processes are thought to create just a target trace; they do not create traces that are redundant with the target trace.

Inductive processes, on the other hand, either create new schemata or adjust existing schemata to make them more in tune with experience. Inductive processes may either extend or restrict the range of applicability of schemata (Carbonell, 1984, 1986). A more generalized schema is produced if a set of successful solutions in a class of related problems is available. In that case, the schema may abstract away from the details in order to allow categorization and further thought and action based on the categorization. A more specific schema is produced if a set of failed solutions in a class of related problems is available. In that case, particular preconditions are added to the schema which restrict its range of use. Thus, in accordance with recent developments in research on production systems of learning, generalizations and discriminations are conceptualized as organized declarative knowledge structures that emerge as consequences of the application of inductive processes; they are subject to strategic control (Anderson, 1986, 1987; Dulany, Carlson and Dewey, 1984; Elio and Anderson, 1984; Lewis and Anderson, 1985; Neches, 1987; Ohlsson, 1987; Proctor and Reeve, 1988). Like all declarative knowledge, the results of the inductive processes may be compiled into productions that apply to a particular class of related problems.

The ADAPT design model

Our review of production system models of learning and, in particular, our discussion of the ACT* theory of skill acquisition set up the framework for the presentation of the ADAPT design model. It should be noted that ACT* is a rule-based system in which procedures are decomposed into lower-level rules that have the form of productions. Productions may be used for computer implementation. In the following, we adopt, for the sake of generality, the term *procedure* instead of production, because we are not concerned with implementation in the first place. Likewise, we use the psychological concept *schema* for cognitive structures that contain general knowledge instead of some kind of other labeling such as frames or scripts because those structures refer specifically to knowledge representations relevant to computer implementations. In the present section, we will start with a discussion of ADAPT's basic assumptions; after that, we will formulate ADAPT's guidelines for instructional control of transfer.

Assumptions of ADAPT

Table 1 lists the eight basic assumptions of the ADAPT design model. The assumptions will be subsequently elucidated.

Skill acquisition

In the previous sections, we have made a strong case for the importance of both automatization and schematization in skill acquisition. On the one hand, automatization of procedures reduces processing load which allows the system to devote cognitive resources to problem solving activities; on the other hand, schematization of declarative knowledge results in multiple retrieval routes to particular information which allows the system to make better use of its automatized procedures.

Table 1. ADAPT's basic assumptions

1. SKILL ACQUISITION

Both automatization and schematization play a significant role in skill acquisition.

2. INSTRUCTION

Instruction facilitates either automatization or schematization.

3. DECLARATIVE INSTRUCTION

Declarative instruction facilitates the application of either restrictive or elaborative encoding processes.

4. PROCEDURAL INSTRUCTION

Procedural instruction facilitates the application of either compilative or inductive processes.

5. PROCEDURE-OVERLAP

Transfer of training is explained by procedure-overlap to the extent that compiled procedures are available and apply to the current problem.

6. ANALOGY

Transfer of training is explained by analogy to the extent that declarative knowledge from other similar problem solving situations is available and may help to solve the current problem

7. RELATIVE IMPORTANCE OF AUTOMATIZATION OR SCHEMATIZATION

During training, the relative importance of schematization increases as transfer becomes further, that is, as the procedure-overlap between the transfer task and the original training task decreases

8. TRANSFER FAILURE

Transfer failure is explained by the transference of procedures to situations in which they are either not optimal (Einstellung) or completely inappropriate (over-generalization)

Instruction

ADAPT assumes that instruction facilitates either automatization or schematization. Instructional strategies that facilitate automatization are referred to as "Accelerated Automatization Strategies" (AA-strategies); instructional strategies that facilitate schematization are referred to as "Delayed Automatization Strategies" (DA-strategies). AA-strategies and DA-strategies are assumed to have opposite effects on the shape of acquisition curves: AA-strategies are thought to steepen the shape of a typical acquisition curve and DA-strategies are assumed to flatten it (Figure 1). A typical acquisition curve relating performance to time or practice is a negatively accelerated function in which the gain in performance on a trial decreases as practice continues (Lane, 1987).

AA-strategies speed up the process of knowledge compilation, that is, they facilitate the creation of domain-specific procedures, which are required to perform the to-be-learned task. Hence, practicing under AA-strategies will quickly lead to mastery, that is, to a relatively high level of skill competency but with cognitive resources possibly still given to various aspects of the task. Likewise, if practice is continued, automaticity will be attained relatively quickly. However, AA-strategies produce poorly developed schemata because they give little opportunity to apply elaborative and inductive processes.

DA-strategies slow down the process of knowledge compilation, that is, they hinder the creation of domain-specific procedures. However, they produce well-developed schemata because they facilitate the application of elaborative and

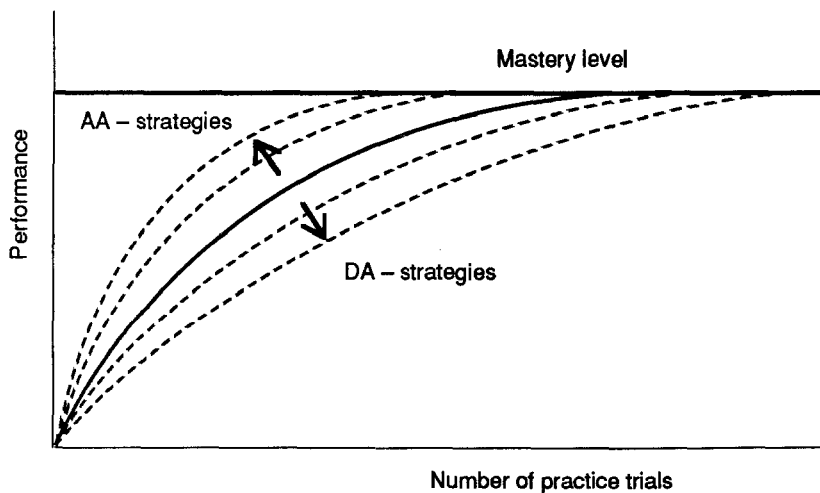


Figure 1. Predicted shiftings in the shape of a typical acquisition curve under AA-strategies and DA-strategies

inductive processes that organize and structure declarative knowledge and produce generalizations and discriminations. Schematizing is an effortful, time-consuming process which is assumed to be at the expense of the knowledge compilation process. Hence, as compared to practicing under AA-strategies, practicing under DA-strategies implies that it takes relatively long to attain a pre-specified mastery level and to develop automaticity.

Declarative instruction

Recently, Van Merriënboer and Krammer (1987) argued that within any instructional strategy for skill acquisition, two basic forms of instruction can be distinguished, namely declarative instruction and procedural instruction. Declarative instruction pertains to the instructional design for acquiring knowledge that is relevant to performance of the skill. Declarative instruction facilitates the application of restrictive processes if a learner is encouraged to maintain new information in an active state just by repeating it over and over, either out loud or mentally. In that way, only the target trace is created because no special reference is made to other related knowledge. Declarative instruction facilitates the application of elaborative processes, if a learner is encouraged not only to rehearse the new information but in addition is stimulated to relate it to previous knowledge. In that way, additional traces are created that are redundant with the target trace.

Procedural instruction

In addition to declarative instruction, procedural instruction is an essential part of any instructional strategy for skill acquisition. Procedural instruction refers to the instructional design for actually performing the skill, that is, to the design of practice. Procedural instruction facilitates the application of compilative processes if new information can be directly and easily incorporated into domain-specific procedures. Procedural instruction facilitates the application of inductive processes and hence the creation of generalizations and discriminations if immediate incorporation of new information into domain-specific procedures is prevented.

Procedure-overlap

A first approximation to an understanding of transfer of training involves comparing two sets of procedures for different tasks. ADAPT assumes that, to the extent that the procedure sets overlap, transfer will be positive from one task to the other. Recently, Singley and Anderson (1985, 1988) have argued that this formulation is in fact a modern version of Thorndike and Woodworth's (1901) Identical Elements Theory. However, whereas Thorndike and Woodworth were vague on what precisely was meant by identical elements, the identification of procedures (production rules) as the units of learning and transfer allows exact quantitative predictions for training time and transfer of training (Kieras and Polson, 1985;

Polson, 1987; Polson and Kieras, 1984). Thus, the commitment to the production system formalism offers an adequate solution to a long-standing problem of transfer of training, namely the problem of similarity of elements.

Singley and Anderson (1985, 1988) suggest that the calculation of transfer based on procedure set overlap naturally leads to a two-component model of transfer. The first or *general component* is the intersection of two sets of procedures; the second or *specific component* is the remainder of a particular set. The larger the set overlap, the larger the general component and the greater the chance that the right procedures are triggered to perform the transfer task. Thus, in pure retention tasks, which can be considered as a special case of transfer (Pieters, Jelsma and Van Merriënboer, 1987; Voss, 1987), it is argued that the general component totally overshadows the specific component and transfer of training can be expressed fairly well in terms of procedure-overlap. Thus, it needs no explaining that specifically near transfer can be well explained by procedure-overlap.

Analogy

If transfer becomes further, that is, if the transfer task increasingly deviates from the original training task, it logically follows that the general component decreases. Now, the specific component becomes too large to be ignored so that transfer of training cannot be completely attributed to procedure-overlap. To predict the magnitude of transfer, ADAPT recommends to take the declarative component in transfer of training into account. In doing so, ADAPT offers a plausible explanation for the tendency to underpredict the magnitude of transfer of training which results from Singley and Anderson's (1988) 'identical production model'. Indeed, Singley and Anderson have often been haunted by more positive transfer than they could readily predict and they suggested one additional source of transfer: "It is our view that *declarative knowledge of a special sort* [italics added] contributed to transfer ..." (p. 267). More recently, Singley and Anderson (1989) once again observed an under-estimation of transfer and they argued that the role of the declarative component in transfer of training had been ignored. They conclude that: "...declarative knowledge provides a basis for transfer between different uses of the same knowledge" (p. 220).

ADAPT assumes that transfer of training may be explained by analogy to the extent that declarative knowledge from other similar problem solving situations is available and may help to solve the current problem. In agreement with Anderson and Thompson (1987), analogy is conceptualized as the mapping process in which declarative knowledge is used to solve new problems. This knowledge can come from the learner's own past, it might come from looking at the behavior of another, or it might come from adapting an example given in a textbook or some other expository medium. In fact, analogy draws comparisons between the current

problem situation and possibly relevant information from other similar problem situations. Encoding processes may store the solution to the current problem. In compiling the results of analogy, procedures may be built that directly produce the effect of analogy without making reference to declarative knowledge. In addition, Carbonell (1984, 1986) describes how the information provided by analogy may be used by inductive processes to produce either generalizations or discriminations. If no relevant information is available, analogy fails and the problem solver has to apply other types of controlled processes to perform the transfer task.

Relative importance of automatization or schematization

ADAPT assumes that the importance of analogy increases as transfer becomes further, that is, as the general component decreases and the specific component increases. ADAPT further assumes that the success of analogy in transfer depends on the presence of possibly relevant declarative knowledge from other similar problem solving situations and in particular on how that knowledge is organized. ADAPT assumes that analogy will be more successful to the extent that the declarative knowledge is better organized. This is in agreement with Anderson (1987) who states that: "... analogy can be much more effective if [it] operates on a *rich representation of the knowledge* [italics added]." (p. 206). Declarative knowledge representation into schemata is considered to be a sound basis for analogy to operate upon because multiple retrieval routes to particular information are available which offer the possibility to derive relevant information relatively easily. Thus, the relative importance of schematization increases as the transfer task becomes more different from the original training task. Further, the relative importance of automatization is thought to increase if only the original training task has to be reproduced (retention); in that case already compiled procedures may apply in performing the transfer task and no special reference has to be made to declarative knowledge.

Transfer failure

Besides the problem of similarity of elements, another long-standing problem of transfer is the occurrence of transfer failures. Singley and Anderson's (1989) theory of the transfer of cognitive skills suggests two possible sources of transfer failures. ADAPT agrees with Singley and Anderson's theoretical point of view. One source of transfer failure pertains to the case in which a procedure that is optimal in one situation is transferred to another domain in which it is not optimal, as in the 'Einstellung phenomenon' (Luchins and Luchins, 1959). However, in terms of production-system models there is perfect transfer of productions leading to nonoptimal performance; it is not a case of the productions firing more slowly or incorrectly.

The second possible source of transfer failures pertains to the case in which two situations are apparently identical. In terms of production-system models, there is transfer of productions whose conditions match but whose actions are completely inappropriate in the transfer task. In this case, subjects fail to discriminate properly between the two situations and mistakenly apply productions which are overly general. At first glance, the situations may seem identical. However, to achieve optimal performance in either of the two situations, subjects have to apply more specific procedures.

Guidelines for instructional control of transfer

Before we present ADAPT's specific guidelines for instructional control of transfer we present some general guidelines for instruction that should be incorporated into any instructional strategy. From the wide variety of guidelines that have been formulated (cf., Gagné, 1985; Gagné and Glaser, 1987; Merrill, 1983, 1987; Reigeluth, 1983a, 1987; Romiszowski, 1981) we highlight three basic guidelines that can be directly associated with our theoretical framework.

First, new information provided with instruction should be encoded correctly. In introducing a skill to a learner, Anderson (1987) has pointed to the importance of correctness of declarative encodings. He argued that any procedure compiled from an incorrect representation of knowledge may be detrimental to later performance of the skill. Similarly, Charney and Reder (1986) argue that although declarative knowledge becomes increasingly superfluous to skill performance as learners gain expertise, the declarative representation is critical to the initial stages of skill acquisition.

Second, instruction should present a relatively small amount of new information at a time in order to prevent processing overload; in addition, this information should have to be held in working memory for a relatively short time in order to prevent working memory failures. Anderson (1987) argues that any factor that increases the amount of information that must be held, or the time for which it must be held, will increase the likelihood of working memory failures. This, in turn, may result in incorrect compilations and negatively affect performance of the skill.

Third, in a system in which skills can be learned only by doing them, the importance of formal instruction diminishes and the importance of practice increases. It should be noted, that practice *per se* does not make perfect but that only consistent practice may improve performance. That is, a learner should be enabled to make the same response to a particular stimulus whenever it occurs (e.g., Schneider, 1985; Schneider and Fisk, 1982, 1983).

Together, these three basic guidelines for instruction suggest that, as Dijkstra (1990) has recently argued, skill development and knowledge construction should take place jointly. In the initial stages of training, the instruction should not

overwhelm learners with a large amount of information that seems all relevant to later performance of the skill. In support of this view is the finding that declarative knowledge provided by verbal instruction is sometimes even unhelpful to the process of skill acquisition (e.g., Gallwey, 1975). Therefore, the necessary instruction should be compacted into its bare essentials and gradually elaborated during the skill acquisition process. Further and of utmost importance, the instruction should encourage consistent practice.

In agreement with this point of view, Carroll (1985) and, in a more recent article strikingly entitled "The Minimal Manual", Carroll, Smith-Kerker, Ford and Mazur-Rimetz (1988) showed that standard text-editor manuals can be made more effective if they are shortened and focus just on the information necessary to perform the skill. Similarly, in the context of operator training, it has become clear that the emphasis on theoretical aspects of a system's functioning, so frequently seen in introductory courses, is often disproportionate to the actual value of such knowledge. Instead, it is suggested that the content of instruction should be directly related to what the operator may be required to do in interaction with the system (e.g., Kieras, 1987a, 1987b; Mann and Hammer, 1986; Morris and Rouse, 1985; Morris, Rouse and Fath, 1985; Rasmussen, 1986).

Guidelines for instruction that pertain to the correctness of declarative encodings, to the prevention of working memory failures, and to the importance of consistent practice, are thought of as basic guidelines to be included in any instructional strategy. Following these guidelines is a prerequisite to attain mastery but it is *not sufficient to warrant transfer of training*. Specific guidelines that exert instructional control of transfer, can be easily derived from ADAPT's assumptions. These guidelines can be formulated in terms of instructional tactics, that is, in terms of specific design plans that prescribe methods to reach desired learning outcomes under given conditions.

Recently, Van Merriënboer and Krammer (1987) have formulated instructional tactics in a goals-conditions-method format. Their view on instructional tactics is very similar to Reigeluth's conception of instructional principles (Reigeluth, 1983b, 1987). Reigeluth formulated instructional principles in a circumstances-method-outcomes format. Outcomes are the effects of certain methods under particular conditions or circumstances. Outcomes may be desired or actual. Desired learning outcomes may be equalized with instructional goals (Van Merriënboer and Krammer, 1987); conditions are factors that may influence the effect of methods, such as the nature of the content or learning task, the nature of the learners, and the nature of the learning environment; methods are instructional manipulations that may achieve different goals under different conditions. The general structure of an instructional tactic comprises at least one goal, at least one condition to delimit its validity, and precisely one method.

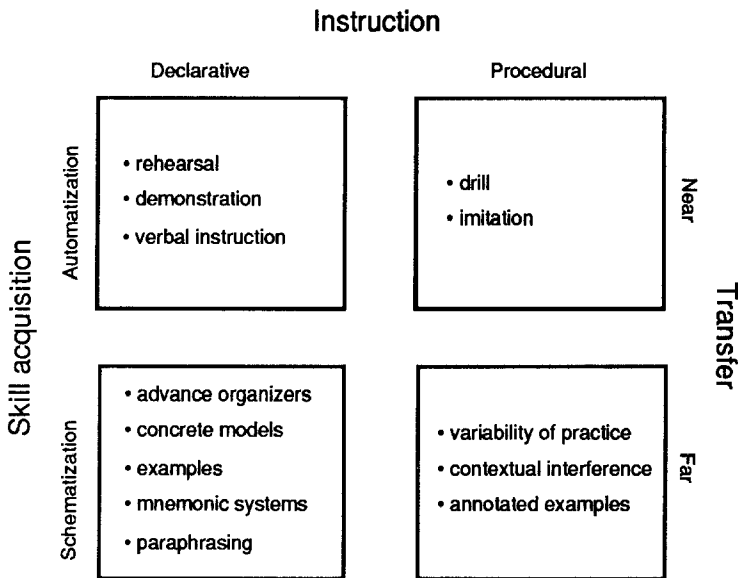


Figure 2. The ADAPT design model

In this article, goals are associated with mastery of the learning task and with transfer of training; conditions are associated with the initial skill level of the learner which, for our purposes, implies that the learners are novices. Under these restrictions, ADAPT gives rise to four categories of instructional tactics:

- A. Tactics for declarative instruction that aim at near transfer
- B. Tactics for declarative instruction that aim at far transfer
- C. Tactics for procedural instruction that aim at near transfer
- D. Tactics for procedural instruction that aim at far transfer

Tactics of category A and category C may be included in instructional strategies which we previously called “AA-strategies”; those strategies are thought to facilitate large gains in performance in the initial stages of training and they may quickly lead to mastery. However, applying tactics of AA-strategies may at best result in near transfer. Tactics of category B and category D may be included in instructional strategies which we previously called “DA-strategies”. As compared to AA-strategies, DA-strategies are thought to produce small gains in performance in the initial stages of training and hence, they attain the same performance level more slowly. However, applying tactics of DA-strategies may produce far transfer. Thus, here we are concerned with what Cormier and Hagman (1987)

characterized as one of the most important issues of contemporary research on transfer of training, namely the issue of how training for transfer differs from training for rapid acquisition. It hardly needs explaining that this issue reinforces the need in instructional design to distinguish between training and post-training performance as indices of training effectiveness.

At this point, it should be noted that a combination of AA-strategies and DA-strategies may, and often should, be used to optimize training programs. For instance, a high performance skill, which may be defined as one which requires over 100 hours of training (e.g., Schneider, 1985), usually includes several subskills that remain consistent over various problem situations. For those subskills, rapid acquisition *is*, but transfer of training is *not*, explicitly required. For instance, in the context of operator training, keyboard operation is a prerequisite for diagnosing system failures. However, this subskill does not require transfer of training, because the operator uses the same keyboard for diagnosing a variety of system failures. Thus, keyboard operation may be practiced most effectively under AA-strategies, while diagnosing system failures may be practiced most effectively under DA-strategies. In general, decomposition of a complex task in subtasks is necessary before deciding on the strategies for training those subtasks; then, an in-depth analysis of subtasks should be used to decide on the tactics that will be used.

Figure 2 presents the basic components of the ADAPT design model. In the following, we elucidate for each of the four categories the presented, characteristic examples of tactics. The list of instructional tactics should not be considered to be complete. In carefully evaluating the effects on transfer of training of other tactics, which are not presented in the present article, this list can and should be extended in order to facilitate the design of instructional systems in which transfer of training is kept effectively under control.

Declarative tactics that aim at near transfer

Probably the most effective instructional tactic to maintain a particular piece of information in an active state, so that it may be stored in memory, is simply to have a learner rehearse that information over and over, either out loud or mentally. This mere repetition of information without any attempt at understanding is sometimes associated with rote learning (e.g., Ausubel, Novak and Hanesian, 1978). There is substantial research evidence that rehearsal is an efficient way to maintain a relative small amount of information in an active state, allowing excellent short-term memory performance (see for an overview, Naveh-Benjamin and Jonides, 1984). However, because during acquisition no particular reference is made to related knowledge the information may be encoded in an isolated nature, that is, it is not adequately embedded into existing schemata. Consequently, as Annett and Sparrow (1985) have argued, the acquired knowledge is only effective

for a small range of conditions, that is, at most near transfer may be attained. Declarative tactics that encourage learners to rehearse a particular piece of information over and over, may be associated with the application of what we previously called restrictive encoding processes.

Declarative tactics that aim at far transfer

What learners already know about a topic may help them to structure and understand new information about that topic. A wide variety of instructional tactics have been put forth that encourage a learner to relate new information to existing knowledge (cf., Annett and Sparrow, 1985; Ausubel, Novak and Hanesian, 1978; Brooks and Dansereau, 1987; Clark and Voogel, 1985; Mayer, 1975, 1981, 1982; Merrill, 1983, 1987; Reigeluth, 1983a, 1983b, 1987). Well-known tactics, that have resulted in beneficial effects on transfer of training, are the application of advance organizers and mnemonic systems, the presentation of concrete models and examples, and the encouragement to paraphrase particular pieces of information. As opposed to tactics that foster rote learning, such tactics may promote "meaningful learning". In meaningful learning new information is assimilated to schemata that already exist in memory. Hence, the newly acquired knowledge makes sense in a variety of contexts, that is, far transfer may be attained (see the section on schematization for a more elaborate discussion). Declarative tactics that encourage learners to engage in meaningful learning, may be associated with the application of what we previously called elaborative encoding processes.

Procedural tactics that aim at near transfer

Once learners have acquired knowledge which is relevant to performance of a skill, they may become adept at any one application of the knowledge through practice. Domain-specific procedures, allowing automatized task performance, may be build after extended consistent practice (e.g., Schneider and Fisk, 1983). In analogy with declarative tactics that encourage learners to rehearse a particular piece of information over and over, procedural tactics that invite learners to mechanically and consistently repeat performance on a particular task are sometimes associated with drill and practice. This term originates from physical education and is still used (especially in military circles) for repetitive practice of some physical or mental operation that has to become automatic (e.g., rifle-drill or droning the multiplication tables). Although automatization of task performance has the obvious benefit of reducing processing load, it has the disadvantage of stimulus specificity, that is, task performance may be disrupted if the task or the conditions under which the task was practiced are changed to a certain extent. Thus, procedural tactics that merely make learners consistently and mechanically repeat performance on a particular task may at best produce near transfer. Such tactics may be associated with the application of what we previously called compilative processes.

Procedural tactics that aim at far transfer

Consistent practice results not only in the acquisition of domain-specific procedures, but also in the modification of declarative knowledge (Proctor and Reeve, 1988). In transfer tasks, these declarative representations are again, at least to a certain degree, required to generate behavior. Procedural tactics that encourage learners only to mechanically repeat performance on a particular task, are expected to have little effect on the declarative representations and hence they produce only near transfer. However, procedural tactics that modify declarative representations into well-organized schemata are expected to produce far transfer.

One of the best-known procedural tactics that encourages learners to develop such schemata and that has almost consistently resulted in beneficial effects on transfer of training, is the application of variability of practice (see for examples, Cormier and Hagman, 1987). Closely related to this tactic is the application of contextual interference. Whereas variability of practice is generally considered as the determinant factor for transfer of training, the results of studies on contextual interference suggest that it is not the variability *per se*, but the way it is structured across acquisition trials which determines the extent of transfer of training (cf., Lee and Magill, 1985; Shea and Zimny, 1983; Van Rossum, 1987). Another powerful procedural tactic that may produce far transfer is the use of annotated examples that are directly available during practice. Anderson *et al.* (1986) suggest that schema induction is facilitated if learners are explicitly told what the critical features in these examples are, that is, when the examples are annotated with information about what they are supposed to illustrate. Summarizing, procedural tactics like the application of variability of practice and contextual interference, and the presentation of annotated examples, may be associated with what we previously called inductive processes.

Illustrative applications of ADAPT

At first glance, ADAPT may seem to be less a model than an organizing framework. However, after closer inspection it may certainly yield renewing contributions to the conduct of instruction. Traditionally, transfer-increasing measures are primarily taken during the presentation of information to the learners; some illustrative examples have been provided in the section entitled *Declarative tactics that aim at far transfer*. Whereas practice is considered to be important in most instructional design models, its design is still often the closing entry of an instructional program. Basically, existing instructional design models continue this tradition (for an overview, see Reigeluth, 1983, 1987; but also see Merrill, 1987).

In contrast, ADAPT considers practice as the central issue in any instructional program for skill acquisition. In particular, ADAPT recommends taking transfer-

increasing measures primarily during procedural instruction and not only during the presentation of information. Hence, ADAPT is especially concerned with the design of practice. In the following, we present two illustrative applications of ADAPT. The first example pertains to the domain of supervisory control and fault management in process industry; the second example is concerned with the domain of introductory computer programming.

Supervisory control

The training of process operators for supervisory control and, in particular, fault management tasks is a complicated area, because the operators must be trained to handle a countless number of combinations of faults that can occur in the process to be controlled. In traditional operator training, formal instruction heavily emphasizes the *theory* of system functioning and of elementary physics and chemistry; Duncan (1981) refers to this as the “educational” approach. Actual skills training is usually provided on the job, with the drawback that most faults seldom occur so that their management cannot be extensively trained.

Within the framework of ADAPT, an alternative form of instruction was designed, which is characterized by a radical change towards “procedure-based” training. Such training has repeatedly been found to be more effective than the educational approach (e.g., Morris and Rouse, 1985; Shepherd, 1986; Shepherd, Marshall, Turner and Duncan, 1977). Particular attention was given to the transfer of training, because the learned fault management skills definitely have to transfer to new situations: given the theoretically infinite number of possible faults, the operators must be able to handle faults, or combinations of faults, that they have not encountered before. As a first requirement, practice for fault management skills has to facilitate automatization of those subskills that are consistent over problem situations, such as keyboard and display operation. To provide for this need in chemical industry, the high-fidelity simulator PROCESS was developed (Program for Research on Operator Control in an Experimental Simulated Setting; Jelsma and Bijlstra, in press), enabling learners to extensively practice these skills, without disturbing the production process.

As a second requirement, the training had to facilitate schematization by inductive processing for those subskills that vary greatly over situations, such as detection of different combinations of faults, diagnosing the causes of those faults, and compensating the faults and re-stabilizing the production process. To be able to center the training around the procedures that had to be followed in case of particular faults, the operator task was carefully analyzed. The task analysis resulted in a large set of procedures (i.e., productions) for handling faults or combinations of faults. Then, an exemplary sample of faults and their related procedures was selected. Using PROCESS, it was possible to predefine both the types of faults and the time at which they should occur in the simulated process;

the selected faults were introduced in the simulation and their related procedures were extensively practiced (several days) under conditions of high *contextual interference*.

In a first experiment, such training under high contextual interference proved to lead to better transfer performance than conventional part-task training techniques, that is, to better management of faults not previously encountered (Bijlstra, Jelsma and Van Merriënboer, 1990a; Jelsma and Bijlstra, 1988). In addition, when the procedural overlap between the learned tasks and the transfer task decreased (i.e., fewer of the learned procedures are applicable to the performance of the transfer task), the superiority of the high contextual interference group over the low contextual interference groups increased; this finding is in agreement with the predictions of ADAPT and supports the view that training under high contextual interference facilitates schematization by inductive processing, and so improves transfer performance.

With regard to the declarative instruction, two paths were taken in a second experiment. First, all procedures that had to be practiced were presented according to the "minimal manual" principle. A decision support system was designed that described the procedures step by step; this help system was directly available during practice and could be consulted by the learners in case they did not know how to continue. The decision support system was supplemented with interactive video-instruction which actually demonstrated the performance of the procedural steps.

Second, measures were taken to promote elaboration of the presented procedures by the application of declarative tactics that aim at far transfer. In particular, a concrete model of the process was presented prior to the training, and a general approach to fault management was discussed (i.e., the detection, diagnosis and compensation phases); this information was also available during practice. The audio-visual presentation of this information led to improved transfer performance, and when the procedural overlap between the learned tasks and the transfer task decreased, the superiority of the elaboration group over the non-elaboration group increased (Bijlstra, Jelsma and Van Merriënboer, 1990b). This finding is in agreement with the results of the first experiment and yields support to the view that the effects of inductive and elaborative processing on transfer are similar. To conclude, the systematic application of tactics from all four categories as defined by ADAPT proved to be very useful to the design of instructional strategies for operator training.

Introductory computer programming

In most conventional introductory programming courses, practice mainly consists of the generation of increasingly complex computer programs. In these traditional courses, learning outcomes and both near and far transfer performance (i.e.,

respectively, the ability to solve *new* programming problems and the ability to apply learned skills in new domains) are usually very low (for an overview, see Linn, 1985; Pea and Kurland, 1984). Within the framework of ADAPT, an alternative form of practice was designed to increase transfer performance. As a first requirement, this form of practice had to facilitate automatization (compilative processing) of those subskills involved in programming that are consistent over problem situations, such as the use of the keyboard, the editor and the interpreter or compiler, the selection of basic language commands, the application of the syntactic rules of the language, and so on. As a second, even more important requirement, the new form of practice had to facilitate schematization (inductive processing) for those subskills that heavily vary over situations, such as the decomposition of the programming problem and the subsequent composition of the program by putting programming language templates, that is, stereotyped patterns of programming code, together.

The resulting training strategy has been referred to as the “completion strategy” (Van Merriënboer and Krammer, 1990; Van Merriënboer and Paas, 1989). In this strategy, the students start with the running and hand-tracing of existing computer programs, then complete increasingly larger parts of well-structured, well-readable, but incomplete computer programs, and finally – after a relatively large number of lessons – generate programs on their own. Thus, instead of the generation of increasingly complex computer programs, the completion of increasingly larger parts of incomplete computer programs forms the kernel of the strategy.

The completion strategy provides extensive training of subskills that are consistent over problem situations, so that these basic skills may be readily automated. But moreover, it provides examples in the form of incomplete computer programs that may be annotated with information on what they are supposed to illustrate. These examples are directly available during practice and must be studied carefully in order to be able to correctly finish them; thus, they are expected to facilitate schematization. In several experiments (e.g., Van Merriënboer, in press; Van Merriënboer and De Croock, 1989), the completion strategy yielded better transfer performance than traditional training strategies for the construction of new computer programs. In particular, the students who trained according to the completion strategy showed a superior use of programming language templates; as learned templates should be seen as a special kind of schemata, it seems highly plausible that the completion strategy indeed facilitated schematization by inductive processing and so improved transfer performance.

So far, no mention is made of the presentation of information that is relevant to the performance of the programming skill (declarative instruction) within the completion strategy. In general, the “minimal manual” principle (Carroll *et al.*, 1988) can be applied to present information on the use of the programming environment and the programming language. This approach includes the most

important declarative tactics that aim at near transfer. Moreover, it is easily embedded in the completion strategy because all presented information can be illustrated by the programs that have to be completed, and are thus directly coupled to practice.

Finally, measures were taken to promote elaboration of the presented materials by the application of declarative tactics that aim at far transfer. For instance, three important tactics pertain to the presentation of concrete computer models (How does a computer work?), general design schemata (How do I write a program?), and high-level language templates (What does a program look like?). To summarize this section, the ADAPT design model proved to be very useful to the conduct of instructional design. And whereas the model is still formulated at a rather high level of detail, its fundamental concepts may lead to innovative instructional strategies.

Discussion and research implications

In this article, we presented the ADAPT (Apply Delayed Automatization for Positive Transfer) design model. It was shown that ADAPT provides several guidelines for the development of instructional systems that offer positive transfer of training. In accordance with Van Merriënboer and Krammer (1987), a distinction was made between instructional strategies and instructional tactics. Instructional strategies were conceptualized as general design plans that mainly differ in their control of learners' processing load. Instructional tactics were described as specific design plans that prescribe methods to reach desired learning outcomes under given conditions. Further, it was argued that instructional strategies typically include sets of instructional tactics; it was explained that those sets consist of at least two tactics.

Based on recent developments in production system models of learning, and in particular the ACT* theory of skill acquisition, instruction was further classified as either declarative or procedural. Declarative instruction pertains to the instructional design for acquiring knowledge that is relevant to performance of a skill; procedural instruction refers to the instructional design for actually performing the skill, that is, to the design of practice. Following Van Merriënboer and Krammer (1987), and also in accordance with Dijkstra (1990), it was argued that any instructional strategy for skill acquisition comprises both declarative instruction and procedural instruction.

It was explained that, in fact, two fundamental instructional strategies can be derived from ADAPT, namely instructional strategies that facilitate automatization (AA-strategies) and instructional strategies that facilitate schematization (DA-strategies). Application of the former strategies will quickly lead to mastery

level but produces at best near transfer; application of the latter strategies will slowly lead to mastery level but produces the opportunity for further transfer. For a complex task, the optimal strategy should be selected for each of its constituting subtasks.

Given that any instructional strategy includes at least a set of two instructional tactics and given that any instructional strategy for skill acquisition always involves declarative and procedural instruction, it was further explained that ADAPT gives rise to four categories of instructional tactics, namely declarative tactics for near transfer (e.g., rehearsal), procedural tactics for near transfer (e.g., drill), declarative tactics for far transfer (e.g., advance organizers, concrete models, examples, mnemonic systems, paraphrasing), and procedural tactics for far transfer (e.g., variability of practice, contextual interference, annotated examples). The optimal tactics to be incorporated in a strategy will depend upon the nature of the task or subtasks to be learned.

The presented theoretical framework leads to several implications for future research. First, in discussing the ADAPT design model it was pointed out that the presented list of instructional tactics is far from complete. Much research is needed to extend this list with other tactics that have an effect on the balance between automatization and schematization, in order to enable the design of instructional systems in which transfer of training is kept effectively under control. In this respect, it should be noted that, until now, the scope of the ADAPT design model is rather limited: the four categories of instructional tactics are based on the assumptions that learners are novices and that the goals of instruction are associated with the attainment of mastery of the learning task and with transfer of training. If the conditions or circumstances change (for instance, learners are advanced instead of novices), or if the goals of instruction change (for instance, attainment of automaticity instead of mastery), it is likely that additional categories of instructional tactics can be distinguished. For example, Schneider (1985) suggested a number of tactics (e.g., training under moderate speed stress or training time-sharing tasks) which should encourage learners to develop more efficient automatic procedures. Application of such tactics could be specifically suitable to attain automaticity. It is clear that more research is needed to extend the ADAPT design model into this direction in order to enlarge its scope.

With regard to a second implication for future research, it should be noted that besides training parameters, other parameters, such as subject parameters and task parameters, may have an effect on the balance between automatization and schematization (see Pieters, Jelsma and Van Merriënboer, 1987). For instance, a subject parameter like reflection-impulsivity (Kagan, Rosman, Day, Albert and Phillips, 1964) may be an important factor in schematization because it is a construct that is probably related to a person's tendency to hold on to the interpretive use of declarative knowledge during training (see also, Jelsma and Pieters, 1989a;

Jelsma and Van Merriënboer, 1989). Similarly, a task parameter like task difficulty may be related to a shift into the direction of schematization as well (see also, Jelsma and Pieters, 1989b). More research clearly is needed to reveal the effects of subject parameters and task parameters on the extent of transfer of training.

As a third and final implication for future research, it should aim at an extension of the ADAPT design model to include instructional tactics that facilitate meta-cognitive processes such as, for instance, “learning to learn”. As several authors have argued, the study of such processes may prove to be important to the development of models on transfer (see Perkins and Salomon, 1989). As the tactics that were presented in the categories ‘automatization’ and ‘schematization’ are directly related to an analysis of the task to be learned in “primary” cognitive processes, it might be fruitful to add a third category that contains both declarative and procedural tactics for the facilitation of certain *meta*-cognitive processes that are also important to reach transfer.

To conclude, the ADAPT design model as presented should be seen as an attempt to put both older and more recent knowledge on transfer of training in the perspective of recent developments in cognitive psychology. It is realized that the model is still in its infancy. However, the basic idea of the model that training for transfer differs from training for rapid acquisition may prove to be of relevance for the development of instructional systems, in which transfer of training is indeed kept under control. In the future, the importance of such instructional systems may be expected to increase because new jobs that arise as a consequence of applications of modern information technology, such as supervisory control, programming, business administration, and computer aided design and manufacturing, usually do not involve routine work but instead require flexible problem solving activities.

Acknowledgements

We thank Sanne Dijkstra and Jules M. Pieters for their useful comments on an earlier version of this article.

References

- Adams, J. A. (1987). Historical review and appraisal of research on the learning, retention and transfer of human motor skills. *Psychological Bulletin*, *101*, 417-4.
- Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review*, *89*, 369-406.
- Anderson, J. R. (1983). *The architecture of cognition*. Cambridge: Harvard University Press.
- Anderson, J. R. (1986). Knowledge compilation: the general learning mechanism. In R. S. Michalski and J. G. Carbonell (Eds.), *Machine learning: an artificial intelligence approach*. Vol. 2 (pp. 289-310). Los Altos, CA.: Morgan Kaufmann Publishers.
- Anderson, J. R. (1987). Skill acquisition: compilation of weak-method problem solutions. *Psychological Review*, *94*, 192-210.

- Anderson, J. R., Boyle, C. F., Corbett, A. and Lewis, M. (1986). Cognitive modelling and intelligent tutoring. (Technical Report No. ONR-86/1). Pittsburgh, PA.: Carnegie-Mellon University, Department of Psychology.
- Anderson, J. R. and Reder, L. M. (1979). An elaborative processing explanation of depth of processing. In L. S. Cermak and F. I. M. Craik (Eds.), *Levels of processing in human memory* (pp. 385-403). Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Anderson, J. R. and Thompson, R. (1987). Use of analogy in a production system architecture (Technical Report). Pittsburgh, PA.: Carnegie Mellon University, Department of Psychology.
- Annett, J. (1985). Motor learning: a review. In H. Heuer, U. Kleinbeck and K. H. Schmidt (Eds.), *Motor behavior: programming, control and acquisition* (pp. 189-212). New York: Springer Verlag.
- Annett, J. and Sparrow, J. (1985). Transfer of training: a review of research and practical implications. *Programmed Learning and Educational Technology*, 22, 116-124.
- Ausubel, D. P., Novak, J. D. and Hanesian, H. (1978). *Educational psychology: a cognitive view* (2nd edition). New York: Holt, Rinehart and Winston.
- Bartlett, F. C. (1932). *Remembering*. Cambridge, England: Cambridge University Press.
- Battig, W. F. (1966). Facilitation and interference. In E. A. Bilodeau (Ed.), *Acquisition of skill* (pp. 215-244). New York: Academic Press.
- Battig, W. F. (1972). Intratask interference as a source of facilitation in transfer and retention. In R. F. Thompson and J. F. Voss (Eds.), *Topics in learning and performance* (pp. 131-159). New York: Academic Press.
- Berry, D. C. and Broadbent, D. E. (1984). On the relationship between task performance and associated verbalizable knowledge. *The Quarterly Journal of Experimental Psychology*, 36A, 209-231.
- Bijlstra, J. P., Jelsma, O. and Van Merriënboer, J. J. G. (1990a). *Basic requirements for fault management training programs*. Manuscript submitted for publication.
- Bijlstra, J. P., Jelsma, O. and Van Merriënboer, J. J. G. (1990b). *Effects of audio-visual information presentation on transfer of fault management skills*. Manuscript submitted for publication.
- Bobrow, D. G. and Winograd, T. (1977). An overview of KRL, a knowledge representation language. *Cognitive Science*, 1, 3-46.
- Bransford, J. D. (1979). *Human cognition: learning, understanding and remembering*. Belmont, CA.: Wadsworth.
- Brooks, L. W. and Dansereau, D. F. (1987). Transfer of information: an instructional perspective. In S. M. Cormier and J. D. Hagman (Eds.), *Transfer of learning: contemporary research and applications* (pp. 121-150). San Diego, CA.: Academic Press.
- Carbonell, J. G. (1984). Learning by analogy: formulating and generalizing plans from past experience. In R. S. Michalsky, J. G. Carbonell and T. M. Mitchell (Eds.), *Machine learning: an artificial intelligence approach* (pp. 137-161). Berlin: Springer-Verlag.
- Carbonell, J. G. (1986). Derivational analogy: a theory of reconstructive problem solving and expertise acquisition. In R. S. Michalsky, J. G. Carbonell and T. M. Mitchell (Eds.), *Machine learning: an artificial intelligence approach*, (Vol. 2, pp. 371-392). Los Altos, CA.: Morgan Kaufman Publishers.
- Carroll, J. M. (1985). Designing minimalist training materials (Research Report). Yorktown Heights, NY.: IBM Watson Research Center.
- Carroll, J. M., Smith-Kerker, P. L., Ford, J. R. and Mazur-Rimetz, S. A. (1988). The minimal manual. *Human-Computer Interaction*, 3, 123-153.
- Chamey, D. H. and Reder, L. M. (1986). Initial skill learning: an analysis of how elaborations facilitate the three components (Technical Report). Pittsburgh, PA.: Department of Psychology, Carnegie-Mellon University.
- Clark, R. E. and Voogel, A. (1985). Transfer of training principles for instructional design. *Educational Communication and Technology Journal*, 33, 113-123.
- Cormier, S. M. and Hagman, J. D. (Eds.) (1987). *Transfer of learning: contemporary research and applications*. San Diego, CA.: Academic Press.

- Cox, B. D., Valsiner, J. and Ornstein, P. A. (1987, April). *Children's generalization of strategies: an historical perspective on transfer*. Paper presented at the meeting of the Society for Research in Child Development, Baltimore.
- Dijkstra, S. (1990). The description of knowledge and skills for the purpose of instruction. In S. Dijkstra, B. H. A. M. van Hout Wolters and P. C. van der Sijde (Eds.), *Research on instruction*. Englewood Cliffs, NJ.: Educational Technology.
- Dulany, D. E., Carlson, R. A. and Dewey, G. I. (1984). A case of syntactical learning and judgement: how conscious and how abstract? *Journal of Experimental Psychology: General*, 113, 541–555.
- Duncan, K. D. (1981). Training for fault diagnosis in industrial process plants. In J. Rasmussen and W. B. Rouse (Eds.), *Human detection and diagnosis of system failures* (pp. 553–573). New York: Plenum Press.
- Elio, R. and Anderson, J. R. (1984). The effects of information order and learning mode on schema abstraction. *Memory and Cognition*, 12, 20–30.
- Ellis, H. C. (1965). *The transfer of learning*. New York: MacMillan.
- Gagné, R. M. (1985). *The conditions of learning* (4th edition). New York: Holt, Rinehart and Winston.
- Gagné, R. M., and Glaser, R. (1987). Foundations in learning research. In R. M. Gagné (Ed.), *Instructional technology: foundations* (pp. 49–83). Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Gallwey, J. (1975). *The inner game of tennis*. London: Jonathan Cape.
- Hasher, L. and Zacks, R. T. (1979). Automatic and effortful processes in memory. *Journal of Experimental Psychology: General*, 108, 356–388.
- Jelsma, O. and Bijlstra, J. P. (1988). Training for transfer in learning to detect, diagnose and compensate system failures. *Proceedings of the Seventh European Annual Conference on Human Decision Making and Manual Control* (pp. 256–262). Paris, France.
- Jelsma, O. and Bijlstra, J. P. (in press). PROCESS: Program for Research on Operator Control in an Experimental Simulated Setting. *IEEE Transactions on Systems, Man, and Cybernetics*.
- Jelsma, O. and Pieters, J. M. (1989a). Practice schedule and cognitive style interaction in learning a maze task. *Applied Cognitive Psychology*, 3, 73–83.
- Jelsma, O. and Pieters, J. M. (1989b). Instructional strategy effects on the retention and transfer of procedures of different difficulty level. *Acta Psychologica*, 70, 219–234.
- Jelsma, O. and Van Merriënboer, J. J. G. (1989). Contextual interference: interactions with reflection-impulsivity. *Perceptual Motor Skills*, 68, 1055–1064.
- Jonides, J. (1981). Voluntary versus automatic control over the mind's eye movement. In J. Long and A. Baddeley (Eds.), *Attention and performance* (Vol. 9, pp. 187–203). Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Kagan, J., Rosman, B., Day, D., Albert, J. and Phillips, W. (1964). Information processing in the child: significance of analytic and reflective attitudes. *Psychological Monographs*, 78, (1, Whole No. 578).
- Kieras, D. E. (1982). What people know about electronic devices: a descriptive study (Technical Report No. 12), UARZ/DP/TR-82/ONR-12). Phoenix, AZ: Department of Psychology, University of Arizona.
- Kieras, D. E. (1987a). The role of cognitive simulation models in the development of advanced training and testing systems (Technical Report No. 23 TR-87/ONR-23). Ann Arbor, MI.: University of Michigan.
- Kieras, D. E. (1987b). What mental model should be taught: choosing instructional content for complex engineered systems (Technical Report No. 24 TR-87/ONR-24). Ann Arbor, MI.: University of Michigan.
- Kieras, D. E. and Polson, P. G. (1985). An approach to the formal analysis of user complexity. *International Journal of Man-Machine Studies*, 22, 365–394.
- Klahr, D., Langley, P. and Neches, R. (Eds.) (1987). *Production system models of learning and development*. Cambridge, MA: The MIT Press.
- LaBerge, D. and Samuels, S. J. (1974). Toward a theory of automatic information processing in reading. *Cognitive Psychology*, 6, 293–323.
- Lane, N. E. (1987). *Skill acquisition rates and patterns: issues and training implications*. New York: Springer-Verlag.

- Lee, T. D. and Magill, R. A. (1985). Can forgetting facilitate skill acquisition? In D. Goodman, R. B. Wilberg and I. M. Franks (Eds.), *Differing perspectives in motor learning, memory and control* (pp. 3–22). Amsterdam: Elsevier Science Publishers.
- Lewis, M. W. and Anderson, J. R. (1985). Discrimination of operator schemata in problem solving: learning from examples. *Cognitive Psychology*, 17, 26–65.
- Linn, M. C. (1985). The cognitive consequences of programming instruction in classrooms. *Educational Researcher*, 14(5), 14–29.
- Logan, G. D. (1985). Skill and automaticity: relations and future directions. *Canadian Journal of Psychology*, 39, 367–386.
- Logan, G. D. (1988). Toward an instance theory of automatization. *Psychological Review*, 95, 492–506.
- Luchins, A. S. and Luchins, E. H. (1959). *Rigidity of behavior: a variational approach to the effect of Einstellung*. Eugene: University of Oregon Books.
- MacKay, D. G. (1982). The problem of flexibility, fluency and speed-accuracy trade-off in skilled behavior. *Psychological Review*, 89, 483–506.
- Mandler, J. M. and Mandler, G. (Eds.) (1964). *Thinking: from association to Gestalt*. New York: Wiley (Original work published in Germany, 1913).
- Mann, T. L. and Hammer, J. H. (1986). Analysis of user procedural compliance in controlling a simulated process. *IEEE Transactions on Systems, Man and Cybernetics*, 16, 505–510.
- Mayer, R. E. (1975). Information processing variables in learning to solve problems. *Review of Educational Research*, 45, 525–541.
- Mayer, R. E. (1981). The psychology of how novices learn computer programming. *Computing Surveys*, 13, 121–141.
- Mayer, R. E. (1982). Contributions of cognitive science and related research in learning to the design of computer literacy curricula. In R. Seidel, R. Anderson and B. Hunter (Eds.), *Computer literacy*, (pp. 129–159). New York: Academic Press.
- Mayer, R. E. and Greeno, J. G. (1972). Structural differences between learning outcomes produced by different instructional methods. *Journal of Educational Psychology*, 63, 165–173.
- Merrill, M. D. (1983). Component display theory. In C. M. Reigeluth (Ed.), *Instructional-design theories and models: an overview of their current status* (pp. 278–333). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Merrill, M. D. (1987). The new component design theory: instructional design for courseware authoring. *Instructional Science*, 16, 19–34.
- Minsky, M. (1975). A framework for representing knowledge. In P. N. Winston (Ed.), *The psychology of computer vision* (pp. 211–277). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Morris, N. M. and Rouse, W. B. (1985). The effects of type of knowledge upon human problem solving in a process control task. *IEEE Transactions on Systems, Man and Cybernetics*, 15, 698–707.
- Morris, N. M., Rouse, W. B. and Fath, J. L. (1985). PLANT: an experimental task for the study of human problem solving in process control. *IEEE Transactions on Systems, Man and Cybernetics*, 15, 792–798.
- Naveh-Benjamin, M. and Jonides, J. (1984). Maintenance rehearsal: a two component analysis. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 10, 369–385.
- Neches, R. (1987). Learning through incremental refinement of procedures. In D. Klahr, P. Langley and R. Neches (Eds.), *Production system models of learning and development* (pp. 163–221). Cambridge, MA.: The MIT Press.
- Neisser, U. (1967). *Cognitive psychology*. New York: Appleton-Century-Crofts.
- Norman, D.A. (1981). Categorization of action slips. *Psychological Review*, 88, 1–15.
- Ohlsson, S. (1987). Truth versus appropriateness: relating declarative to procedural knowledge. In D. Klahr, P. Langley and R. Neches (Eds.), *Production system models of learning and development* (pp. 287–329). Cambridge, MA.: The MIT Press.
- Olsen, S. E. and Rasmussen, J. (1989). The reflective expert and the prenovice: notes on skill-, rule-, and knowledge-based performance in the setting of instruction and training (Technical Report). Roskilde, Denmark: Risø National Laboratory.
- Orata, P. T. (1928). *The theory of identical elements*. Columbus: Ohio State University Press.

- Osgood, C. E. (1949). The similarity paradox in human learning: a resolution. *Psychological Review*, *56*, 132–154.
- Pea, R. D. and Kurland, M. (1984). On the cognitive effects of learning computer programming. *New Ideas in Psychology*, *2*, 131–168.
- Perkins, D. N. and Salomon, G. (1989). Are cognitive skills context-bound? *Educational Researcher*, *18*, 16–25.
- Pieters, J. M., Jelsma, O. and Van Merriënboer, J. J. G. (1987, September). *Skill acquisition: ADAPT instructional time to desired level of transfer*. Paper presented at the Second European Conference for Research on Learning and Instruction (EARLI), Tübingen, FRG.
- Polson, P. G. (1987). A quantitative theory of human-computer interaction. In J. M. Carroll (Ed.), *Interfacing thought: cognitive aspects of human-computer interaction* (pp. 184–235). Cambridge, MA.: The MIT Press.
- Polson, P. G. and Kieras, D. E. (1984). A formal description of users' knowledge of how to operate a device and user complexity. *Behavior Research, Methods, Instruments and Computers*, *16*, 249–255.
- Posner, M. I. and Snyder, C. R. R. (1975). Attention and cognitive control. In R. L. Solso (Ed.), *Information processing and cognition: the Loyola Symposium* (pp. 55–85). Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Postman, L. (1971). Transfer, interference and forgetting. In L. W. King and L. A. Riggs (Eds.), *Experimental Psychology* (pp. 1019–1132). New York: Holt, Rinehart, and Winston.
- Postman, L. and Underwood, B. J. (1973). Critical issues in interference theory. *Memory and Cognition*, *1*, 19–40.
- Proctor, R. W. and Reeve, T. G. (1988). The acquisition of task-specific productions and modification of declarative representations in spatial-precueing tasks. *Journal of Experimental Psychology: General*, *117*, 182–196.
- Rasmussen, J. (1986). *Information processing and human-machine interaction*. New York: North-Holland.
- Reder, L. M. (1979). The role of elaborations in memory for prose. *Cognitive Psychology*, *11*, 221–234.
- Reder, L. M., Charney, D. H. and Morgan, K. I. (1986). The role of elaborations in learning a skill from an instructional text. *Memory and Cognition*, *14*, 64–78.
- Reigeluth, C. M. (1983a). Meaningfulness and instruction relating what is being learned to what a student knows. *Instructional Science*, *12*, 197–218.
- Reigeluth, C. M. (Ed.) (1983b). *Instructional-design theories and models: an overview of their current status*. Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Reigeluth, C. M. (Ed.) (1987). *Instructional theories in action: lessons illustrating selected theories and models*. Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Romiszowski, A. J. (1981). *Designing instructional systems*. London: Kogan Page.
- Royer, J. M. (1979). Theories of the transfer of learning. *Educational Psychologist*, *14*, 53–69.
- Rumelhart, D. E. and Ortony, A. (1976). The representation of knowledge in memory. In R. C. Anderson, J. R. Spiro and N. E. Montague (Eds.), *Schooling and the acquisition of knowledge* (pp. 99–135). Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Schank, R. C. and Abelson, R. P. (1977). *Scripts, plans, goals and understanding*. Hillsdale, NJ.: Lawrence Erlbaum Associates.
- Schneider, W. (1985). Training high-performance skills: fallacies and guidelines. *Human Factors*, *27*, 285–300.
- Schneider, W. and Fisk, A. D. (1982). Degree of consistent training: improvements in search performance and automatic process development. *Perceptions and Psychophysics*, *31*, 160–168.
- Schneider, W. and Fisk, A. D. (1983). Attention theory and mechanisms for skilled performance. In R. A. Magill (Ed.), *Memory and control of action* (pp. 119–143). Amsterdam: North-Holland.
- Schneider, W. and Shiffrin, R. M. (1977). Controlled and automatic human information processing: I. Detection, search and attention. *Psychological Review*, *84*, 1–66.
- Shea, J. B. and Zimny, S. T. (1983). Context effects in memory and learning movement information. In R. A. Magill (Ed.), *Memory and control of action* (pp. 345–366). Amsterdam: North-Holland.

- Shepherd, A. (1986). Issues in the training of process operators. *International Journal of Industrial Ergonomics*, 1, 49–64.
- Shepherd, A., Marshall, E. C., Turner, A. and Duncan, K. D. (1977). Diagnosis of plant failures from a control panel: a comparison of three training methods. *Ergonomics*, 20, 347–361.
- Shiffrin, R. M. and Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review*, 84, 127–190.
- Singley, M. K. and Anderson, J. R. (1985). The transfer of text-editing skill. *International Journal of Man-Machine Studies*, 22, 403–423.
- Singley, M. K. and Anderson, J. R. (1988). A keystroke analysis of learning and transfer in text editing. *Human-Computer Interaction*, 3, 223–274.
- Singley, M. K. and Anderson, J. R. (1989). *The transfer of cognitive skill*. Cambridge, MA.: Harvard University Press.
- Sternberg, S. (1969). Memory scanning: mental processes revealed by reaction time experiments. *American Scientist*, 57, 421–457.
- Thomdike, E. L. and Woodworth, R. S. (1901). The influence of movement in one mental function upon the efficiency of other functions. *Psychological Review*, 8, 247–261.
- Thomdyke, P. W. and Hayes-Roth, B. (1979). The use of schemata in the acquisition and transfer of knowledge. *Cognitive Psychology*, 11, 82–106.
- Van Merriënboer, J. J. G. (in press). Strategies for programming instruction in high school: program completion vs. program generation. *Journal of Educational Computing Research*, 6(3).
- Van Merriënboer, J. J. G. and De Croock, M. B. M. (1989, September). *Strategies for computer-based programming instruction: program completion vs. program generation*. Paper presented on the Third European Conference for Research on Learning and Instruction (EARLI), Madrid, Spain.
- Van Merriënboer, J. J. G. and Krammer, H. P. M. (1987). Instructional strategies and tactics for the design of introductory computer programming courses in high school. *Instructional Science*, 16, 251–285.
- Van Merriënboer, J. J. G. and Krammer, H. P. M. (1990). The “Completion Strategy” in programming instruction: theoretical and empirical support. In S. Dijkstra, B. H. M. Van Hout-Wolters and P. C. Van der Sijde (Eds.), *Research on instruction* (pp. 45–61). Englewood Cliffs: Educational Technology.
- Van Merriënboer, J. J. G. and Paas, F. G. W. C. (1989). Automation and schema acquisition in learning elementary computer programming: implications for the design of practice (Technical Report No. IST-MEMO-89-03). Enschede, The Netherlands: University of Twente, Dept. of Education.
- Van Rossum, J. H. A. (1987). *Motor development and practice: the variability of practice hypothesis in perspective*. Amsterdam: Free University Press.
- Voss, J. F. (1987). Learning and transfer in subject-matter learning: a problem solving model. *International Journal of Educational Research*, 11, 607–622.