Chapter 1

Introduction

Background

The research reported in this thesis grew from a period of study leave during 1989-90 spent at the Learning Research and Development Centre (LRDC), University of Pittsburgh, Pittsburgh, USA. During my period at the LRDC I worked with a research team, led by Stellan Ohlsson, on the development of an intelligent tutoring system, the *Fraction Tutor*. This form of software contains not only the content to be learned by students, but includes a student model that is meant to take the place of a teacher. The software stores each response of the user, recalls it when appropriate and generally sets the same kind of work, and gives similar hints and directions as a teacher would to the user's response. This is the *intelligent* aspect of the system.

I had come to the LRDC with some years of experience in teaching and writing in the fields of mathematics education and information technology education: and brought with me a range of values and beliefs connected to teaching, mathematics education and computer applications in education. My professional experiences were different from other members of the research team working on the Fraction Tutor.

A version of the Fraction Tutor had been tested on a small number of students prior to my arrival at the LRDC, and the learning outcomes were regarded as disappointing by the research team. I hinted that an essential part of the Fraction Tutor, a computerised version of a rectangle that could be subdivided to represent fractions, did not reflect the way a classroom teacher would use this representation.

The result of these and other comments about the system, resulted in my investigating the literature on concrete materials in mathematics education. Much of this literature was familiar, but less familiar was the cognitive science literature: literature concerned with intelligent systems, the role of analogy in learning, machine learning, perceptions of student learning, and the role and structure of intelligent tutoring systems. These investigations, together with a series of lively discussions, led to the development of the Procedural Analogy Theory (Ohlsson & Hall, 1990). A number of papers about this theory have been published[1], largely to gain the research community's reactions to the concepts expressed in the theory. The research reported in this thesis is the first empirical investigation of this theory, apart from a small scale pilot project conducted in 1990.

Ideology

What it means to teach and what it means to learn are far from simple notions. Ought we teach children to write by having them *be writers*, initially through invented spelling and grammar, or should we emphasise spelling and grammar early on so they are correct from the start? In the teaching of science do we emphasise facts, or is investigation to establish a sustainable version of the facts important? In mathematics, should we teach rules and then give practice exercises, or should we emphasise a more divergent problem solving approach? There is no one correct answer to any of these questions. Research may show that a particular pedagogy is effective: but a critic will always be able to question what is meant by 'effective'. This is made all the more difficult in the field of Education by the frequency with which contradictory research findings are reported and the inability of much work to be replicated. This comes about, not because there was anything wrong with the experimental design, but simply because the set of circumstances in the research was unique to that time and place.

What one person means by teaching may be not what a neighbour means by teaching, yet we may both be correct. *Teaching* for me requires the learner to be responsible for learning, particularly through problem solving emphases and through the development of investigation strategies. Teaching aims at the learner making meaning, and it is the learner who has to make this meaning; the teacher can assist the learning process but cannot create meaning for the learner. For other people, teaching means giving facts for children to learn and practise. I see this as a last resort and as an inferior approach - but no research I produce will convince my opponents that I am correct. They, just like me, have a strongly held set of values and beliefs about teaching. My research, and that of others, may provide support for my views, and may prove something to me - but this is because it is one of a range of truths, the one I happen to identify with. The same can be said for learning. What I mean by learning emphasises understanding, making meaning, the development of a richly-connected cognitive network of concepts and skills, and the ability to transfer and apply knowledge to a novel context. For others, learning may mean little more than the recollection of historical or mathematical facts, the ability to spell correctly or to remember rules, and to pass examinations. Leinhardt (1990) claims that

[1] For example, papers by Hall (1991, 1992a, 1993, in press)

> Education philosophies of teaching and learning are tangled up in political philosophies and are held as statements about the comparative virtue of the individual holding the particular perspective. The difficulty with the entanglement is that it leads to the usual round of name-calling and intellectual narrowness rather than a cooperative engagement in the task of steady improvement (p. 1).

The purpose of this statement of ideology is to recognise that this research was not conducted in a vacuum. It involved a researcher who brought to the activity a set of experiences and beliefs, and these matters cannot be totally set aside in research no matter how hard the researcher tries. Indeed Romberg, an internationally respected mathematics educator, claims that an individual researcher needs to address his or her writings conscientiously to "(t)he ideology and paradigms of different research communities" (1992, p.53). Though the ultimate research critique, and a blight on intellectualism, is the method of criticism by omission or failure to engage[2]. Millward (1980), discussing research into concept formation, noted that

> The major problem with the current state of theorising about concept formation is a lack of any integration of a number of reasonable and supported ideas. Researchers taking different points of view do not mention the work of others (p. 272-3).

Ideologies that I associate with mathematics education in schools see learners as individuals, see learning as an active and reflective process, and beliefs that learning cannot be *given* knowledge as in the common analogy of filling an empty jug. I believe learners have rights, so even if the 'fill-the-jug' model was plausible I would view it as unethical to subject learners held compulsorily in classrooms to lesson after lesson of inactivity, rote learning and subjugation. Nor are learners mice or pigeons whose behaviour is modified to some predetermined uniform blandness, conservatism or conformity through the use of rewards and punishments.

There is considerable support for these views if one looks to a sufficiently wide range of literature, as I have noted elsewhere (Hall, 1978, 1982, 1985). Both these views and those of a range of constructivists can find support in teachers' and curriculum developers' interpretations of Piaget's work, in the child-centred educational approaches of the Progressive Education movement, and in the range of educational values associated with names such as Rousseau, Froebel and Dewey. An important addition to this list of supporters is that of the influential Nuffield Mathematics Project dating from the 1960s where the idea of learning by doing was quintessential to the project's philosophy. Indeed the so-called 'old Chinese proverb'

> I hear and I forget,
> see and I remember,

[2] Perhaps the truly ultimate critique is to criticise the individual and his or her personal characteristics, experiences and behaviours. Such critiques have no place in genuine academic debate.

do and I understand

was a statement of the Nuffield project's philosophy. Given the lack of research in this area, and the general lack of empirical evidence at the time, these philosophical and ideological supports were important to those seeking change in mathematics education.

The study

This study involves two major fields of research: one area is cognition, the other mathematics education. More specifically, this research investigates the Procedural Analogy Theory, devised by Ohlsson and Hall (1990), as a basis for designing efficient instruction in mathematics, and for explaining the value of concrete materials in teaching arithmetic skills. The theory grew out of the unsuccessful development of an intelligent tutoring system, and involves concepts from cognitive science, including the role of analogy in learning, and the roles of practice and chunking.

The use of concrete materials in the teaching of school mathematics has long been supported by mathematics educators and by teacher educators. Such materials are common in schools, but generally used by few teachers, typically with younger children and then in idiosyncratic manners. The research literature has for many years reported contradictory findings. This research will examine the extent to which the procedural analogy theory provides effective guidelines for the planning and comparison of pedagogies before their implementation in classrooms, and the usefulness of a formula to calculate an isomorphism index which predicts the effectiveness of a given pedagogy.

Though the theory is not intended to be applied to individuals, it is more concerned with planning for group instruction; this research is concerned with individuals as learners. Luria's research, on simultaneous and successive patterns of brain functioning, appears to be an appropriate way in which to examine individual differences in the context of the application of the procedural analogy theory in classrooms, since they both have their foundations in cognitive processing. This Lurian theory is concerned with dimensions of cognitive processing, and has been applied to student learning in language and, to a lesser extent, in mathematics. Interactions between implementation of the procedural analogy theory and dimensions of cognitive processing will also be studied.

Chapter 2

Cognitive Science

Introduction

A range of concepts and issues typically associated with research, development and applications in contemporary cognitive science are considered in this chapter. These issues and concepts are not a comprehensive collection from the literature, but form a selection appropriate to the present work, where each of the issues raised will be applied at some later point in this thesis. There is a need to give at least a brief background statement as to what each concept or issue is about, where it came from, and what current research says about it, before later placing it in the narrower context of the present research focus. For example, consider the issue of learning by analogy. Analogy is important because it is such an everyday feature of human learning, and is used frequently in formal educational settings; it is also an important topic in cognitive science, particularly as it relates to machine learning. It is especially important to this thesis because the Procedural Analogy Theory that forms the basis of much of the research reported here has, as one of its quintessential elements, the application of analogy in teaching and learning.

Cognitive psychology is concerned with the human mind, with the development and processing of human thinking. Cognitive psychology has a long history. For example, even though Piaget may have thought of himself as an epistemologist, his work may also be seen to fall into the field of cognitive psychology. In their examination of human behaviours, there was something of a gap between North American psychologists, and their contemporaries elsewhere. For example, at a time when Piaget and other cognitive psychologists elsewhere were conducting investigations into human learning, United States' psychologists were typically more concerned with pragmatism, with action and observable behaviour. The literature of the period does not necessarily reflect this divergence, dominated as it was, and still is, by

North American journal publications, where levels of research funding and proliferation of graduate students are orders above what is available and produced elsewhere. In particular, with regard to educational psychology, Grinder (1989) argued that there was a loss of direction in the United States by mid-century, as a result of educational psychologists not addressing broader educational policy issues, through the lack of development of a coherent theoretical perspective, and by emphasising studies inconsistent with practical educational problems of everyday classrooms.

The behaviourist ideology that so dominated American psychology during these years meant widespread research into cognitive psychology was delayed until the 1960s (Anderson, 1985; Mayer, 1992; Schoenfeld, 1992). In the intervening years, psychology had turned to the study of rats rather than humans, emphasising animal learning and motivation, which may have been interesting, but had little relevance to cognitive psychology (Anderson, 1985). Cognitive psychology re-emerged in part through a combination of growth of knowledge about human information processing, largely concerned with tracing sequences of mental actions and their products, and closely related developments in computer science, particularly artificial intelligence[1]. The movement away from the behaviourists' stimulus-response perspective to a greater concern for cognitive processes and information processing (Di Vesta, 1989; Mayer, 1992; Scandura, Frase, Gagne, Stolurow, Stolurow & Groen, 1981), had been concerned with describing various stages in the information processing sequence, together with characteristics of these stages. Shuell (1980) argued that these changes were not particularly concerned with the study of learning, but both Mayer (1992) and Di Vesta (1989) later argued that there was a recognition of the importance of the learner in instruction.

Cognitive Science dates from the mid 1970s, and is a multidisciplinary field initially based on cognitive psychology and computer science, but also involving philosophy and linguistics, particularly through the use of computers to perform intelligent actions. The journal *Cognitive Science* was first published in 1976, with the field of cognitive science increasing in breadth as applications of these intelligent operations were applied in areas as diverse as education, science and business. Important issues from Cognitive Science that are relevant to this present work include the well-researched point that meaningful information is better remembered than meaningless information (Anderson, 1985), that meaningful information can be represented as propositional networks (Anderson, 1985) or as planning nets (VanLehn and Brown, 1980; VanLehn, 1988), and that analogy is helpful in transferring meaning both from teacher to learner and from the known to the unknown. This thesis is concerned with aspects of cognitive psychology, with cognition and with cognitive science. This work is also concerned with the concept of chunking, where linked concepts are stored and used as if they were only one concept, and so is concerned with working memory and with long-term memory. That is, the research reported here contains elements of all Mayer's (1992) three

[1] the journal *Cognitive Psychology* was first published in 1970.

views of learning and instruction. The research is concerned with the automatisation of basic skills (learning as response acquisition), with increasing knowledge (learning as knowledge acquisition), and with learners as constructors rather than recipients of knowledge (learning as knowledge construction). Mayer also described the movement away from a general theory of learning to one that is more subject specific, and the movement away from looking only at the average learner to a more detailed study of individuals as they learn. The present research is consistent with this pattern, in its concern with the learning of one topic within mathematics, its use of a detailed instructional strategy, and its concern with learning taking place within groups, with individual learner's mathematical knowledge, and with the study of individuals' specific cognitive processing styles.

An important part of all learning, and of all problem solving, is human working memory. Working memory, which can also be thought of as short-term memory, loses information rapidly[2]. Without rehearsal or practice, information entered into working memory will begin to decrease immediately, and after nine or ten seconds has generally been lost (Anderson, 1985). The amount of data we are able to hold in short-term memory limits our mental capacity, our ability to process large quantities of data quickly and simultaneously. For example, in completing a written multiplication algorithm, a student has to recall each multiplication fact required by the components of the algorithm. Such recall places minimum strain on short-term memory when the student is able to immediately recall the multiplication fact from long-term memory. However, if the student is unable to recall the number fact from long-term memory, then each multiplication fact will have to be calculated in short-term memory. In such cases, the student has both to calculate the particular fact, and to continue to be aware of, and move towards completion of the larger algorithm. In these cases, students' short-term memories are likely to be overloaded, and so errors will result as students lose their way. In school settings, this is all too frequently visible as students use fingers or tally marks to calculate number combinations which should have been committed to memory. While their working memory is taken up with this counting, they increase the likelihood of forgetting where they were up to in the procedure, or do not have sufficient memory capacity for their working memory to deal with the problem at hand. Miller (1956) is well known for his research indicating that working memory is able to cope with 7 plus or minus 2 items. One way to increase the efficiency of short-term memory is to ensure that prerequisite skills are readily available, such as number facts or spelling, and to decrease the short-term memory required through chunking, that is, by abbreviating the information. For example, the telephone number 321250, could be though of as six digits, and so would be six pieces of data in short-term memory. Of course this is highly inefficient since it would take up most of an individual's short-term memory. The telephone number could also be thought of as thirty two, twelve, fifty so reducing the data pieces to three: but the number also lends itself to the sequence 321 (sequential digits) and 250 (a round

---

[2] In this thesis *working memory* and *short term memory* are terms that will be used interchangeably.

number, easily recalled) reducing further the call on short-term memory space. There are hundreds of such facts encountered in everyday life, especially for those still studying in educational systems.

The brain uses both working memory and long-term memory in learning, storing and retrieving information. Long-term memory is where data are stored in the brain for long periods of time, generally inactive; when data are retrieved from long-term memory they become part of working memory where they can be manipulated. Well-practised facts are retrieved much more quickly from long-term memory than less practised facts (Anderson, 1985), and one would expect long-term memory to be more efficient through chunking. In considering ways of organising teaching materials, or in considering aspects of pedagogy it is useful to help learners "organise material in such a way that subjects can systematically search their memories for the items" (Anderson, 1985, p. 183): such efficiency comes from a form of chunking, where related items are linked together cognitively, so recall is aided since the identification of one fact allows the recall of others through links in individual's cognitive networks. That is, the development of a richly-connected cognitive network of concepts and skills will assist both working memory and long-term memory, and the storage and retrieval processing within and between them.

This storage of information leads to the questions of how knowledge is represented and stored. These are important aspects of cognitive science. For example, how is knowledge best represented in a computer program, and what are the characteristics of those interfaces most likely to assist users in both learning this knowledge, and establishing an efficient cognitive network for themselves? Other examples concern teachers wanting to use effective representations to help learning, to assist the development of concepts and processes. What are the characteristics of 'good' representations? For learners, ineffective representations may be worse than useless in that they are poor examples of the concept or skill, and so encourage the development of faulty information and poor cognitive structure (Anderson, 1985; Larkin & Simon, 1987). The issue of representation is one of particular importance to mathematics education, and increasingly important in the field of information technology in education; it forms an important part of this present work.

Declarative and procedural knowledge

Cognitive science and cognitive psychology are concerned with how learning takes place, that is, the process by which knowledge is gained. Especially in formal learning situations, the process of learning involves declarative and procedural knowledge. This raises many questions. For example, what is declarative knowledge, what is procedural knowledge, why are they important, what do they tell us about learning and what implications do they have for guiding instruction? Further, how are declarative and procedural knowledge linked to the use of models, representations and concrete materials in teaching?

Declarative knowledge is that which the learner internalises on hearing the teacher make some statement about a fact or process, but declarative knowledge is limited in that the learner may be able to repeat it to the teacher but will be unable to put it into practice. That is, the learner in a sense 'knows' what the teacher said, is aware of the facts, but is unable to operationalise it (Neves & Anderson 1981; Mostow, 1983; Ohlsson, 1991). Procedural knowledge is the knowledge needed to take what the student knows declaratively and put it into practice. These distinctions are important because they suggest that learning is not a simple process where knowledge is somehow fed directly into the student by the teacher, that learning is not a one-step process, and they illustrate the need in instruction both to explain clearly and to help operationalise whatever is being taught. The distinction between declarative and procedural knowledge also draws out the point that while information may be given by the teacher, it is the learner who has to re-construct that knowledge, to give it meaning by fitting it in with an existing cognitive structure, or by amending the structure. For example, a definition of some mathematical or scientific property would be declarative knowledge; giving examples of the rule, particularly through providing learning activities where the learner applies the rule helps proceduralise that rule. Of course, there is more to good teaching and to learning than this simplistic two-step process, but the process does describe some of the essential elements of declarative and procedural knowledge. The role of modelling would appear to be important simply because it does exactly what it says: it models the procedure. Of course, there are many forms of models and modelling, from pen-and-paper examples of applying rules, to physical embodiments intended to represent the underlying concepts and principles, and to powerful and intricate computer-based models (Reif, 1987).

As Ohlsson (1991) claims, declarative knowledge may include facts, events and generalisations; it is descriptive and is constructed of propositions, whereas procedural knowledge is prescriptive. Procedural knowledge includes strategies, tactics, heuristics, plans and the like, so that, given a situation and a desired goal, by following a particular set of rules the given goal will be satisfied. Declarative knowledge involves generalisable facts, but their application requires procedures that are domain or instance-specific (Millward, 1980; Ohlsson, 1991). It is this procedural knowledge that allows rapid response to everyday situations. Both declarative and procedural knowledge are essential for intelligence (Neves & Anderson, 1981), but there are many aspects of the creation and roles of declarative and procedural knowledge which as yet we do not know (Ohlsson, 1991).

One of the consistent findings about novice learners in a domain is that given the opportunity to read instructions of modest complexity they are not able to perform the described operations without error (Anderson, 1986). According to Anderson's ACT[3] theory (1983), instructions are initially stored in a declarative form, but behaviour requires procedures that are represented as productions. Instructions are insufficient to set up procedures to perform skills,

[3] Adaptive Control of Thought

interpretive productions must convert this knowledge into behaviour. The ACT system is a series of *if-then* rules: the *if* jogs memory by recognising or matching some incident, so that the *then* part is executed. Anderson describes the importance of procedural skills in terms of increasing efficiency of completing tasks, "proceduralisation is a process that builds specialised versions of productions by eliminating retrieval of information from long-term memory...the information that would be retrieved from long-term memory is encoded directly into the specialised version of the production' (1983b, p. 207). That is, Anderson is claiming that procedural skills are important in that they allow learners to proceduralise solutions to problems and algorithms in an efficient manner, and that this proceduralisation minimises the need to refer to long-term memory in order to complete the process. This view is consistent with the role of practice, with the notion of increasing performance speed, with the importance of chunking steps in procedures and with the importance of automatisation of procedures. Proceduralisation allows learners to perform operations, and through a process of simplification and chunking, allows sets of procedures to be collapsed into one procedure. This means a sequence of actions to solve a particular problem can be reduced to a smaller number of procedures, and in some instances, without having to refer to long-term memory (Anderson, 1986).

In the context of teaching and learning mathematics it is not difficult to agree with VanLehn and Brown's claim that "we have all been forced to learn the procedural skills that supposedly comprise mathematical literacy... through the process of rote memorisation, perhaps, enhanced by the use of 'models' " (1980, p.95). How were we taught these skills? Typically we were given exemplars of correctly solved problems, perhaps an everyday problem or more likely just numbers to manipulate, and there may have been some form of representation or embodiment of the concepts and process being illustrated. Generally these models were intended to provide the learner with an intuitive basis for a given procedure, clearly intended to help learning. Several important questions come from this common practice. What is meant by a 'model' of a procedural skill, what is being modelled, what distinguishes a good model from a not-so-good one, and to what extent must the model look like what it actually represents? How does all this help the learner either to learn the procedure or to make meaning from the procedure? That is, as VanLehn and Brown imply, not only ought we be concerned with the surface structure of a procedure but also with its underlying connections to what the learner already knows. Anderson (1986) supports such a view when he talks of learners' needs both *to know* and *to know how to* before they are able to efficiently apply new learning. In addressing this issue, VanLehn and Brown (1980) examine the details of the design of a procedure, its prerequisite knowledge and other concepts and principles to which it is related. This exploration takes place in the context of planning nets which are discussed later in this chapter.

A Teaching Perspective

Areas of cognitive science have relevance to educational practice in classrooms: consider the case of declarative and procedural knowledge. Teachers' lessons involve both declarative and procedural knowledge (Anderson, 1983; Leinhardt & Smith, 1985). The declarative knowledge involves the facts associated with the teaching content, the procedural knowledge involves the algorithms and heuristics used with those facts. Declarative knowledge is frequently represented as a network of interconnected concepts and skills. For example, Figure 2.1 shows a network for the concept of addition of fractions.

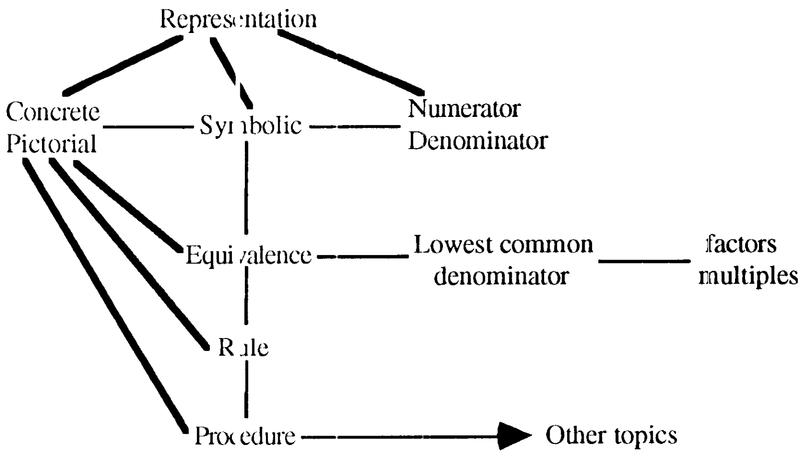


Figure 2.1: Networks[4]

Procedural knowledge is frequently represented as a series of production rules (Anderson, 1983; Newell & Simon, 1972; Ohlsson & Hall, 1990). There is a number of forms of such production rules, for example, Figure 2.2 shows the logic of each step for the addition of two two-digit numbers.

Rule 1:	If then	the goal is to add two two-digit numbers add the digits according to their place value
	If then	the goal is to add the digits according to their place value add the digits in the units column
	If then	the goal is to add the digits in the units column add each digit on the right hand side
	If then	the goal is to check the size of the sum of the digits compare the sum with 10
	If then	the sum is less than 10 write the digit in the units answer position
	Else	write the right hand digit in the units answer position write the left hand digit in the tens column

---

[4] The term network has been referred to by other names including semantic network, cognitive structure and schema.

then	add digits in the units column
If	the sum is less than 10
then	write the digit in the tens answer position
Else	write the right hand digit in the tens answer position
	write the left hand digit in the hundreds answer position

Figure 2.2: Production Rules

Though declarative and procedural knowledge have importance in classroom practice, it is unlikely that teachers know much about, or give much consideration to, these concepts as they plan and implement instruction. In particular, it would be most unusual for teachers to think much about the kinds of networks represented by Figure 2.1 or the production rules represented in Figure 2.2. It is more likely that teachers are concerned with the broader aspects of the content they are teaching. Their professional training and experience, together with curriculum statements focussing on content, predispose them not to consider the minute detail of instruction and the inter-relatedness of concepts. Yet this very detail has to be learned by the student if he or she is to develop rich understandings of the inter-relatedness of much knowledge.

The research reported in this thesis is concerned with the fine-detail of instruction and learning. It is especially concerned with the use of concrete materials to assist the development of declarative and procedural knowledge; and with the role of analogy in efficiently using such materials to perform mathematical actions and through this, to represent mathematical ideas symbolically.

Analogy

An analogy is where the circumstances of one situation are seen to have a similarity to the requirements of a new situation, where the steps in one series of actions seem to have parallels to the series of steps in another action. In everyday life, and in classrooms, it is common for one person to describe an event or action to another as *like* something else, that is, to draw an analogy from a common experience to one not experienced by the second person. As Holyoak (1984) says "the function of analogy is to allow transfer of knowledge from a known situation to a novel one" (p. 200). For VanLehn and Brown (1980) an analogy is

> a comparison of two 'things' that can be broken down into three parts: (1) an analysis of the first thing into some abstract description (or deep structure); (2) an analysis of the second thing into another abstract description; and (3) a mapping between the two descriptions (p. 98).

This three-component view is the foundation of the general theory of analogy. The cognitive structures of solving a novel task analogous to a task whose solution is already known requires the two abstract descriptions and the mapping between them. The expression of the two analogous sets will differ as their contexts differ. They will each have a unique chain of semantic links, but there will exist nodes of commonality between them in order for there to be an analogy. From this a mapping or match between semantic nets can be thought of in terms of links that have a common label, the links may not be identical but will represent equivalences. In such cases the isomorphism, or correspondence, of the two different sets can be thought of as a "one-to-one correspondence on the links that preserves the adjacency, direction, and label of the links" (VanLehn & Brown, 1980, p. 100-1).

For Carbonell (1983) "analogy is one of the central inference methods in human cognition as well as a powerful computational mechanism" (p.137). Carbonell's central theme is that in a new problem situation an individual looks for past experiences highly similar to the present problem. He argued that in solving a problem, a person looks to similarities with past experience, so as to retrieve behaviours that were successful in the earlier instance, and claimed that such transfer is "inextricably woven into the pedagogical practices of our educational institutions" (p.140). Anderson (1983) agrees when he argues that learning by analogy is effective because people have a facility to partially match the background and givens of one problem to the background and given of another problem. In this way, people retrieve earlier behaviours that were effective in solving problems, that can now be adapted to the demands of the present problem. Holyoak (1984) notes the importance of analogical thinking, that "it is natural to view the development of a theory of analogical thinking as a major focus of cognitive science" (p. 200), and believes that successful mappings between analogies can come about because there are mapped identities, the core meanings are the same, and through structure preserving differences, where the transformation remains clear because the set of schema in one situation are identifiable in the other domain. He argues that failure in mappings come about because of structure violating differences, where there are inconsistencies and a lack of parallelism between the two situations. and through indeterminant correspondences, where aspects of the base and the target are not mapped.

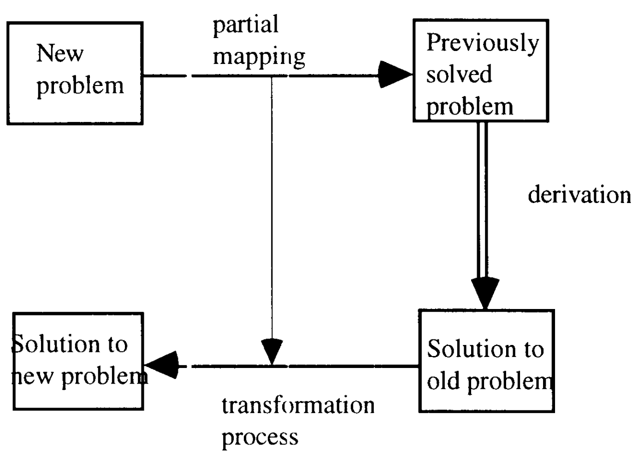


Figure 2.3: Transformational analogy
(Modified from Carbonell, 1986, p.376)

Later, Carbonell (1986) distinguished between transformational analogy (Figure 2.3), where an earlier solution can readily be applied to a new problem, and a derivational analogy (Figure 2.4) where the transfer from old to new was not so obvious and required some further analysis. The derivational analogy is the stronger in the sense that the transformation analogy is a relatively simple copy, while the derivational model requires the learner to follow the reasoning and proceed through the steps that led to the previous solution, that is to reconstruct the relevant aspects of earlier solutions then apply them to the new problem. Derivational analogy is a powerful reasoning tool, one that draws upon past experiences, encourages gradual chunking of parts of these experiences, and so increases problem solving expertise in novel situations.

Even quite simple analogies will often allow for a range of inferences to be drawn, when the teacher is actually hoping for a particular one: hence the need for instruction to move forward in small steps. Burstein (1986) stresses the importance of applying incremental steps when learning through analogies, particularly the need to ensure errors in moving from step to step are quickly identified and corrected. Holyoak (1984) mentioned the importance of giving learners a hint to increase the likelihood of using an analogy, such as a reminder that there may be something in a problem or story previously studied that may help in solving the present problem. Holyoak expressed surprise that learners did not notice analogies, and argued that this came about because the story was seen simply as a story and not as a 'problem'; and therefore the initial encoding was not efficient. That is, learners will benefit if a teacher calls their attention to the potential analogy. It will be argued later in this thesis that this is a crucial element in the use of manipulative materials in the teaching of subtraction algorithms.

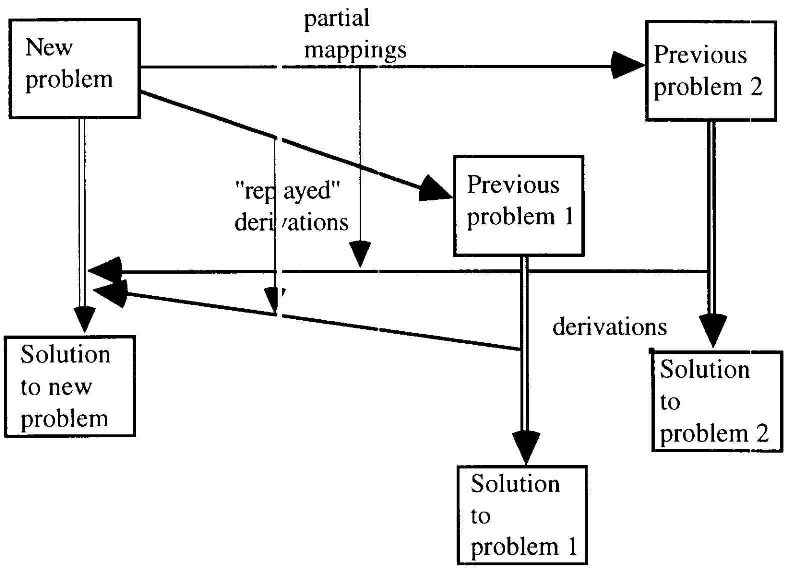


Figure 2.4: Derivational analogy
(Modified from Carbonell, 1986, p.377)

Anderson (1983) suggested that learning by analogy applied only in the short-term, while the learner was still able to recall similar problems. He also suggested that in learning by analogy, similarities had to be emphasised for the analogy to work. So that, in the case of school mathematics, one of the elements of effective teaching is establishing analogies between manipulative materials and written symbols, so that they not only assist learning at the time, but that they provide a model for later retrieval and reflection. In this way, the use of some earlier procedure in explaining a present procedure, attempts not only to make sense of the latter procedure in the short-term, but attempts to provide a tool for understanding now and in the future. Indeed, meaningful analogies must go beyond similar problem statements, to the structures of their solutions, and so have a temporal presence. Anderson acknowledges these points when he argues the importance of structural analogy in solving problems similar to ones already solved, and the importance of knowledge compilation in extracting new production rules from solving problems, so that future problem solving is streamlined (1986). He also mentions the need to see knowledge compilation as more than making existing paths of processing more efficient, by allowing for the possibility of novel pathways to be developed, and this is what is meant above when reference is made to later retrieval and reflection, and a tool for future understanding. Burstein (1986) mentions the need to know a good deal about a topic already in order to learn more: "it has often been said among (artificial intelligence) researchers that learning something new requires knowing a lot about it already. This is certainly true for learning by analogy" (p. 351). So analogies are useful, but one has to have sufficient knowledge to take advantage of them. Indeed, errors in using analogy for learning are almost inevitable when the learner has very limited knowledge of the domain, and this should be expected since analogies are useful but not perfect matches between situations.
It's clear then, that analogies are an important learning tool, and are an important part of cognitive science literature. Looking briefly now to the role of analogies in mathematics learning, VanLehn and Brown (1980) describe the use of Dienes blocks[1] in adding two digits. These blocks are used to represent numbers, and provide a tool for learners to act on. The blocks are related by place value, so that the Base 10 blocks, have a one centimetre cube referred to as a unit, a 10cm length with markings at each centimetre referred to as a long or ten, a 10cm by 10cm square representing 100 (a flat), and a 10cm by 10cm by 10cm cube representing 1000 (a block). There are other bases available in MABs, with the unit remaining the same size, but the long, flat and block altering according to the base. These materials are widely used in whole number work, to a lesser extent in fractions and decimal representations, and widely in representations of the four operations. VanLehn and Brown (1980) note that they "were recently struck by the way Dienes Blocks were being used in a school" (p. 97), where in the operation of addition, the blocks were not manipulated so as to answer the units section,

[1] Dienes' blocks are generally called Multibased Arithmetic Blocks, which here and elsewhere, is frequently reduced to MAB, MABs, MAB blocks, or MAB materials.

then the tens, and so on, but were piled together and only then considered in their place value. In such a case the trading aspect of addition was delayed until after the actual addition of the numbers as represented by the blocks being joined together. They commented that "(o)ne intuitively feels that this second, two-pass procedure is not as closely analogous to standard addition as the previous, one-pass Dienes Block procedure" (p. 97), and go on to say that a "theory of analogy should have some formal measure that can predict how close an analogy is" (p. 97), and suggest using a *closeness metric*. This closeness metric is a measure of the match between the three parts required for an analogy to occur: the original abstraction or procedure, the new, modified or target abstraction and the mapping of the first onto the second. Put simply, the closeness metric is the degree of closeness, or correspondence, between the original situation and the new analogous situation. In their paper, measures for the closeness metric are arbitrary in the sense that experts are asked to judge the closeness of analogies. VanLehn and Brown (1980) note the difficulty of quantifying their closeness metric, noting "there are many difficulties and fine points involved in determining such a definition" (p. 120), but recognising that "the weight of some planning inferences is quite close to zero" (p. 120). As we will see later, the procedural analogy theory has a much more objective, ordered, quantifiable and generalisable approach to measuring degrees of analogy[6].

In dealing with subtraction, VanLehn and Brown (1980) consider three possibilities with MAB materials. One possibility is to use only units where subtraction involves taking one unit at a time (though they use a method of pairing of which could be criticised since it misrepresents the operation of subtraction). A second possibility involves all blocks placed together regardless of their size, with the third possibility that where blocks are placed in categories related to their place value In contrasting these methods, the authors note that the analogies reflected by the arrangement of nets, and the number of nodes and links they have in common, does not always reflect intuition. That is, we find here an instance of what seemed to be analogous. is not. Perhaps, more correctly, we should say that this example illustrates that there are varying degrees of analogy, from no analogy at all to a perfect analogy. The case here falls somewhere between these extremes.

In their first subtraction method, using only MAB units, they place two piles of materials to represent the numbers involved in the subtraction. It could be argued, though, that this is the first error. Addition is the joining together of sets of objects to form a third set, but subtraction is the removal of elements from a given set. That is, a high level of analogy would require only one set of MAB units to be used, and the subtraction involve removing objects from that set. The next step in VanLehn and Brown's subtraction, remembering that there are still two piles, is to move one unit from the minuend, find its partner in the subtrahend and remove both. The analogy of subtracting one thus becomes *find one unit from one place, find another in a second place, form a pair, then remove them* This is confusing: a learner could quite justifiably think

---

6 Refer to the Procedural Analogy Theory, chapter 4.

'aren't I subtracting two?' or 'haven't I taken two away?' Further, this method suggests that subtraction involves subtracting one at a time. This is appropriate when exploring materials to discover their properties and to use them when the concept of subtraction is first introduced, but does not correspond to the procedure for completing a written subtraction algorithm where numbers other than one are regularly subtracted. This approach also ignores place value so the materials do not reflect numbers in an efficient manner, and as a consequence of this, the materials cannot show the trade procedure so necessary in subtraction requiring decomposition. Taken together, materials used in this way provide a poor analogy to the concept of subtraction, provide little correspondence as to the structure of subtraction algorithms and the procedure for completing them.

In the second subtraction method suggested by VanLehn and Brown, the full range of MAB materials are used but materials are still brought together to form two piles of blocks, representing the minuend and the subtrahend. The difficulty of representing both piles, of subtraction as 'pairing off' stills exists. Setting this aside for the sake of the argument, the use of the range of materials, as opposed to using only units is an improvement from pedagogic and meaningful learning points of view, but the other weaknesses outlined above for the first methodology still exist. In particular, there is no necessity to trade in the manner that a written algorithm requires, the sequence in which trading is carried out and the exact nature of an individual trading sequence are arbitrary. In this manner the materials are not being used in as sound a pedagogic approach as they might, there are more opportunities for learning incorrect ideas than there need be. One of the benefits of MAB embodiments is that they so readily reflect the concepts and processes of arithmetic operations. There is clearly a place, indeed an importance, for allowing learners to experiment with MABs, but the argument here is concerned with the point where the MAB procedure becomes formalised, so as to eventually lead to a corresponding written algorithm. To some extent, grouping materials according to their place values and then going through the sequence – units first, trade if necessary, subtract units, now the tens, and so on – is forced on the learner by the nature of the materials. Further, used in this manner, the materials better reflect numbers and operations, show why subtraction algorithms are structured and processed in the manner they are, and so provide a meaningful embodiment on which learners can act. This latter description is that of VanLehn and Brown's third method, the only one that would be reasonable in a classroom where the teacher was at the point of attempting to use MAB to model or develop a conventional subtraction algorithm.

In contrasting the second and third methods, VanLehn and Brown (1980) suggest "(i)ntuitively, these two procedures are quite closely analogous" (p. 105), but this is obviously not so: the use of one pile of MABs hides the place value component of this operation, muddies the decomposition concept with its trading operation, decreases the level of analogy between actions with materials and the corresponding written algorithm, and so lessens the value of using MAB materials. VanLehn and Brown's third representation shows the place value of the number, the necessity of trading questions and the corresponding algorithm, which after all is one of the major reasons for using such materials. For all the educational value of their first or second representation, one may as well use any form of representation, or indeed a calculator. That is, the intuition here is an ill-informed intuition.

However, continuing to accept their intuition for argument's sake, there remains the problem of quantifying or recognising the 'better' approach. Once again, VanLehn and Brown take the path of analysing the situation in its own right, as if it were a closed system, whereas the analysis needs to look at the pedagogy involved and the purpose of using these materials. Modelling algorithms using embodiments is not just to reflect the solution to a problem or to reflect experts' thinking about or solving such a task. The materials are there to allow complex and abstract ideas to be represented and acted upon, with the intention of allowing learners to construct meaning through the successful completion of algorithms at first using the materials alone, but at all times from the teacher's point of view of moving closer towards a formal, conventional symbolic representation of the algorithm. Indeed, VanLehn and Brown recognise the range of possibilities in representing subtractions, and claim there are numerous possibilities between subtracting by 'one unit at a time' and the standard written algorithm, but ask the question "how should the intermediate models be sequenced?" (p. 125). The procedural analogy theory described later provides specific and detailed guidelines on this, interpretable as a guide to instruction. Analogy then, is an important component of teaching and learning, and in this thesis. Yet it is quite complex, as the discussion of VanLehn and Brown's work has illustrated. This complexity is increased further, through the application of analogy to computer based systems.

One of the roles of contemporary cognitive science is the development of intelligent software, that is, the creation of computer programs that will either assist human decision making, or that are able to learn in their own right. This endeavour falls into the field of *artificial intelligence*. There has been a number of approaches to problem solving in artificial intelligence. The most common approach involves a computer selecting a sequence of operators from a given set so as to change an initial problem state into an intended goal state (Carbonell, 1983; Newell & Simon, 1972). There is a second view where problem solving is seen as a planned increment in movement towards a solution, where there are no specific approaches to select from and where approaches have to be developed through subtle changes to likely directions in a recursive manner (Carbonell, 1983; McDermott, 1967; Wilensky, 1978, 1983). A third approach is to solve problems through analogy, possible because the machine recognises commonalities between the present problem, and problems previously solved (Carbonell, 1983, 1986). Such approaches have their basis in the observation of human behaviour, where computer systems are built that attempt to replicate such human behaviour. Carbonell believed that analogical problem solving consisted of transferring knowledge from past problem solving episodes to new corresponding problems using the transferred knowledge to construct solutions to the new problems (1986). Correct approaches encouraged generalisation while incorrect approaches caused cognitive adjustment. This works well for humans, but Carbonell (1983) noted that whereas humans have the ability to learn from all tasks, artificial intelligence systems seldom have this adaptive quality. This is supported in part by Anderson (1983) who mentioned that his subjects did not plunge into endless searches looking for solutions to posed geometry problems, they restricted the paths they followed, and one approach was based on analogy to prior problems.

The point of including the above material here is to make explicit that this thesis, in part, is about intelligent computer systems their specification and development[7]. This material is included too, to illustrate the point that not only humans learn through analogy; it is a method that is being applied in intelligent computer systems (Lenat, cited in Michalski, et al., 1986). As Dershowitz (1986) says "analogy is a tool that automatic programming systems can use to learn from experience, just as programmers do". Dershowitz reports on an artificial intelligence program that debugs incorrect program codes and modifies existing programs to perform different tasks. In this way he hopes that the great bulk of time that programmers spend debugging code, adapting known techniques to similar problems, and modifying existing code to new specifications can be formalised and so incorporated into automatic programming techniques. That is, here is a clear example of intelligent machines performing useful tasks through applying analogies[8]. There is then the question of just how do machines learn? Intelligent tutoring systems are one area of the field of artificial intelligence. In relation to the background of the research reported here, particularly with regard to intelligent tutoring systems, the question arises as to how machines learn. That is, if artificial intelligence aims to make machines perform analyses and predictions that would require intelligence if done by people (Minsky, 1968), how are these machines to learn from their experiences? The issue of machine learning, fundamental to developing applications in artificial intelligence is now considered. This topic is relevant because the models of machine learning provide valuable comparisons for human learning.

Machine learning

The purpose of this section is to provide a brief description of some of aspects of machine learning that are relevant to the present thesis. A major reason for including machine learning in this section is that it is indelibly linked to that part of artificial intelligence concerned with developing intelligent machines. That is, with developing computer systems where the machine learns (Gilmore & Self, 1988). Secondly, the development of machine-based intelligence

---

[7] The manner in which this research grew from work with an intelligent tutoring system is described in the Background, Chapter 1.

[8] Dershowitz (1986, p.394) cites numerous references to where human and automated analogical reasoning has been reported in the literature.

generally requires some form of knowledge base, that is, knowledge has to be represented in some way (Anderson, 1988; Burns & Capps: 1988, Cerri, 1988). In representing knowledge in various ways, there may be insights into human learning and knowledge representation, and this has implications for the procedural analogy theory discussed later in this report.

How humans learn has long been the object of research and discussion for biologists, psychologists, educators, philosophers and artificial intelligence researchers. Understanding human learning sufficiently well to be able to reproduce aspects of it on computer systems is a worthy scientific goal, and this may assist cognitive psychologists in testing consistency and completeness of learning theories. There are important areas of practical significance for this research too, for example, in the development of intelligent tutoring systems (Carbonell, 1983). The study and computer modelling of learning processes form the basis of machine learning. Machine learning is concerned with developing and analysing learning systems to improve performance on a set of tasks, for specific computer based applications, in cognitive modelling where computers are programmed in an effort to simulate human learning processes, and with the theoretical exploration of possible learning methods and general algorithms independent of the application domain (Carbonell, 1983; Michalski et al., 1986).

Machine-learning approaches may be classified according to the underlying learning strategy being investigated. For example, there is rote learning, where learning means remembering facts; there is learning from instruction, for example, by being told, or learning from a book; and, there is learning by deduction, where learning takes place by analysis. There is also learning by analogy, which requires more inference than these other learning strategies. In computer terms, learning by analogy would involve designing a system that can convert an existing program into one that performs a closely-related function.

According to Michalski, Amarel and Lenat (1986), there are three major research paradigms in machine learning:

> neural modelling and decision theoretic techniques, where systems start with little knowledge and a network of elements. Such a system learns by incrementally modifying the network, by strengthening some connections within the network.
> symbolic concept acquisition, where a system learns through the analysis of examples and non-examples, to develop a symbolic representation of a given set of concepts. Such systems typically have decision trees, production rules or semantic networks to provide the logical base for the system.
> knowledge-intensive, domain specific learning, where the system contains numerous predefined concepts, knowledge structures, domain constraints, heuristic rules and built-in transformations relevant to the specific domain of that system. The system learns new concepts as it operates (1986, p. 12ff).

Simon believes learning has taken place when the system doing the learning is able to perform the same task more efficiently the next time (1983). He regards human learning as inefficient and asks if "the inefficiencies of human learning derive from peculiar properties of the human information processing system (or) will they be present in any system that tries to extract patterns or other kinds of information from complex, noisy situations and to retain those patterns in a manner that makes them available for later use" (p. 26). As he says, human learning is slow, and its transfer to other areas is problematic. This, of course, is quite unlike machine learning where once the program runs correctly, that's it, it can be copied and transferred to any number of computers. In looking at the need for research in machine learning, Simon (1986) suggests that high priority be given to research aimed at simulating, and so understanding human learning, particularly basic research aimed at understanding why human learning is so slow and inefficient, with the hope that machine learning can be devised that will avoid, for machines as well as people, some of the tediousness of learning.

Traces, Planning Nets

The purpose of this section is to explore the possible representations of cognitive procedures; that is, to consider ways in which to represent the abstract in a more accessible form. Given the importance of the role of analogy, and given the significance of machine language to contemporary cognitive science, how can knowledge and procedures best be represented? In particular, are there ways of representing knowledge and procedures that reflect human cognition and information processing? These representations are important, because they have the potential for helping the specification of intelligent computer software, and for providing insights into possible pedagogies for teachers. Included in the possibilities are traces, flowcharts, planning nets, production rules, cognitive networks and procedural sequences.

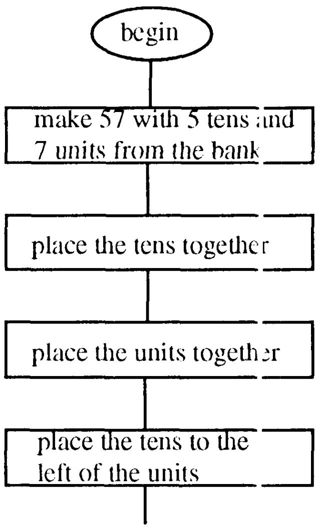
One way of representing a procedure is as a *trace*. According to VanLehn and Brown (1980) a trace of a procedure "is simply a chronological list of the actions it performed during one particular execution" (p. 103). They argue that traces are superficial, and give an example of writing '4' as equivalent to four actions (the action of obtaining one MAB unit 4 times), but this ignores the reality of having four units and writing "4" to represent them. That is, one can analyse the use of MAB and other representations at too microscopic a level to seek analogies that are not there. While the choice of a suitable level of analysis may seem pragmatic, the value of observing expert teachers successfully using these kinds of models is a justifiable and powerful mechanism for coming to understand both their educational value in assisting learning and for gaining insights into the cognitive actions and structures that actions on these materials may represent. Indeed, VanLehn and Brown note such a possibility in their footnoted reference to 'the grain-size issue', and note there has to be a balance between a practical solution and a source of uncertainty in the theory. An example of a trace is shown in Figure 2.5.

Figure 2.5 illustrates the step-by-step action in solving the subtraction 57-19, using MAB materials. It operates at the level of 'doing' rather than thinking or analysing, and it reflects physical actions rather than cognitive actions on the part of the learner. The value of a trace then seems connected to the level at which the trace is constructed. Is it a macro or a micro view of a reality, exactly what is to be incorporated into it, and how is the resulting trace to be used? For the present, answers to these questions are sufficiently vague to suggest the value of looking at alternative representations.

Take from the bank 5 tens and 7 units
Place the tens together, and the units together
The tens to the left of the units
Return 1 ten to the bank, exchange it for 10 units
Place these with the existing 7 units
Take 9 units from the 17 units
Count remaining units
Write the number in the units answer space
Take 1 ten from the remaining 4 tens
Count remaining tens
Write the number in the tens answer space

Figure 2.5: Traces for the question 57 -19.

An extension of the notion of traces is the use of flowcharts to outline the sequence of steps to move to the completion of a particular question. The characteristics of flowcharts are that they indicate a sequence of actions, where there is a need to select from alternatives they make these decision points clear, and they allow for repetition, iteration and recursion (Hall, 1992b; Juliff, 1984). Figure 2.5 could easily be altered to Figure 2.6: the flowchart is simply an altered version of the earlier figure. The alterations add more detail to the sequence, but they provide little else. The questions concerning cognitive actions rather than physical actions, and the level of detail continue to be unanswered, or indeed unanswerable.



continued over....

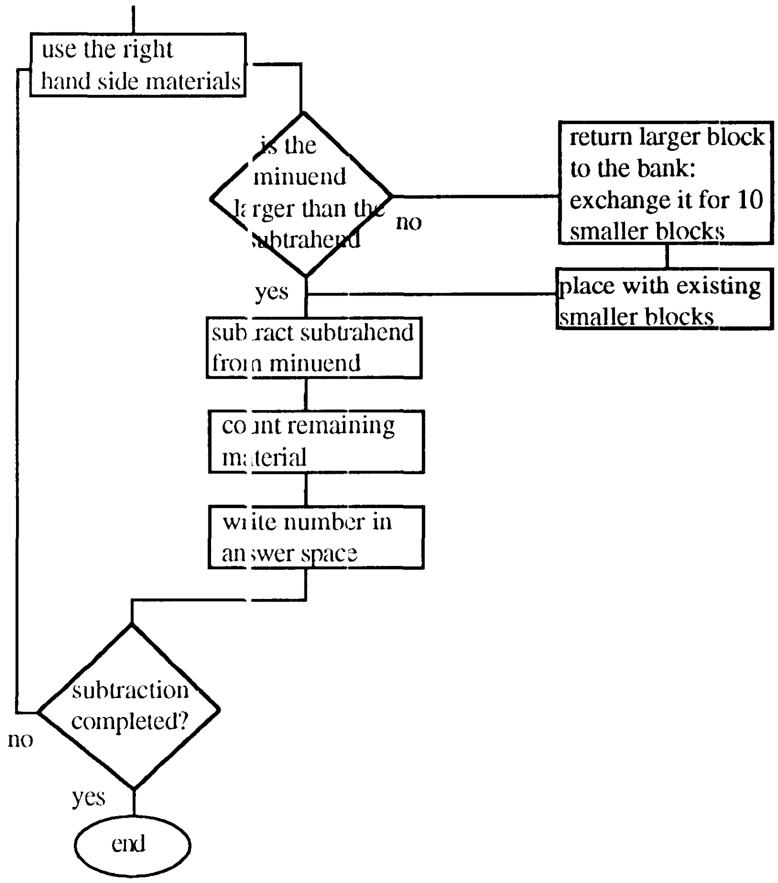


Figure 2.6: Flowchart for the question 57 -19.

The above figures do have one other value though. They show that a series of steps is necessary to complete questions such as this, perhaps in more detail than one would at first suppose. They also provide examples of chunking, where each of the steps listed is a one-step representation of a more complex physical/cognitive action. Thus, each step in the trace or flowchart could be enlarged to show more detail of the cognitive actions that are involved. For example, the exchange of ten smaller blocks for one large one could be thought of as twelve steps: take the larger block to the bank, count out ten smaller blocks (one, two, three ...ten), now take these back to the existing smaller blocks. With practice these moves are automatised and become one step: 'trade'. At the same time, the figures provide an illustration of one plausible instructional strategy, one in which the teacher has encouraged learners to automatise a number of steps into one chunk prior to coming to performing such arithmetic operations; in a sense, the prerequisite knowledge is at hand.

VanLehn and Brown (1980) also considered representing solutions to arithmetic algorithms as procedural nets, as in Figure 2.7, but rejected this because they did not have the hierarchical nature or subprocedures that are typically found in computer programs designed to replicate these algorithms. Figure 2.7 is an incomplete modification of VanLehn and Brown's procedural nets illustrating standard subtraction, but sufficient to show the nets of interconnections in such representations.

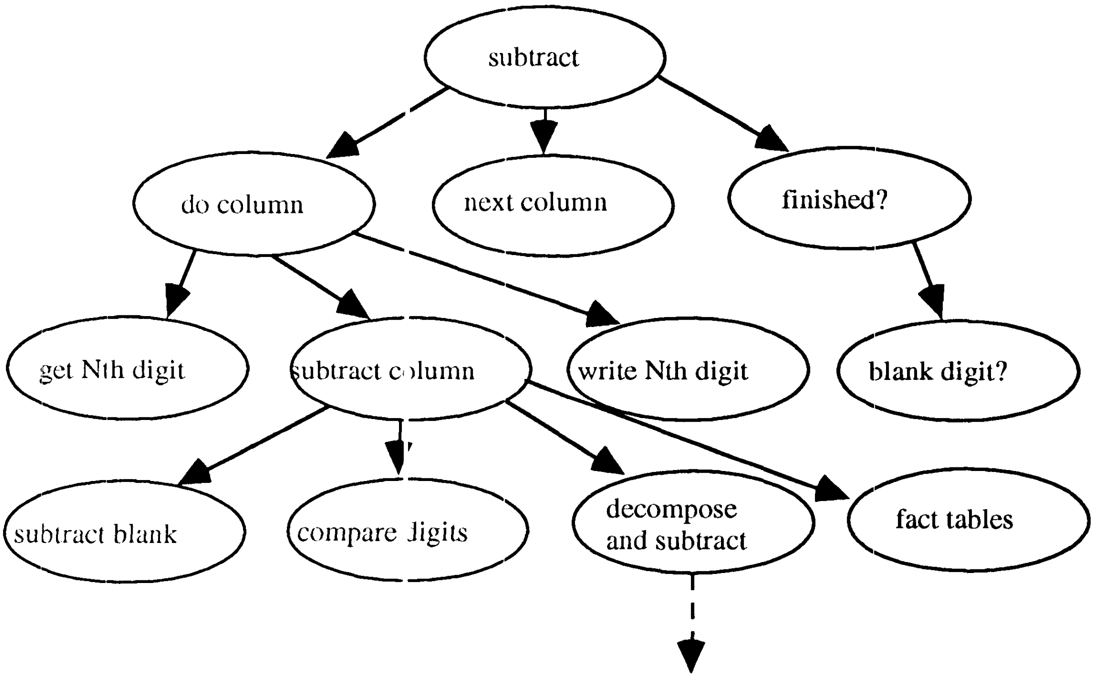


Figure 2.7: Procedural nets
(Modified from VanLehn & Brown, 1980 p.106)

Procedural nets are complex and though they do move beyond the representation of solely physical actions, they do not represent the detail necessary for a representation of cognitive actions. VanLehn and Brown (1980) felt they were not of value in establishing analogies since they did not provide sufficient detail for an analogy to be rated 'close'. These nets may have more value than suggested by the present literature: but not in their present format, which is indeed cumbersome, and rapidly becomes more so if the more abstract cognitive actions are to be represented. Reference to the later chapter on the procedural analogy theory will show some similarities to representation of algorithms in that theory and their representation as planning nets here. This comparison will also illustrate the complexity of developing planning nets, described below, as contrasted to developing algorithmic sequences in the procedural analogy theory, and the simplistic view of planning nets as a representation of arithmetic procedures in comparison to the detail of those developed under the procedural analogy theory.

Planning nets are directed graphs where the nodes represent flow charts for decision making and the links show inferences (VanLehn & Brown, 1980). That is, for any given procedure the node provides the choice of actions to solve that particular part of the procedure, the link directs the solver to the next step (node) in the problem's solution. Such planning nets not only show a sequence of structure, but also show reasoning and intent. VanLehn and Brown claim their planning nets provide a complete representation of the particular procedure.

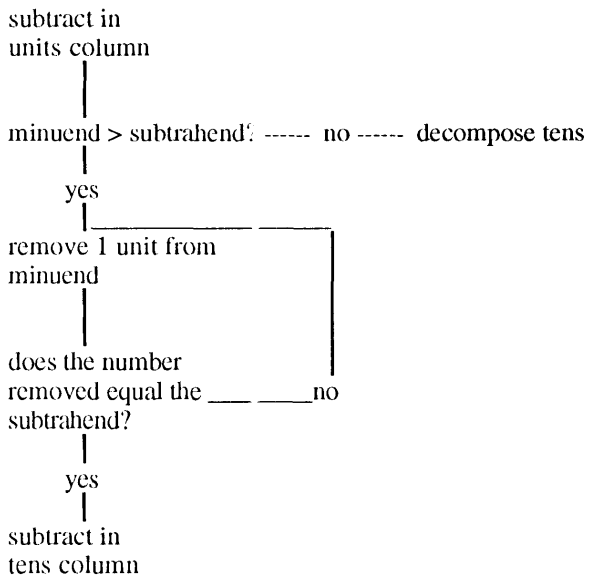
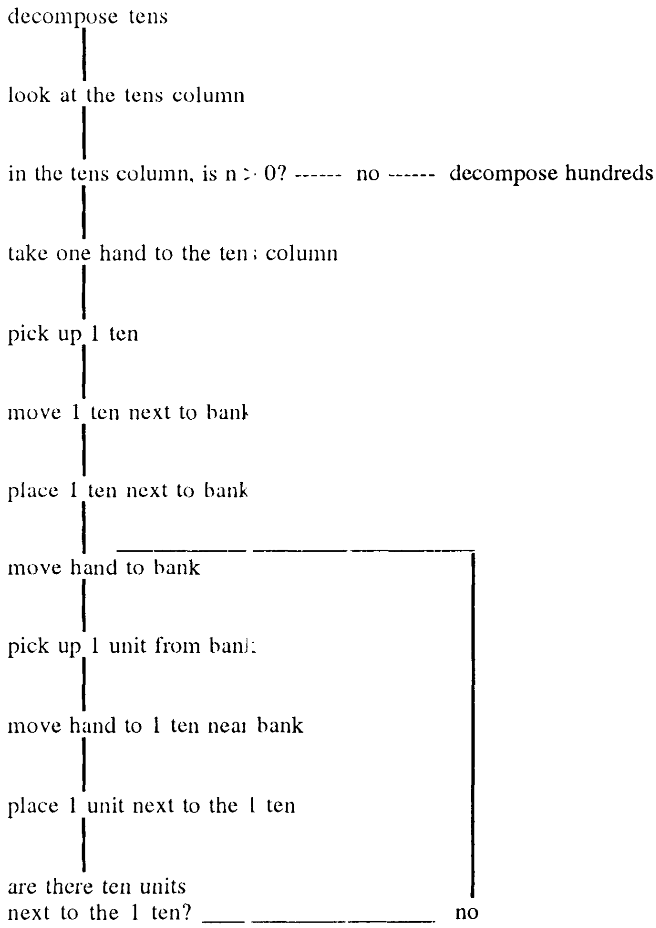


Figure 2.8: Subprocedure in planning nets: subtraction in units
(Modification of VanLehn and Brown, 1980, p.112.)

Figures 2.8 and 2.9 show two steps in the sequence of solving a subtraction algorithm using MAB materials. That is, these figures represent sections of the more complete planning net shown in Figure 2.10.



continued over ....

move 1 ten to bank

return hand to 10 units next to bank

collect ten units in one hand

place these ten units
with the existing units
in the minuend

Figure 2.9: Subprocedure in planning nets: decomposing tens
(Modification of VanLehn and Brown, 1980, p. 115.)

Figure 2.10 represents the planning net for whole number subtraction using MAB materials. These planning nets may allow a comparison between nets, for example, where the net for subtracting with MAB materials is contrasted with the net for a subtraction algorithm. In this way it would be possible to measure the similarities and differences between the nets. That is, to assess the extent to which the two nets are analogous. Secondly, the planning net for a complete operation indicates the value of chunking and automatisation, in terms of efficiency of operation and in terms of effective learning. Without chunking, there would be numerous sidesteps as the procedure was completed, such as represented by Figures 2.8 and 2.9, and the learner would have to divert from the main procedural sequence to complete a subprocedure. Without automatisation, there would be continual sidesteps, and both these and the main procedural sequence would constantly be interrupted as the learner worked laboriously from step to step.

look at written algorithm

represent minuend
with MABs ________ ________ are there 1000s _____ Y _____ represent 1000s
N

are there 100s _____ Y _____ represent 100s
N

are there 10s _____ Y _____ represent 10s
N

are there 1s _______ Y _____ represent 1s
N

subtract in units column

Figure 2.8

write 1s in unit answer space

are there 10s ________ Y ________ modified
N Figure 2.9

subtract in tens column

continued over ....

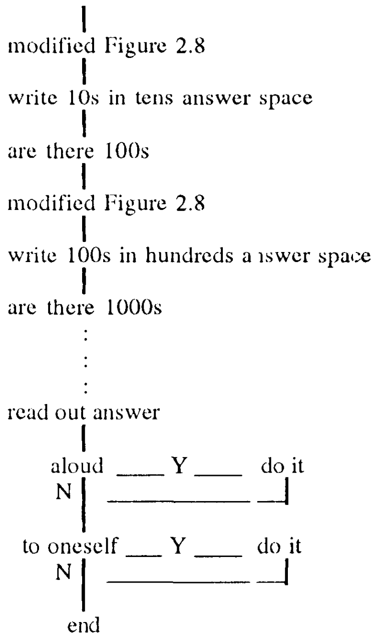


Figure 2.10: Planning nets: main procedural sequence
(Modification of VanLehn and Brown, 1980, p. 115 & p. 117)

These planning nets raise a series of questions. Is this what we want learners' cognitive structures to look like? Do these nets reflect what learners really do? Is there any way of quantifying the comparison between nets? Do the nets distinguish essential steps from the non-essential, the sensible from the unnecessary? Are the nets helpful from an instructional point of view? Individuals' cognitive structures may have some elements in common with these nets, though they may not reflect them exactly. This may be a reasonable representation of what some learners do, but individuals will chunk different steps, and will have different cognitive structures. Further, the representation may have a number of arbitrary decision-making points, which will again individualise cognitive networks. The nets do not distinguish the necessary from the unnecessary, though they do infer some areas where chunking may take place, and are helpful from that point of view. However, from a pedagogic point of view they are difficult to construct, and so describe a representation system that is problematic for teachers.

The various principles outlined by VanLehn and Brown provide useful insights into the development of a quantifiable, procedural instructional strategy. For example, Principle 1 states that in contrasting the planning nets of two procedures with the same goal, there will be a number (C) which will indicate the differences in constraints between the planning nets. They then go on to represent, in symbolic form, further principles:

Principle 3.	For any $i \neq j$	$d(p_i - p_{i-1}) \cap d(p_j - p_{j-1})$	$= \emptyset$
Principle 4.	For any $i \neq j$	$d(p_{i-1} - p_i) \cap d(p_{j-1} - p_j)$	$= \emptyset$
Principle 5.	For any $i \neq j$	$d(p_i - p_{i-1}) \cap d(p_{j-1} - p_j)$	$= \emptyset$

(Note: $\cap$ means 'intersection')

In general, these principles outline the rules for which planning nets are generated and contrasted. They may be useful in terms of machine learning and generating computer solutions, but they are complex and incomplete in terms of human learning, and are not especially helpful in terms of pedagogy and classroom practices. VanLehn and Brown's hope "to develop a representation of all principled sequences to a given target procedure" (p. 128) may be important from a cognitive science perspective, but it will need to be simplified if it is to be useful to teachers in classrooms.

Consequently, the analysis provided by VanLehn and Brown is helpful but limited. It is true, as VanLehn and Brown claim "the constraints and heuristics that appear in the net represent, in some sense, the *essence* of the procedure" (p. 120). At the same time, their analysis is not especially useful in classroom contexts, in that their approach is pedagogically naive. For example, in step 13 in explaining the teleologic of some examples of subtraction, they indicate that 15 - 3, can be thought of as 14 - 2, then 13 - 1 and finally 12 - 0. It is doubtful, though, if teachers would do this in a classroom at the stage of developing an algorithm. Children would have practised number combinations in all kinds of activities and situations prior to coming to questions of this type. To ask 15 - 3, if a child is not versed in 5 - 3, is inefficient for both the teacher and the learner. 15 - 3 has to be seen as 10 + (5 - 3) = 10 + 2 = 12, and then as 15 - 3 = 12. Their second example, where 25 - 7, becomes 24 - 6 and so on, until it becomes 20 - 2, which they then suggest becomes 10 + (10 - 2) is another inefficient approach. If this approach is intended to reflect the manipulation of MAB materials, then the MAB is being used incorrectly. It is clearly the intention to arrive at a symbolic written procedure where 5 - 7 cannot be done, the learner thinks 'trade', and so the question becomes 15 - 7. The argument above is repeated here: you would not attempt questions like this in a classroom, if the learners were not experienced in the number combinations needed. Further, the step of 25 - 7, becoming 20 - 2 hides the step in the written algorithm where trading takes place; that is, the representation is not leading to a high level of correspondence with the written algorithm. The correct isomorphism requires that MAB materials show a trade, where a ten is traded for ten units, which are then added to the existing units prior to the subtraction taking place. While acknowledging VanLehn and Brown's note that "this chapter is more a proposal to investigate a promising line of thought than a report on completed research" (p. 132), it remains important to stress yet again that machine learning and human learning may have a deal in common, but teaching people has a range of subtleties quite different from the subtleties required in machine learning.

VanLehn and Brown (1980) mention "the incredible amount of work that goes into analysing a procedure in terms of its planning" (p.120). VanLehn and Brown describe their process as "(f)irst one constructs the flowchart, then the constraints and a sequential plan for the flowchart, and last calculates the planning net by noting which planning inferences are not ordered with respect to each other" (p.120). The Procedural Analogy Theory, described later in Chapter 4, makes an interesting comparison with VanLehn and Brown's ideas. In particular, the procedural analogy theory provides a much simpler approach to planning and comparing instruction strategies.

VanLehn and Brown appear to have an intention to provide a pragmatic theory, useful in teaching and learning. For example, they ask a series of questions, "Exactly how close to standard arithmetic procedures can procedures built around a particular representation of numbers, say Dienes Blocks, be made to be?" and "Can a Dienes Block procedure be devised that is totally isomorphic to a standard written procedure?" (p. 130). They note, though, that they are not making "any claims that the process of building a planning net, either for analogy or design, exactly models the human process of building a planning net" (p. 131), but the procedural analogy theory shows just this: that by beginning with concrete materials, and then applying analogy and simplification, learners build an efficient cognitive structure that reflects the procedure being used.

Chunking

Reference has been made to the concept of chunking on several occasions, it will now be considered in more detail. A system, human or machine, is engaged in practice when it repeatedly performs one task or a set of similar tasks. Newell and Rosenbloom (1981) have coined a rule, *the power law of practice*, which states that as humans practise a task, the time taken to complete that task improves as a power-law function over the number of performances of the task. That is, when expressed graphically as in Figure 2.11, with the vertical axis representing the logarithm of the performance score and the horizontal axis representing the logarithm of the trial number, the resulting graph approaches a straight line, especially as the

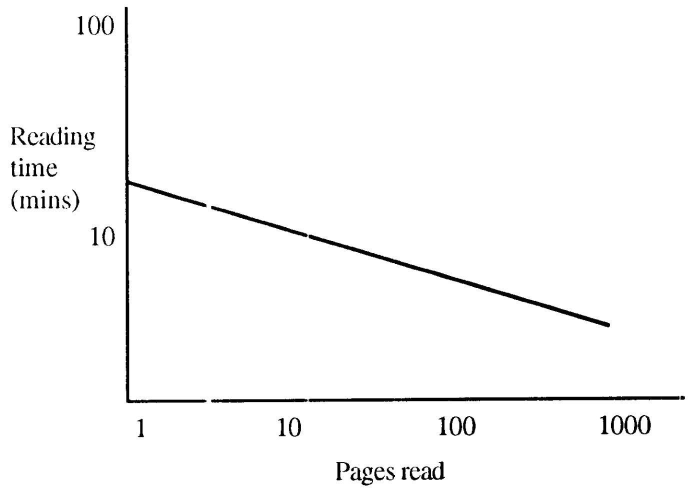


Figure 2.11: The Power Law of Practice:
learning to read inverted text (log-log coordinates).
(Modification of Rosenbloom and Newell, 1986, p. 251.)

number of trials increases beyond ten or twenty. The results hold over the entire domain of human performance, including perception tasks (Neisser, Novak & Lazar, 1963) and cognitive tasks (Neves & Anderson, 1981), but equally apply to some more esoteric applications such as cigar making, as an instance of motor-perceptual coordination (Crossman, 1959, as cited in Newell & Rosenbloom, 1981, p.6). Rosenbloom and Newell report on its longevity (Snoddy, 1926) and its applicability across a range of domains, such as perceptual motor skills (Crossman, 1959), perception (Neisser, Novak & Lazar, 1963), motor behaviour, elementary decisions, cognitive skills and problem solving. Results in this area of research have been consistent since the mid 1920s.

Newell and Rosenbloom (1981) hypothesised that the reason for the success of the law of practice was related to the cognitive architecture of the human brain, and in particular, to chunking, a concept also well established in the literature as present in all humans. They (Rosenbloom & Newell, 1982a, 1982b) developed a production system showing the chunking theory of learning, and so showed that the theory was a viable model of human practice and a viable mechanism for artificial systems. They suggested that practice improved performance because the learner became familiar with patterns in the task, and these patterns were chunks (Miller, 1956). For example, by replacing a string of several letters or numerals by a single chunk, the learner's memory load is reduced, and more data can be handled in working memory. Rosenbloom and Newell (1986) extend the notion of a chunk, as more than a set of data: for them, a chunk consisted of three processes, encoding the data, mapping it, and decoding it. For example, reading a phone number (a many-to-one mapping), mapping the stimulus symbol to the response symbol (a one-to-one mapping), then decoding the response (a one-to-many mapping). In their view chunking is a process, not just the result of some simplification process. The acquisition of chunks speeds up performance by reducing the number of mappings that have to be performed, by replacing the normal processing of a goal and its subgoals with the faster process of encoding, mapping and decoding.

In terms of sensible instructional strategies, a teacher would have encouraged learners to automatise steps into one chunk, prior to coming to the use concrete materials in a more complex arithmetic operation. This is consistent with effective teaching and with classroom practice, and it reflects a situation common in classrooms, where theory is frequently used to explain effective practice, rather than to develop it. In this sense, the development of theory is not keeping pace with classroom practice. The past decade has seen the emergence of a new paradigm for the study of learning based on information processing concepts (Anderson, 1983; Holland, Holyoak, Nisbett, & Thagard, 1986; VanLehn, 1988). The learning scenario most frequently described in these theories is the acquisition of a cognitive skill through practice. These theories assume the sources of information available to the learner are demonstrations (solved examples), and feed-back about the correctness of either final answers or single problem solving steps. The notion of *instruction* is largely absent from these theories, and no theory of the cognitive function of embodiments has been proposed. The procedural analogy theory proposes such a theory, and illustrates good instructional design (Ohlsson & Hall, 1990).

Intelligent tutoring systems

The purpose of this section is to explain intelligent tutoring systems, and both to link the section on cognitive science with such systems, and to provide further theoretical background to the procedural analogy theory and to the research reported in this thesis. In particular, this section of the thesis will describe intelligent tutoring systems, and discuss some of the theoretical and practical problems that exist in their construction.

For Sleeman (1983) an intelligent tutoring system has two functions: one is the production of responsive teaching systems; the other stems from the theoretical interest in establishing algorithms that create such systems. That is, such systems may be useful teaching devices, but they may also tell us something about how people learn. Developing an intelligent tutoring system requires knowledge of the domain of instruction, a model of the student which the systems modifies as the student uses the system, a teaching model outlining effective pedagogy in the domain, rules for applying this teaching model to the data accumulated in the student model, and an easy-to-use computer-human interface. There is, of course, the need to be able to write production rules and computer software that accurately reflect all these data and constraints (Anderson, 1988; O'Shea, Evertsz, Hennessey, Floyd, Fox & Elsom-Cook, 1988; Polson & Richardson, 1988; Self, 1988; Sleeman, 1983; VanLehn, 1988).

Numbers of such systems have been developed: with early systems, such as SOPHIE and GUIDON, less than perfect. The shortcomings acknowledged by their creators include the inability for the system to accurately interpret students' responses so as to identify the optimal instructional step, the domain limitations of each system, the limitations of the pedagogy where instruction sometimes seems little more than ad hoc, and the limits placed on the reactions possible by learners using the system. More recent systems such as LISP Tutor (Reiser, Anderson & Farrell, 1985) and the Geometry Tutor (Anderson, Boyle & Yost, 1985) are quite effective in modelling procedural knowledge. Typically the latter systems are based on a set of if-then rules in which "their recognise-act cycle capture(s) the basic data-driven character of human cognition" (Anderson, 1988, p.35). The systems remain in the developmental phase, but because these advances in production systems allow for the possibility of modelling human learning, they are nonetheless important[9] (Anderson, 1983; Holland, Holyoak, Nisbett & Thagard, 1986; Langley, 1985; Payne, 1988; VanLehn, 1983). Indeed Anderson (1983) claims

---

[9] Polson and Richardson briefly describe a number of intelligent tutoring systems (1988, Appendix 1). For example, the ACM system that automatises the construction of cognitive processes, Algebraland used for the study of acquiring problem solving skills in the context of Algebra the Geometry Tutor to study geometric proof, the Leeds Modelling System determines sources of error in algebra problem solving, Guidon for medical diagnosis, and Steamer simulating a steam propulsion plant.

that intelligent tutoring systems are built so as to learn more about the systems themselves rather than to provide something useful. If intelligent tutoring systems are to be useful in human teaching and learning, their specification and development require considerable clarification of the nature of learning, the nature of teaching and the nature of the subject matter.

An intelligent tutoring system could also be thought of as an expert system. An expert system is a computer-based system where the knowledge of an individual or group in a particular field has been computerised through in-depth discussion and the development of rules by the domain experts, knowledge engineers and computer programmers. Such a system allows a practitioner in a field access to more expert knowledge, and it allows non-experts access to specialised knowledge. For example, expert systems have been helpful in medical diagnosis, in quality assurance in production systems and office routines, and in geological exploration to mention but a few applications. If it were possible to provide a learner with an expert system, then learning could well occur through exploration and discovery – but this would be inefficient. The learner would need assistance to cope with the knowledge, structure and complexities of the expert system – and so we come to the need to develop a teaching model and a learning model. This leads us back to the difficulty of our incomplete understanding of teaching and learning, and the complexities of computer-based systems using machine learning in intelligent ways to assist instruction. That is, we are back to developing an intelligent tutoring system.

An intelligent tutoring system may be based on a sequential set of skills which are regularly tested, where success on the evaluation task means the student can progress to the next step in the curriculum, or learning could be arranged to be less hierarchical where the learner is more in control. Through diagnosis and inference of a student's responses to the system over time, the student model component of an intelligent tutoring system represents that student's current state of knowledge on a topic. Intelligent tutoring systems may also give advice based on student responses, or they may generate problems as required rather than selecting from a given list, and they may be able to adapt their explanation to better suit what the student already knows (Payne, 1988; VanLehn, 1988). As Littman and Soloway state

> As an ITS interacts with a student, it builds up an understanding of the student's knowledge and skills, which it uses to interpret the student's behaviour and, in part, to guide its own actions (1988, p. 212).

Payne (1988) acknowledges the importance of the student model and argues that such a model "must be generated from (or at least implicitly based upon) a theory of cognitive skill and its acquisition" (p.69), and Anderson supports this but notes that "(w)e still need to deepen our understanding of human cognitive processes and how they can be model(l)ed)" (1988, p. 48). Some intelligent tutoring systems allow a high degree of student choice, so allowing a range of routes through problem solving and learning tasks (Glaser, Raghavan, Schultz, Channarasappa, Chalawsky, Schauble, Katterman, Merchandt & Bowen, 1988; Glaser, Shute, Bonar, Raghavan, Schultz & Katterman, 1989; VanLehn, 1988). These systems do not direct learning but track the student's cognitive processes. For example, Anderson's computational theory of learning can be used to generate an ideal student model for the task at hand, which forms the basis for moving students from their present knowledge, as displayed in errors, towards the ideal student model.

Student models in intelligent tutoring systems are constructed through specially prepared task modelling, as in the Leeds Modelling System, by contrasting learner response to a predetermined set of bugs, as in the BUGGY/DEBUGGY systems, or by inferring from learner responses, partly the case for Anderson's ACT systems (Gilmore & Self, 1988). Gilmore and Self describe an application of machine learning to intelligent tutoring systems, where the techniques of machine learning are applied so as to develop and maintain individual models of students, and to provide a more dynamic student model. At the same time, Anderson argues that no intelligent tutoring system "actively uses a learning model in its computations" (1988, p.48). Once again, this provides a strong incentive to develop a theory of instruction appropriate to computer-based systems, and this in part is the point of this thesis, at least to the extent of testing a procedural analogy theory that has its raison d'être in intelligent tutoring systems and the development of effective instructional strategies.

Burton (1988) argues that for intelligent tutoring systems to be successful, their developers need to take into account the environment of learning at a computer, particularly the extent to which the system actively assists students' activities. Littman and Soloway (1988) support the idea of having a flexible student model allowing students more control as they structure their knowledge and skills. The achievement of such a capability means combining technological and psychological knowledge. Burton argues that notions such as constructivism, conceptual understanding and self-efficacy need to be incorporated into the specifications of intelligent tutoring systems.

Payne (1988) warns that the emphasis on production rules, which specify mechanisms for change, may prove a limitation on developing more effective intelligent tutoring systems. Given the machine learning background to production rules, this situation is understandable. That is, in order to have a machine learn, production rules will necessarily focus on a defined, programmable procedure, where the goals and steps towards it are defined. Such a view may be necessary to show that learning has taken place, especially in machine learning, but it has led to a deterministic view of learning encouraging developers of intelligent tutoring systems to ignore other cognitive contexts associated with human learning. Payne argues that there is a need to allow for a range of learning styles, which requires investigation into those qualitative differences that impact on learning in particular situations. Seemingly rising to this challenge are O'Shea, Evertsz, Hennessey, Floyd, Fox, and Elsom-Cook (1988) who describe the development of an intelligent tutoring system where the intention was not only to enhance students' arithmetic calculations and procedures but also to "foster mathematical understanding" underlying their use (p.257). O'Shea et al. argue that the point of applying artificial intelligence to education is to develop better quality individualised instruction, and that the key issue to achieving this is the student model used in the systems. They note that present models range from the open-ended ideas put forward by Papert (1980) in extolling the virtues of Logo and microworlds in general as a place for problem solving, discovery and learning through actions on objects, to the much more directed approaches of Anderson where students are moved toward the successful completion of conventional procedures. Their student model allows learners the flexibility of constructing their own arithmetic algorithms as they make cognitive constructions of the concepts and skills involved. O'Shea et al. have taken on the rather ambitious task of instruction to get it right the first time - "(w)e hope to pre-empt the development of buggy procedures by helping children to construct correct concepts in the first place" (p. 260). They criticise the limitations of production rules, arguing that learning is more than this, and that students' prior learning cannot be ignored: "a system modelling mathematical behaviour cannot ignore, but must attempt to build on, previously acquired knowledge" (p. 262). They aim to develop an intelligent tutoring system that "will track the development of individuals and attempt to explain their acquisition of mathematical skills over time" (p. 262).

Of course one of the problems with developing student models is our paucity of knowledge about students' learning and cognitive processes. While it is admirable to claim that a student-model generator "should be able to represent all student models which correspond to empirically observed behaviour" (O'Shea et al., p. 262), and that systems that fail to do this will inevitably disappoint some students, these intentions are longer-term goals rather than problems likely to be solved in the short term. The goal of developing a series of microworlds for developing understanding of mathematical concepts would certainly be a valuable contribution to education and artificial intelligence, especially if those microworlds can then be mapped onto procedural and conceptual knowledge to form powerful intelligent tutoring systems (ITSs). Taking all its components together, this is certainly a commendable and ambitious project, and indeed one can sympathise with its intentions; nonetheless, sensible scepticism implies awaiting the development and evaluation of their systems.

Intelligent tutoring systems have a good deal of potential, but our knowledge of production systems is insufficient for providing optimal intelligent tutoring systems which tend to be developed with little regard for cognitive fidelity. Intelligent tutoring systems differ from computer-aided instruction in the importance they place on student modelling, as Littman and Soloway say "ITSs are intended to understand students in a fundamental sense" (1988, p. 222). More needs to be known about human cognitive processes and how they can be modelled, and more about human knowledge acquisition (Payne, 1988). Anderson claims that the development of expert modules is in the hands of a few cognitive scientists, and argues that there is a need to teach "the use of cognitive science formalisations to curriculum developers" (1983, p. 49). Both Halff (1988) and Hall (1990) encourage researchers in this field to relate their work more to research findings in education and training, to develop a more precise theory of learning, and to learn more about good teaching. It is in this area that the procedural analogy theory may provide a theoretical base for the specification of important components of intelligent tutoring systems. It is not being claimed that the theory will provide the answer to a broad range of problems, but it appears to have the potential to enable a procedure for specification of computer-based intelligent systems in a defined area of mathematical knowledge. This idea is extended by Anderson who argues for the value of authoring systems based on production rules, that would require instructional-materials developers to complete only the expert module in order to have a tailor-made intelligent tutoring system. The potential of authoring systems as a vehicle for developing intelligent educational systems remains: though the researcher's experience suggests authoring systems for educational software are simplistic, partly in the manner in which the capabilities of such systems do not match the capabilities of the computer platform for which they were developed, and partly through an incomplete or non-existent model of student learning. The procedural analogy theory may have some potential in assisting the movement towards these various ends.

This chapter considered a wide range of issues related to cognitive science, in the context of both human and machine learning, and concerning the development of intelligent tutoring systems. Though this thesis is also concerned with mathematical learning, particularly the use of concrete materials as a form of knowledge representation that assists learning. So it is to this aspect of mathematics that this thesis now turns.